DEVELOPMENT OF NEW KNOWLEDGE DISCOVERY TOOLS TO EXPLORE BIOMEDICAL DATASETS IN BREAST CANCER

Nathan Stuart Hill

PhD

January 2009

Tenovus Centre for Cancer Research
Welsh School of Pharmacy
Cardiff University,
Cardiff
CF10 3NB
UK

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Summary

The explorative power of high throughput technologies in cancer research has become well established in recent years, exemplified by diverse gene microarray studies. However, development of the necessary biomedical data analysis tools has historically been confined to a commercial environment, while comprehensive, user-friendly analysis approaches are still needed. Availability of freely-available software, notably the 'R' project statistical programming language, allowed development of a user-friendly multivariate statistics application – Informatics Tenovus (I-10) – in this project. I-10 provides a platform through which powerful existing and future 'R' project statistical analysis methodologies can be applied, without prior programming knowledge. The new system was tested in the context of exploring antihormone resistance in breast cancer, analysing microarray datasets from in vitro models of acquired Tamoxifen (TAMR) or Faslodex resistance (FASR) versus endocrine responsive MCF-7 cells. The analysis not only revealed known de-regulated genes, but also further potential future markers/targets for endocrine response/resistance. The advantages of the 'R' programming environment together with Microsoft Visual Basic.net technology for producing user-friendly biomedical analysis tools facilitated subsequent development of a tool which could explore SEER cancer patient datasets. This new cancer query survival tool – Superstes –allows detailed statistical modelling of the impact that multiple patient attributes (in this instance derived from the SEER breast and colorectal cancer datasets) have on patient survival. The versatility of 'R' was additionally demonstrated in further exploring classifiers, where it was able to interface with the sophisticated, freely available machine learning application 'Weka'. Using 'R' and Weka, breast cancer patient survival was modelled using equivalent patient attributes to the Nottingham Prognostic Index and a 10 year survival subset of the SEER breast cancer dataset. Several machine learning methodologies were compared for their ability to accurately model survival, with their value in routine clinical use for prediction of patient survival then critically evaluated.

Acknowledgements

I would like to thank Dr Paul Lewis, Dr. Julia Gee and Professor Robert Nicholson for giving me the opportunity to study at the Tenovus Centre for Cancer Research as well as their expert support and guidance throughout this study.

Funding from the Tenovus Cancer research charity was gratefully acknowledged as well as the ability to attend conferences and meetings.

Special thanks to all Tenovus staff, Welsh School of Pharmacy and colleagues at the Swansea Medical School, Swansea University who have helped make my study a highly enjoyable and memorable experience.

Publications

Hill N, Gee JMW, Quirk P, Sharma-Oates A, Nicholson RI, Thomas G, Leonard R, Wagstaff J and Lewis P (2007). Superstes: cancer survival query tool. *Eur J Cancer* (submitted)

Lewis P, Hill N, Gee J, Nicholson RI, Leonard R (2007). SUPERSTES - breast cancer survival query and report system. NCRI conference abstracts. (poster) www.ncri.org.uk/ncriconference/abstract/pdf/2007%20pdfs/B61.pdf

Contents

Chapter 2 – Informatics system Tenovus – 'I-10' development

Chapter 3 – Application of I-10 to *in vitro* endocrine response and resistance microarray data

Chapter 4 – Superstes – Development of a clinical cancer survival query tool

Abbreviations

ASCII	American standard code for information interchange
ANOVA	analysis of variance
APN	average proportion of non-overlap
AD	average distance
ADM	average distance between measures
ASCO	American society of clinical oncology
AU	approximately unbiased
BHI	biological homogeneity index
BSI	biological stability index
BP	bootstrap probability
BRCA	breast cancer susceptibility protein
CoA	correspondence analysis
DAVID	database, annotation visualisation and integrated discovery
DCOM	distributed component object model
DMT	data mining tool
DNS	domain name server
EST	express sequence tags
FISH	fluorescence in situ hybridisation
FOM	figure of merit
HCA	hierarchical cluster analysis
LINUX	operating system created by Linus Torvalds
PAM	partitioning around medoids
OLEDB	object linking and embedding database
ORF	open reading frame
PDF	portable document format
ER	oestrogen receptor
ERα	oestrogen receptor alpha
HER2	human epidermal growth factor receptor 2
QPCR	quantitative polymerase chain reaction
IDE	integrated development environment
IHC	immunohistochemistry

PCA	principal components analysis
RMA	robust multi-array
RNA	ribonucleic acid
RT-PCR	reverse transcriptase PCR
SAM	significant analysis of microarray
SAGE	serial analysis of gene expression
SERM	selective oestrogen receptor modulator
SOM	self organising maps
SQL	structured query language
SVM	support vector machine
MAS5.0	microarray suite 5.0
MDS	multidimensional scaling
GCRMA	GC robust multi-array
UCLA	university of California, Los Angeles
VB	Visual Basic programming language
VBA	Visual Basic applications

Chapter 1

Introduction

1 Introduction

Breast cancer is the second most prevalent type of cancer throughout the world after lung cancer, with an incidence rate in women in the UK of 1 in 9, according to cancer research UK (Office for National Statistics, 1999) [1]. The majority (95%) of breast cancers are sporadic, yet a small proportion are familial where 5% of these may relate to loss of function of the genes *BRCA 1 and 2* (Thompson et al, 2008) [2]. Breast cancer is a multifaceted disease in terms of genetic, phenotypic and clinical characteristics, where decision making by oncologists for the most effective treatment regime depends upon clinical and pathological prognostic and predictive factors. Importantly, significant improvements have been made in the way breast cancer is diagnosed and treated. Improvements are due to new research discoveries which have improved survival by over 20% in the last 10 years (Rakha et al, 2008) [3]. For example, improved mammographic screening resulted in detection of the onset of early invasive disease and also ductal carcinoma in situ, a pre-neoplastic condition with the potential to progress to invasive disease (Yaffe et al, 2008) [4].

A further landmark has been discovery of steroid hormone signalling via the oestrogen receptor-α (*ER*) playing a central role in the growth and development of breast cancer (Yaffe et al, 2008) [4]. Epidemiological studies indicate that increased risk of breast cancer is associated with cumulative life-time exposure to steroid hormone-related factors, having associations with an earlier menarche, late menopause and pregnancy. Discovery of the importance of such signalling has provided a mechanistic target to selectively treat and improve outcome for many patients through use of various anti-hormonal agents (including anti-oestrogens such as Tamoxifen and Faslodex, and also oestrogen deprivation treatments notably aromatase inhibitors). Similarly, discovery that *c-erbB2* amplification can contribute to growth and aggressive behaviour of some tumours has resulted in Herceptin (Trastuzumab) antibody therapy (Kapp et al, 2006) [5].

Historically, there has been a good overall association between the standard clinico-pathological covariates currently used in breast cancer management and patient's outcome. Prognostic factors enable identification of patients whose prognosis is either good enough to not warrant adjuvant systemic therapy after local surgery of the tumour or poor enough to justify a more aggressive

adjuvant approach. Secondly, covariates can be predictive, enabling selection of patients whose tumours are more likely to be responsive or resistant to a particular type of therapy. Of considerable interest in the latter regard is steroid hormone receptor status, where *ER+* tumours (a feature of ~70% breast cancer patients) are enriched for anti-hormone responses, which are largely absent within *ER* negative (*ER-*) patients. Equally, responses to Herceptin are confined to patients whose tumour cells have amplification of the *c-erbB2* gene.

Unfortunately however, significant proportions of patients relapse following treatment, and ultimately will die from progression of the disease (Slamon et al, 1987) [6]. There is therefore an increasing need for additional prognostic and predictive factors both to improve patient risk accuracy, to improve targeting of existing treatments to those who will truly benefit, and equally to determine further tumour targets for development of new therapies.

1.1 Current Breast Cancer Prognostic Factors, Impact on Treatment Strategies, and Limitations

Histopathology supplies a substantial amount of information through routine examination of breast cancer (and associated lymph node) sections allowing generation of morphological prognostic factors such as tumour size, differentiation in terms of histological type and grade, as well as lymph node stage and vascular invasion. Prognostic groups according to such measurements have been created which can subset patients according to their chance of survival over a 10 year period. The Nottingham prognostic index (NPI) is a clinicopathological classification system based specifically on tumour size, histological grade, and lymph-node status (Galea et al, 1992) [7]. The higher the NPI value the worse the prognosis. It was one of the first systems to be developed to show a correlation between the three different parameters and adverse outcome. It was developed before high throughput technologies such as microarray analysis became available in the late 1990's and subsequent studies confirmed the value of the combination of lymph node stage, histological grade, tumour size could improve prediction of prognosis. The NPI system is still widely used throughout the UK for breast cancer prognostication (Galea et al, 1992) [7].

However, it is acknowledged that histopathological parameters are also associated with certain limitations, e.g., many histopathological variables (such as grade) are subject to significant pathologist variability even after many attempts of standardisation (Yu et al, 2004) [8], and appropriate cut-off points are often difficult to define when the histopathological parameter being measured is scored over a continuous range of values (Yu et al, 2004) [8]. Indeed, there is still little general agreement as to which tumour prognostic factors should be used routinely in clinical practice. The only factor used consistently as a guide for therapy to date has been lymph node status however this alone is incapable of identifying patients who have 100% risk of death from breast cancer. The inaccuracies of such prognostic indices is further confirmed by several studies that have shown that approximately one-third of lymph node-negative breast cancer patients who are classified within a 'good prognostic group' actually go on to develop disease recurrence (Feng et al, 2007) [9], while a similar proportion of node-positive patients paradoxically remain free from development of distant metastases (Feng et al, 2007) [9]. Prognosis of breast cancer also depends upon the presence of distant metastases, and evaluation of intrinsic biological characteristics to further indicate aggressive behaviour of the tumour, for example by examining growth rate (e.g. using *Ki67* immunostaining), may also be important (He et al, 2006) [10]. In total, taking all these parameters into account suggest an improved prognostic index may need to include both time-dependent and biological information. Indeed, it is becoming increasingly established within the breast cancer research community that current prognostic factors fail to adequately reflect the clinical and molecular heterogeneity of the disease, and in some instances prove inaccurate when used to direct management decisions (Rakha et al, 2008) [3]. Of note, clinical studies of individual gene expression has revealed there are distinct sub-classes of breast cancer (Perou et al, 2000) [11], with the concept of sub-classes being re-capitulated at the protein level (Yu et al, 2004) [8] and where such sub-classes appear to have bearing on prognosis (Modlich et al, 2006) [12]. Such concepts will be discussed in detail in a later section.

Clinically, the NPI has also proven valuable in assessing criteria for receiving adjuvant systemic antihormone therapy in primary operable breast cancer. If patients fall into a good prognostic group, Tamoxifen treatment has shown to give a good survival outcome. However due to certain side effects such as increases in endometrial hyperplasia and occasionally endometrial cancer; it could be argued that there may be patients with an inherently good prognosis where Tamoxifen

treatment could be withheld. In the poor and moderate prognostic groups revealed by the NPI, additional factors are taken into consideration notably *ER* status and menopausal status. For example pre-menopausal, *ER+* patients can receive ovarian suppression using Zoladex and tamoxifen in combination (Gnant et al, 2008) [13]. Post-menopausal, *ER+* tumours receive adjuvant tamoxifen (or increasingly aromatase inhibitors). *ER-* patients would receive chemotherapy treatment as they predominantly lack the target receptor for anti-hormones; however those patients too weak to receive chemotherapy may be given hormone therapy as up to 5% response rates can occasionally be observed in *ER-* disease. Of note, treatment for tumours also over expressing *c-erbB-2* protein can include Herceptin which also serves to increase sensitivity to chemotherapy, inducing apoptosis more readily (Dahabreh et al, 2008) [14]. However, again the prognostic factors prove inaccurate. 40% of ER+ tumours fail to respond to anti-hormones and have an inherently poorer prognosis (Harris et al, 2007) [15], while a proportion of initially responsive patients subsequently relapse during treatment despite retention of *ER* positivity, again an event ultimately associated with poorer outlook (Belkhiri et al, 2008) [16]. There are clearly further factors determining growth and progression of some *ER+* tumours and hence durable response to anti-hormonal agents. Equally, responses to Herceptin are confined to ~30% of *erbB2*-overexpressing patients with relapses again a problem in these initial responders (Belkhiri et al, 2008) [16].

Together with using the NPI system, the St Gallen and NIH conference has also outlined guidelines for the eligibility of adjuvant chemotherapy, again based on tumour histological and clinical characteristics in relation to predicting outcome after diagnosis (Modlich et al, 2006) [12]. Results according to these guidelines showed that along with lymph node positive disease, up to 90% of lymph node negative early breast cancer patients are candidates for consideration of adjuvant systemic treatment. However studies have shown that many would remain disease free, where such over treatment may incur unwanted side effects (Van't Veer et al, 2002) [17].

Clearly, there are inherent inaccuracies in the current prognostic indices and also limitations in their effectiveness in predicting treatment. However, there is considerable scope to improve upon the NPI and existing prognostic factors used to select patients for therapy using new computational techniques as applied to microarray data and expanded datasets offering extra

covariate information such as that available through the SEER program in the USA (Edwards et al, 2005) [18].

1.2 Approaches to Discover Improved Prognostic Markers and Treatments for Breast Cancer

It has become clear that patients with similar clinical and pathological features may show distinct prognostic outcomes and also vary in their response to therapy. Recent advances in biomedical data modelling and high throughput technology may improve understanding of these phenomena.

The phenotype of cells – encompassing growth, differentiation and migratory behaviour of malignant (versus benign) cells- can classically be measured through protein expression studies, often examining individual or small numbers of proteins (e.g. using immunohistochemistry and western blotting) which can be used as prognostic/predictive markers (e.g. *ER* and *HER2* measurement), as well as to understand mechanisms underlying response and failure to current therapies and to derive new drug targets. Many research groups are active in this area, drawing on clinical material and in some instances experimental models. For example, *in vitro* examination of *ER+* breast cancer cell lines such as *MCF-7* have confirmed the importance of *ERα*, a nuclear transcription factor, and shed light on its signalling mechanism (Lisztwan et al, 2008) [19]. Scientists in the Tenovus centre for cancer research, Cardiff University, have derived a model from the *ER+ MCF-7* breast cancer cell line *in vitro* to understand resistance to Tamoxifen (TAMR cell line). The TAMR cells have a very aggressive phenotype showing increased growth rate, motility and invasiveness (a feature that can be associated with antihormone resistance in the clinic) and has been found to rely on the tyrosine kinases *EGFR*, *erbB2*, *IGFR* and *c-Src* signalling (Frasor et al, 2006) [20]. It appears that a network of receptor tyrosine kinase signalling contributes to breast cancer growth and progression with pathways, interacting with *ER* when this is present. Consequently, it should be of no surprise that drugs which inhibit the candidate *EGFR*, *erbB2*, *IGFR* and *c-Src* pathways results in inhibition of TAMR cells *in vitro*. Research within the Tenovus Centre has provided proof of principal that useful biomarkers (e.g. *erbB* receptors such as *erbB2* and *EGFR*) and targeted therapies (e.g: *erbB* and kinase inhibitors)

can result from in depth study of breast cancer growth signalling biology in the TAMR cells (Nicholson et al, 2005) [21].

The emergence of genomic technologies since the turn of the century has facilitated exploration of multiple genes and associated pathway information to further explain the progression and development of breast cancer. Three key multigene tests have been developed which can be routinely used for analysis. These are summarised and compared in table 1.1.

The prognostic value of Immunohistochemistry (IHC) is well established for testing of *ER*, *PgR*, *HER2* and the proliferation marker *Ki-67*. Fluorescence in situ hybridization (FISH) differs in that it is generally used to determine the copy number of the *HER2* gene for treatment selection processes with Herceptin, for example. Quantitative PCR (QPCR) is very reliable due to its sensitivity of detecting RNA from very little starting material (Lisztwan et al, 2008) [19]. The quantitative polymerase chain reaction (QPCR) technique has the added advantage over the other techniques of being able to assess multiple biological processes simultaneously such as hormone receptor status, proliferation and *HER2* pathway information. It has been extensively used to predict overall prognosis and response to hormonal therapies (Lisztwan et al, 2008) [19].

Platform	**IHC**	**FISH**	**PCR**	**Microarray**
Number of genes tested:	Small	Small	Intermediate	Large
Type of measurement	Semi quantitative	Semi quantitative	Quantitative	Quantitative
Statistical algorithm complexity	Simple	Simple	Complex	Highly complex
False Discovery risk	Low	Low	Intermediate	High
Ability for Multiple pathway discovery	Low	Low	Intermediate	High
Individual prognostic value	Established	Established	Established	Established
Standardisation ability	Low	Low	High	High

Table 1.1 – Comparison of available breast cancer multi-gene predictor platforms (Ross et al, 2008) **[22]**

Although algorithms based on basic clinico-pathological data are now routinely used to define prognostically significant groups and to tailor systemic therapy for breast cancer patients (for example, Adjuvant! Online), further improvements are required. Combination studies of small numbers of existing markers stemming from techniques such as ISH, FISH and QPCR could have potential to reveal improved prognostic groupings when analysed using clustering techniques. This is particularly evident of the interaction between *c-erbB2* and also *c-myc* – an effect which is not seen in profiling of the individual genes – only in combination (Fei et al, 2002) [23]. Exploration of the combination effect that existing biomarkers can have could potentially be very effective.

As a result of human genome sequence completion, it is feasible that many more assayable markers of relevance to cancer behaviour and prognosis may be revealed through high throughput profiling of the whole molecular signature of human tumours. This is feasible through Microarray technology where a complete set of genes (up to ~40,000) can be measured in a single hybridisation experiment, in contrast to techniques such as Quantitative PCR which can only measure at best several hundred genes at a time. The microarray approach has the potential to reveal a detailed molecular description of malignant tumours which ultimately may lead to novel tumour markers and classification algorithms. In turn, these factors can be evaluated for their ability to improve prediction of clinical behaviour and potentially could encompass targets for new therapeutic approaches.

1.3 High-throughput Technologies and Associated Bioinformatics for Marker Discovery

High-throughput approaches using microarrays require robust bioinformatics strategies in parallel for successful and meaningful data analysis if we are to determine genes that can refine prediction of prognosis and treatment outcome, as well as provide targets to augment treatment regimes. An inherent goal of array analysis is the need to maximise the biological data that can be obtained from microarray technology. This requires optimisation of the identification of significantly differentially-expressed genes from microarrays, including consideration of

microarray platform, experimental design, expression call and normalisation. Data reduction to determine significantly different genes which encompasses filtering for false positives and inability to more efficiently obtain comprehensive gene annotation is also important. Additionally, array results are viewed in a way that differs to other more molecular views in that maps drawn from results show order and logic of the genetic 'program' as opposed to the order in which genes appear on individual chromosomes (Brown et al, 1999) [24]. The representation is the cornerstone of the value which microarray results can yield – association of individual genes with others showing similar expression patterns. Complex analysis strategies are involved; however there is considerable potential to assign signature components within regulatory pathways using different analysis methods, albeit as determined at the transcriptional level.

1.3.1 Microarray technology and experimental design

Multiple types of microarray platform exist. The technology roots are largely based in cDNA library and differential display studies which ultimately led to the first nylon and plastic microarrays. As engineering for array technology improved, whole genome microarrays have been developed; however, custom designed arrays using the *Cy3/Cy5* system remain popular in the biological research community– particularly for smaller genome organism studies such as in the Fugu fish (Bassett, 2001) [163].However, in the cancer research setting, well developed commercial platforms such as whole genome arrays from Affymetrix facilitate maximum exploration (Robinson et al, 2007) [25]. Disease specific arrays from Affymetrix (such as breast cancer) are also taking high throughput exploration to a further level of performance. The quality benefits of the Affymetrix technology are widely publicised stemming from its fundamental principle of utilising perfect match and mismatch oligonucleotides. The present/marginal/absent (PMA) expression call performs as an intrinsic quality control step as well as being an inherent part of the quantification technique (Robinson et al, 2007) [25]. The Affymetrix approach also is reported to be able to reveal weakly expressed mRNA species more accurately.

To produce robust results from a microarray, a number of fundamental steps need to be monitored starting with sample preparation. RNA is initially extracted from the samples under test, from which cDNA is then reverse transcribed, with subsequent hybridisation of the samples

to a chip such as the Affymetrix HGU-133A chip (Homo sapiens). An overview of this process is shown in figure 1.1. Scanning of the array occurs in a specialist facility with high resolution scanners approved by Affymetrix using Affymetrix software. Adopting this approach, the company believes 'quality can be assured' with potential consistency between array runs and facilities (Robinson et al, 2007) [25].

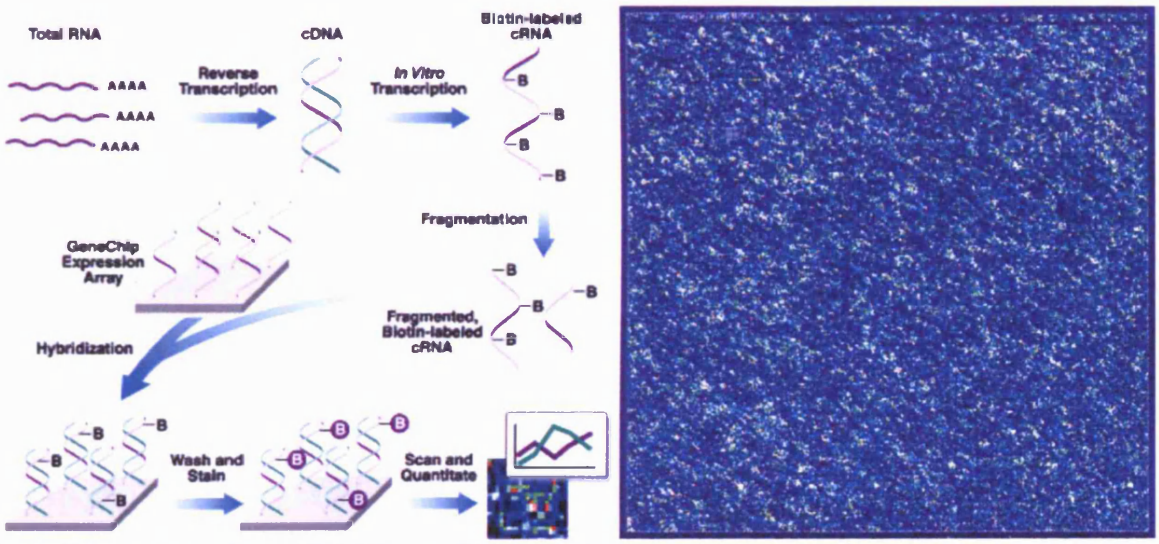


Figure 1.1: An overview of the Affymetrix process and a typical image of a chip after scanning. (Affymetrix, 2001) [26]

Initial experimental design is very important which encompasses the initial microarray technology choice and in particular the number of replicates required of samples to be arrayed. Subsequent analysis of microarray data concerns determining genuine expression level changes from 'noise' inherent in any assay. The way in which experimental background 'noise' is filtered from true genetic results is the first step for all array analysis. Although the Affymetrix system delivers the lowest signal to noise ratio of all the available Array platforms due to 11 sequences per gene including mismatches ("gene probe set"), dependency on a single replicate would still prove unreliable (Robinson et al, 2007) [25]. Careful cost analysis versus optimal number of replicates is often a difficult yet important balance to achieve due to the potential substantial benefits and research value array data can yield. Compromises made at this point could potentially jeopardise the analysis which could result, as a worst case scenario, in further sample sets having to be prepared and samples having to be re-arrayed. This can prove costly overall, especially if only an additional replicate was required.

A minimum of three replicates is generally advised by the literature; however recent experience has shown a need to calculate exactly the optimum number of replicates needed encompassing economic feasibility (Black et al, 2002) [27]. The way in which the optimal number of replicates is calculated varies according to the type of statistical testing which would be envisaged for a particular array results set. Two key calculations are widely accepted depending upon whether parametric or nonparametric statistics will be applied. Parametric methods, utilising traditional sample statistical analysis methods such as the t-test, follows the theory that gene expression data are normally distributed. In comparison, the Wilcoxon test is a non-parametric method based on ranking observed gene expression levels. A method outlined by Black and Doerge is widely accepted as a way of calculating the ideal number of replicates if only parametric methods are to be used (Black et al, 2002) [27]. However, for the ultimate in sensitivity and comparison, some studies suggest there may be more benefit in comparing results from an alternative array system to increasing replicate numbers beyond three (Pedotti et al, 2008) [28].

1.3.2 Improving quality of microarray experiments

Early pioneers of Microarray technology, although obtaining interesting research findings, as exemplified in clinical breast cancer microarray studies by Van't Veer et al (Van't Veer et al, 2002) [17], noted that there were certain shortfalls in the technique. Results have proven difficult to reproduce with many reasons to explain this occurrence including a lack of exact information as to how sample material was prepared, the number of replicates, and data preparation prior to statistical analysis. Consequently, the Microarray and Gene Expression Data (MGED) Society was formed to strive to improve quality and consistency of results in microarray experiments. The MGED society is an international organization of computer scientists, biologists, and data analysts that aims to facilitate the sharing of data generated particularly using microarray technology for a variety of applications including expression profiling (Brazma et al, 2001) [29]. The key emphasis is establishing standards for data quality, management, exchange and annotation whereby facilitating the creation of tools that enable these standards to be achieved (Brazma et al, 2001) [29].

As previously stated, the prime goal for using high throughput approaches is to maximise the biological data that can be obtained from any given sample. Often this involves refining the process to identify significantly differentially-expressed genes, for example improving the filtering procedure for false positives, detailed pattern discovery and approaches which more efficiently perform comprehensive gene annotation. This was one of the motivations for MIAME standardisation. To aid future design of MIAMI compliant array experiments, it is mandated that the inclusion of several pieces of minimum information is required to accompany a dataset of published results. These include:

I. Array design description information – type of array, chip information.
II. Experimental design – Authors, type of experiment, number of replicates
III. Samples used, extract preparation and labelling – Cell type, labelling protocol
IV. Hybridisation procedure and parameters - Sample and corresponding Array information
V. Measurements data and specifications of data processing – Image quantification and Normalisation

Without inclusion of such information, it will be harder in future for referees to accept microarray based research for publication in scientific journals.

1.3.3: Microarray analysis suite version 5.0 (MAS5.0)

Affymetrix recommends that scans created by the Affymetrix scanner are converted into a tabular form of individual intensity values using a software package called 'Microarray Analysis Suite 5.0' (MAS5.0). This was a current version of the application at the start of this project. Recent versions of an equivalent application offered by Affymetrix at the time of writing include the Affymetrix® Expression Console™ software. To date, this algorithm is still referred to as the 'MAS5.0' algorithm in current Affymetrix applications which perform this process. The algorithm uses a multistage process which uses fundamental design properties of an Affymetrix array. It was first launched with the release of the MAS5.0 application suite and remains routinely used in more recent software releases from Affymetrix (Affymetrix, 2001) [26].

The MAS5.0 algorithm transforms the scanned image light intensities (encompassing every probe) into a series of numbers using a file type called a CEL file produced by the Affymetrix scanner. The intensity of light emitted from a particular chip at different areas when scanned with a laser directly relates to the amount of expression of a particular gene in a particular sample at a particular moment in time. The software produces a table of this data in the form of a spreadsheet summarising the samples arrayed and the individual Affymetrix gene ID's to which each spot on the chip corresponds.

The MAS5.0 algorithm uses a multstage process. Firstly, background correction is performed by dividing the array into 'zones' and calculating an average background intensity. The design of an Affymetrix array having Mismatch (MM) and Perfect Match(PM) probes is also fundamental to the process. Each mismatch probe provides a direct measure of background and stray signal (due to cross hybridisation) for its perfect match partner. However the mismatch intensity can be higher than its perfact match value or lower. The algorithm uses smoothing so that there are not jumps in values between large and small values. As a result of all these features on an array, the Tukey biweight algorithm is then used to calculate a robust average signal from each probe. A log base 2 of the values is taken with an additional step to prevent any zero values. Finally, in case of slight manufacuturing variances and other experimental factors, results are scaled by ignoring the top and bottom 2% of expression value and a mean intensity calculated for those that remain. (Affymetrix, 2001) [26]. The resulting summary is usually produced in a tab-delineated format (figure 1.2) which can be viewed using spreadsheet applications such as Microsoft® Excel. For example, in Tenovus samples representing different treatments/resistant or responsive states would have been arrayed (with replicates) as part of the experimental design and consequently there is a need for the information to be subsequently collated and stored in a database, for quick retrieval for detailed analysis.

The initial launch page of the MAS5.0 application can be observed in figure 1.2

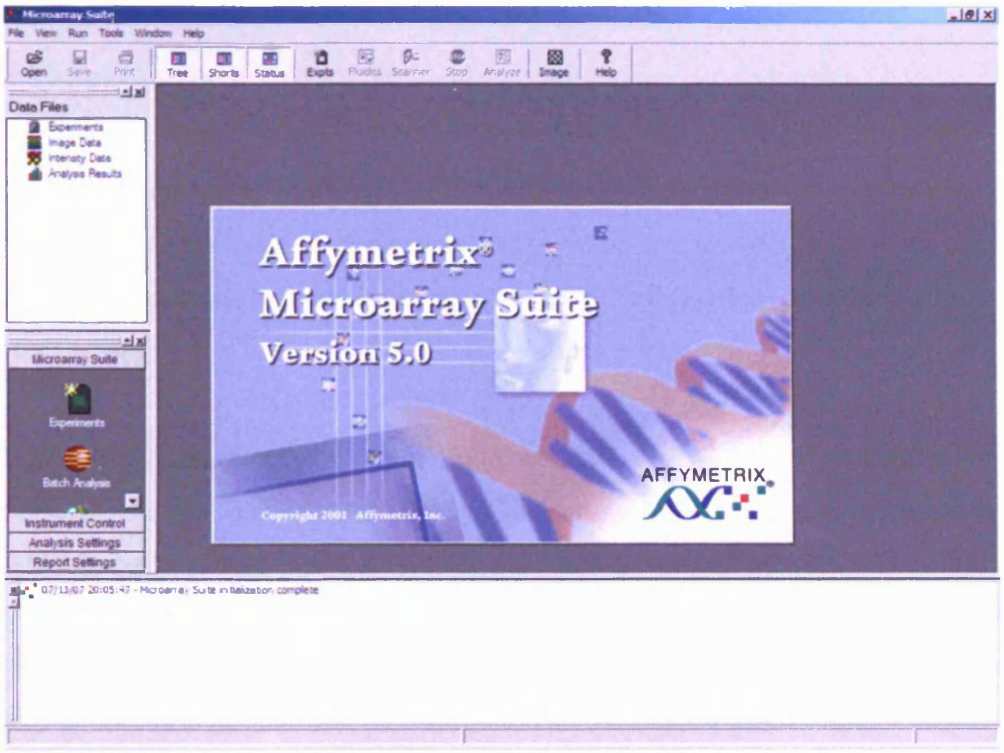


Figure 1.2: Opening screen of Affymetrix microarray suite version 5.0 (MAS5.0) (Affymetrix, 2001) [26]

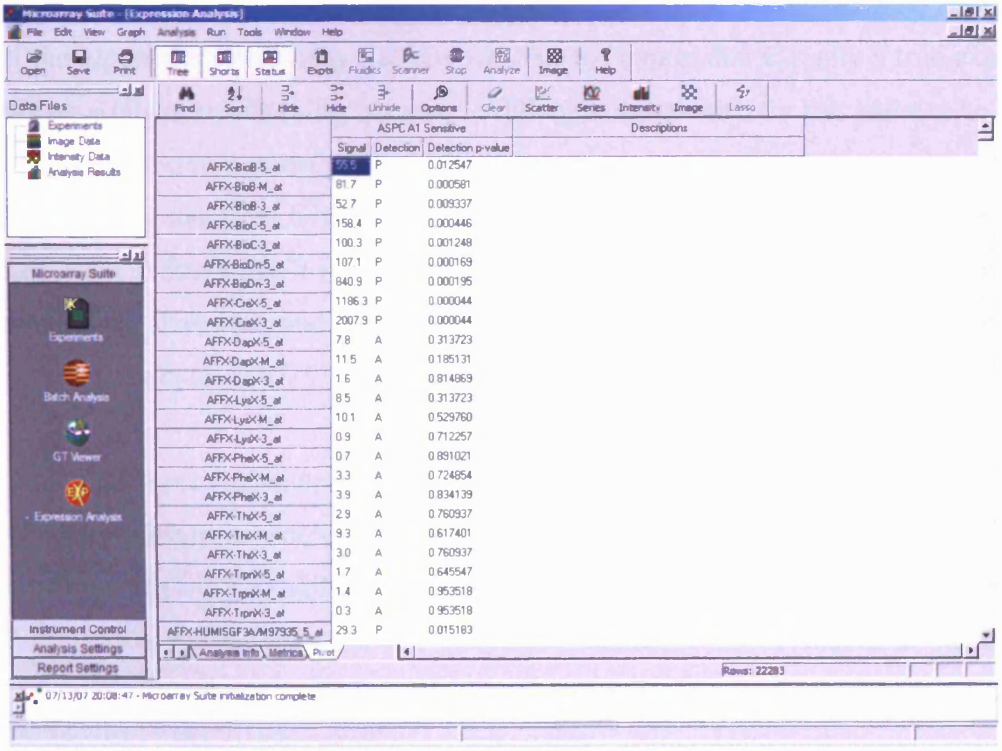


Figure 1.3: Screen capture showing the resulting output of MAS5.0 converting individual microarray chip results into individual probe level intensity values. (Affymetrix, 2001) [26]

1.3.4: PMA call

As shown previously in figure 1.3, it is clear from the figure that not only is an intensity measure for each probe is created, however also a p-value and uniquely a Present, Marginal and Absent (PMA) expression call produced as a result of information read from the scanner microarray 'CEL' files.

Each gene is represented a certain number of times ("probe set") resulting in expression values typically eleven times for each gene. The amount of light emitted represents whether there is RNA being expressed for a particular gene in a particular sample on the chip. For every correct gene sequence there is a mismatch, the same sequence with one change in the middle (25mer) (Robinson et al, 2007) [25]. The true expression value assigned depends on the mismatch expression for that gene and the expression across the test sequences. Essentially, the PMA call is a voting system as to whether a particular gene is likely to be really expressed in that sample and depends whether or not expression is above a certain level in the MAS5.0 application algorithm. If the algorithm statistically decides whether the expression is really a true expression level and not an artefact, it will call it 'present'. Although, the p-value for this call can be adjusted, PMA is usually related to a statistically significant p-value cut-off of 0.05. If slightly elevated, it falls into the category marginal however the exact way in which marginal is determined is unclear. It is important to stress that it is different to the flag scoring system of a *Cy3/Cy5* array which looks at spot shape, size, area, and other components which are issues which do not affect the Affymetrix system (Lee et al, 2007) [30].

PMA call from Affymetrix microarrays can be used as an initial filter before differential gene expression analysis. There are two main benefits of considering PMA call. Firstly, a degree of quality control is imposed by the software in terms of whether a particular result actually shows any real change on the chip and therefore whether or not it should be included for analysis. Secondly, filtering using PMA call is a quick and easy way of initially filtering the data, thereby eliminating subsequent excessive and uninformative statistical testing. However, in some instances use of PMA call may be undesirable as a feature for initial filtering during analysis of

data. It could exclude potential genetic targets of interest shown through reverse transcription polymerase chain reaction (RT-PCR) to have low expression levels which the Affymetrix process would otherwise exclude. This could be a consequence of the stringent thresholds which MAS5.0 assigns internally as part of the normalisation process for a particular probe which determines whether they are present or absent (versus the mismatch controls).

1.3.5: Data analysis - normalisation

Microarray chip normalisation can be a daunting process due to the multitude of procedures in which it can be achieved. As previously introduced, inherent to the Affymetrix system, to minimize mis- and cross-hybridisation problems, the technology includes perfect match (PM) and mismatch (MM) probe pairs as well as multiple probes per gene (Lim et al, 2007) [31]. Consequently to obtain a single signal intensity result for each probe, many calculations are required before an absolute expression level for a specific gene is produced. Such data pre-processing steps which combine multiple probe signals into a single value is known as normalisation. They usually involve three steps: (a) background adjustment, (b) normalization and (c) summarisation. In a recent review by Lim *et al*, four popular normalisation methods were compared – namely MAS5, RMA, GCRMA and Li–Wong (Lim et al, 2007) [31]. These are summarised in table 1.2.

Algorithm [Reference]	Background correction	Normalisation	Summarisation
MAS5.0 (Hubbell et al, 2002)[32]	Ideal MM subtraction	Constant	Tukey biweight
RMA (Irizarry et al, 2003) [33]	Signal and noise close-form transformation	Quantile	Median polish
GCRMA (Wu et al, 2004) [34]	Optical noise, probe affinity and MM adjustment	Quantile	Median polish
Li-Wong (Wong et al, 2001)[35]	None	Invariant set	Multiplicative model fitting

Table 1.2 – Summary of four popular normalisation procedures showing how each differs in the way background correction, normalisation and summary of an individual chip is calculated. MM=Mismatch (Lim et al, 2007) [31]

Depending upon which measure and comparison is made each method has its advantages. However, the RMA and Li-Wong methods tend to produce similar results, with MAS5.0 and

GCRMA exhibiting the largest difference overall in performance. However GCRMA has raised concerns when assessing correlation artefacts where it performs poorly in comparison to the other techniques available. This is particularly concerning if subsequent analysis methods which rely on an accurate measure of gene-pair expression profile correlation are to be used such as the clustering technique hierarchical clustering. It is thought the way in which GCRMA handles background correction is thought to affect its performance and therefore a flaw in the technique in comparison to RMA, for example. Furthermore, in relation to producing false positives, the GCRMA technique appears to introduce a high number in comparison to the MAS5.0 method which performs well in this regard. Consequently it could be argued that studies using the GCRMA could potentially have flawed results (Lim et al, 2007) [31].

Further normalisation of a different type also takes place when arrays are compared with each other to address a particular experimental hypothesis. Although transformation procedures are recommended to be kept to a minimum to preserve originality of the data distribution, log (base 2) transformation is performed initially. Also, for hierarchical clustering, gene median centering followed by sample median centering is also performed to align data before clustering (Eisen et al, 1998) [36].

1.3.6 Statistical testing: Differential gene expression

Following from normalisation (and any filtering based on PMA call), it is important to determine if there are any significant gene expression differences present within the data ("feature selection"). Robust identification of significant gene changes will allow subsequent pattern analysis to reveal potential signatures of interest. Once a feasible number of genes are generated, identification of individual genes known to play key biological roles in a resistance versus response environment, for example, could be identified within clusters and could potentially become therapeutic targets. Initial identification of significant differences can be assessed in a multitude of statistical ways. The motivation of such a step is to filter and discard genes which are unchanged between two different samples. The significant subsets of genes which remain are taken further for subsequent detailed pattern discovery.

Since microarray analysis became routine, assessing which genes are differentially expressed has been performed in a number of ways. Detection of differences between two groups can be determined using the well established and widely used parametric t-test, analysis of variance (ANOVA) and the well known nonparametric Wilcoxon rank sum statistical tests (Thomas et al, 2001) [37]. ANOVA is often used in microarray analysis due to the ability of coping more than two independent groups. However due to the nature of microarray experiments and the data generated, caution should be used when applying such classical statistical tests. There are three main reasons for caution.

Firstly, the t-test assumes a normal distribution of the data and a constant variance for all genes across all samples compared using the microarray. Given the very nature of what is trying to be achieved with a microarray experiment, such assumptions are inappropriate for a subset of genes despite any given transformation (Thomas et al, 2001) [37]. Secondly, the tests are not able to take advantage of the genomic data when correcting for heterogeneity between samples (Thomas et al, 2001) [37]. Finally, as a result of multiple comparisons testing, a phenomenon called the 'false discovery rate (FDR)' must be taken into account for accurate results. For example, if a typical p-value threshold of 0.05 was used to determine differential expression for individual genes between two groups, there would be 50 false positives for every 1000 genes under examination, even though none of these genes are differentially expressed in reality (Thomas et al, 2001) [37]. However more recent approaches tailored specifically to the nature of microarray data have been developed – namely significant analysis of microarray (SAM).

SAM is a statistical technique for finding significant genes within a set of microarray experiments, as proposed by Tusher, Tibshirani and Chu (Tusher et al, 2001) [38]. The input to SAM is gene expression measurements from a set of microarray experiments and their parallel grouping (for example anti-hormone response or resistance; treatment groups). It has the ability to be able to analyse a multiclass grouping (e.g: breast cancer: different treatment arms). SAM computes a statistic (*di*) for each gene (*i*), measuring the strength of the relationship between gene expression and the response variable (Tusher et al, 2001) [38]. It uses repeated permutations of the data to determine if the expression of any genes is significantly related to the response grouping. The point chosen for significance is determined by a tuning parameter delta, chosen by

the user based on the false discovery rate (FDR). The system can be optimised resulting in only 10% of those genes revealed potentially being false positives which is an accepted level of confidence of the data.

There is a compromise between what needs to be included and how well a particular algorithm can perform statistically. Indeed, traditional statistical analysis methods were not designed for analysing large bodies of microarray data (20,000 genes, multiple replicates, and multiple treatment arms). A typical example is pulling out significant genes using t-tests or using ANOVA where the simple p-value is not representative of the true false discovery rate: the more genes there are to analyse, the greater number of statistical tests need to be applied and hence the greater chance of false positives. Existing software such as Genesifter or more recently Array2BIO can be used to address the false positive discovery rate (FDR) alongside such statistical testing, through performing a post hoc test using either Bonferonni or Benjamini-Hoechberg tests (Loots et al, 2006) [39]. However, with some datasets these approaches can be very ruthless (Reimers et al, 2005) [40]. Considering the FDR issue and addressing this aggressively is clearly potentially problematic in that fewer gene hits are obtained, with the potential of key genes being lost in the FDR correction calculations. However if FDR is not considered, inaccurate predictions maybe made, and this could necessitate potentially excessive subsequent PCR verification. Such issues become particularly relevant as the dataset involves more complex multiple arm comparisons in terms of treatments and disease progression. For example, the power of the panel of MCF-7-derived resistant and responsive model systems studied by the Tenovus group is in being able to use them as a complex comparative set. Clearly it is important how array technology is utilised and care taken during analysis to prevent loss of valuable information at each stage. Therefore adoption of a refined and optimised array strategy is essential. Implementation of significant analysis of microarrays aims to therefore to better achieve a balance between overestimation and significance (Tusher et al, 2001) [38].

1.3.7 Class discovery

Following assessment of which genes are differentially expressed, molecular profiling of breast cancers by gene expression microarrays to discover dominant patterns of expression can be

performed by either unsupervised or supervised analysis. Unsupervised analysis refers to methods such as hierarchical clustering analysis for partitioning samples into groups or classes on the basis of gene expression profiles, regardless of other features (Quackenbush et al, 2001) [41]. In this approach, the goal is to determine whether discrete subsets can be defined on the basis of gene expression profiles and to identify new classes (class discovery) that may ultimately have clinical significance and therefore to develop a new molecular classification. Supervised analysis, in the breast cancer scenario, requires samples to be allocated to specific groups based on clinical or pathological features in the case of patient material, or for example response versus resistant models *in vitro*. There are two main subtypes of supervised analysis: class comparison and class prediction which aims to explain the relationships present within the data set. Array analysis aims to identify transcriptomic differences between classes of samples, which differs to supervised analysis where a genetic signature (according to the feature groupings) can potentially be achieved (Reis-Filho et al, 2006) [42].

From a microarray perspective, class discovery within a dataset is closely related to differential gene expression – both are synergistic. Thus, once a set of differentially expressed genes has been revealed, it is possible to generate individual gene profiles, or to identify which genes behave similarly or differently across the samples which is the class discovery element. Identified close expression associations between individual genes (clusters) will highlight genes which are trasncriptionally co-regulated and could potentially form a "pathway unit", overlapping via some of the cluster elements with known pathways or in other instances forming novel networks. Although there are many types of clustering techniques, there are five main approaches which differ from each other which can be used to generate robust clusters (and thus prioritise potential targets) within a dataset.

There are many different ways in which clustering can be performed in microarray analysis. In this section four clustering techniques will be outlined which includes a relatively new approach – 'Fuzzy' clustering to be applied to microarray data.

1.3.7.1 Hierarchical clustering

Hierarchical clustering (HCA) is one of the most widely used forms of clustering which usually uses Euclidean distance to identify associates between individual genes which is represented in a parent/child tree representation (Kaufman et al, 1990) [43]. Many landmark studies have been performed purely dependent upon HCA results, in regard to breast cancer signatures. In general, its reliability varies when used for large gene sets, largely due to the time needed to calculate distances between all the genes. Comparison with other clustering techniques is important to check its performance with a particular dataset. Consequently, HCA is usually performed after all genes which show no significant change have been removed by significant gene analysis processes such as SAM. Often results of HCA are visualised using a heatmap, an example of which is shown in Figure 1.4.

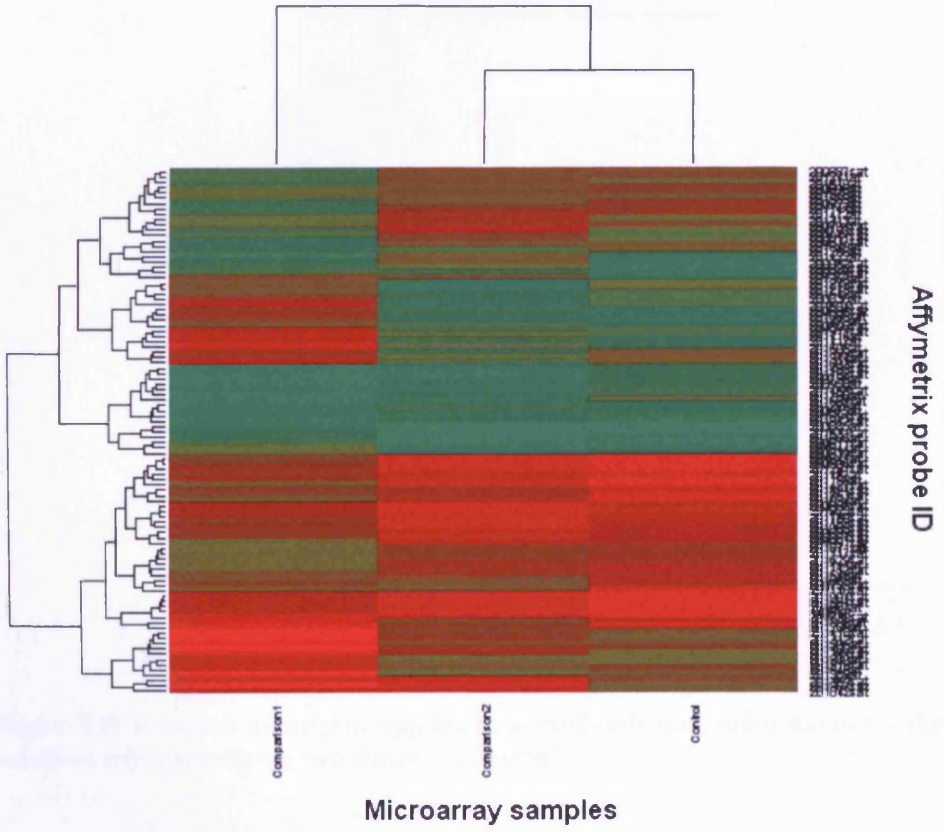


Figure 1.4: Example of a heat map plotted using a hierarchical clustering algorithm showing clustering for sample (columns) and probe sets (rows).

1.3.7.2 K- Means

K-Means is a partition clustering technique. Such clustering differs in that the user specifies the number of clusters to divide the genes into, with the computer randomly in the first instance but then assigns genes to clusters according to a particular similarity measurement with the cluster centroid (Hartigan et al, 1979) [44]. To determine the number of clusters, hierarchical clustering is usually performed before K-Means analysis. K-means is an iterative method which minimizes the within-class sum of squares for a given number of clusters. The algorithm starts with an initial estimate for the cluster centres, and each gene observation is placed in the cluster to which it is closest as measured by Euclidean distance. The cluster centres are then updated, and the entire process is repeated until the cluster centres no longer move (Du et al, 2008) [45]. An example of how K-means can be visualised is shown in Figure 1.5. A report of which clusters particular genes have been assigned can be obtained from the clustering algorithm.

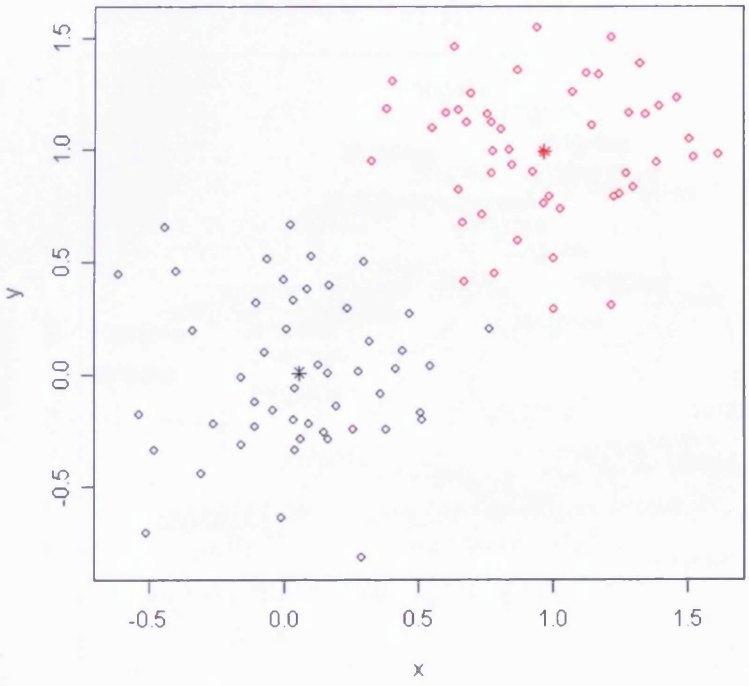


Figure 1.5: K-means algorithm applied to a randomly generated dataset – the * indicates centroid locations when specifying two clusters using 'R'.

1.3.7.3 Partitioning Around Medoids (PAM)

Partitioning around medoids is similar to K-means, however is considered more robust because it admits the use of other dissimilarities besides Euclidean distance (Bozinov et al, 2002) [46]. An example is shown in figure 1.6. Partitioning around medoids was developed by Kaufman and Rousseeuw in 1990. For a specified number of clusters K, the PAM procedure is based on the search for K representative objects, or medoids, among the observations to be clustered. After finding a set of K medoids, K clusters are constructed by assigning each observation to the nearest medoid (Bozinov et al, 2002). The goal is to find K medoids, M, which minimise the sum of the distances of the observations to their closest medoid. Rousseeuw suggested a graphical display, the silhouette plot, which can be used to select the number of clusters and assess how well individual observations are clustered. Intuitively, objects with large silhouette width are well clustered; those with a small width tend to lie between clusters.

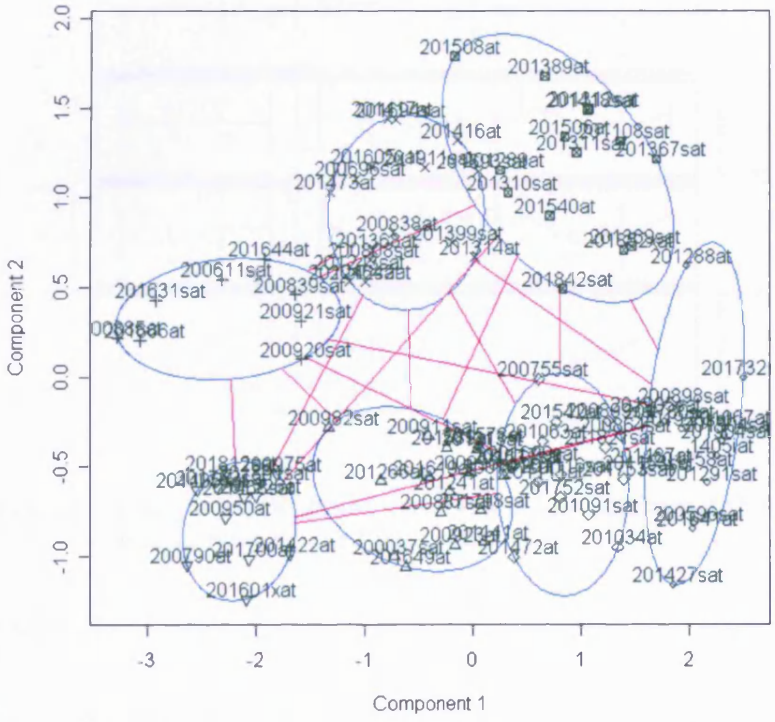


Figure 1.6: An example of PAM using a dataset of 100 Affymetrix probes split into seven clusters.

1.3.7.4 Self Organising Maps (SOM)

Self-organising maps is an unsupervised learning technique (Kohonen et al, 2000) [47]. The model was first described as an artificial neural network by the Finnish professor Teuvo Kohonen, and is sometimes called a Kohonen map (Kohonen et al, 2000) [47]. SOM performs well for in its ability to map and visualize multi-dimensional data in two dimensions. It is of particular power to reveal common profiles of down or up regulation in microarray data sets. This makes SOM useful for visualizing low-dimensional views of high-dimensional data, akin to multidimensional scaling. Typical results of SOM perfomed in the statistical programming environment 'R' can be seen in figure 1.7

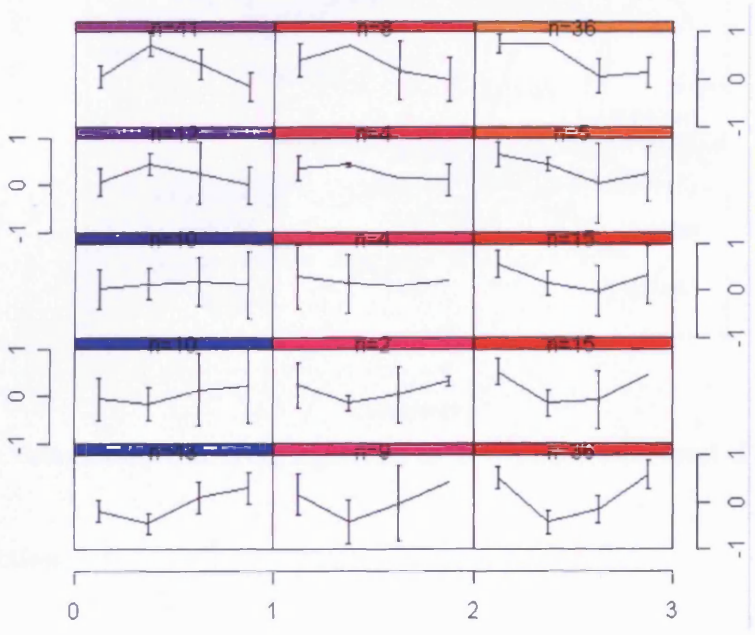


Figure 1.7: Self organising maps using 'R' based on an example yeast dataset showing 15 different distinct profiles and associated number within each profile

1.3.7.5 Fanny (Fuzzy Clustering)

Fuzzy clustering is performed by the Fanny algorithm in the statistical programming language 'R', where each observation can have partial membership in each cluster (Kaufman et al, 1990) [43]. Consequently, each observation has a vector which gives the partial membership to each of the clusters. A so called 'hard' cluster can be produced by assigning each observation to the

cluster where it has the highest membership. The technique was originally introduced by Jim Bezdek in the early 1980's who strived to improve clustering techniques in general. Fuzzy clustering provides a method that shows how to group data points that populate some multidimensional space into a specific number of different clusters (Hathaway et al, 2001) [48]. An example of this is shown in figure 1.8.

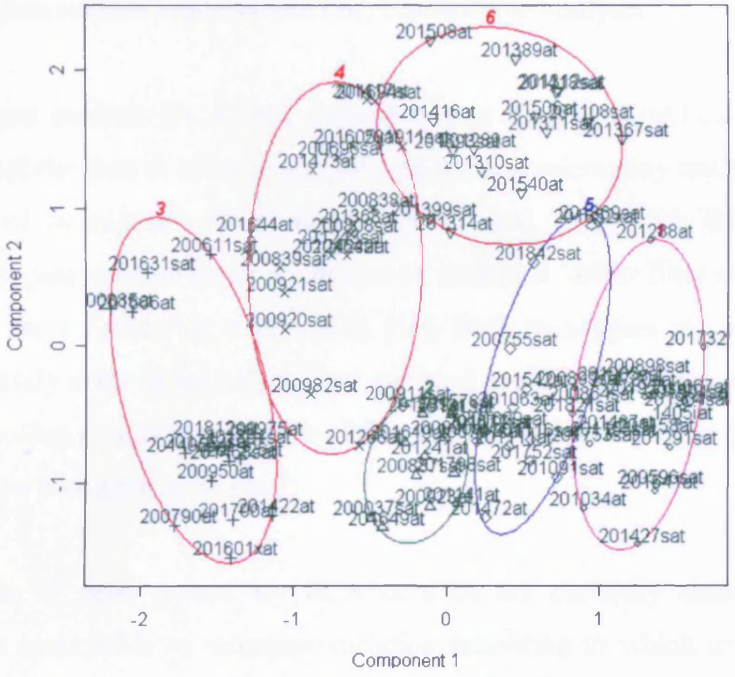


Figure 1.8: Using Fuzzy clustering algorithm in 'R' with a cluster level of 7 clusters

1.3.8: Class prediction

Class prediction differs from class discovery in that it places new samples into previously defined classes. This differs to class discovery which is focused more upon discovering new groups within a dataset and dividing the samples accordingly (Rogers et al, 2005) [49]. Class prediction can either be performed ontologically using online tools which will map a particular uploaded list of genes into their respective ontological categories, or it can be performed using more advanced statistical approaches. Class prediction is well suited to cancer biology. Classifiers can be built which may reliably indicate subtype or expected progression of a particular cancer, for example. This ultimately would have important clinical consequences. For example, in breast cancer,

tumours can range from relatively inactive to aggressive, microarray technology could be used to predict where, based on such a range, a particular sample resides. Therefore such a classifier could be used to avoid unnecessary treatment which itself could have greater consequences for the patient (Rogers et al, 2005) [49]. Class prediction using computer predictive models, is described in detail in Chapter 5 due to the significance it can potentially offer to cancer research.

1.3.8.1 Principal components analysis and correspondence analysis

Principal components analysis (PCA) and correspondents analysis (CoA) can be used to reduce the dimensionality of the data in order to simplify subsequent microarray analysis steps and allow for summarization of the data in a visual manner (Wang et al, 2005) [50]. This approach is more effective on smaller gene sets; however it can also be used as a further filter after significant gene analysis generated from SAM (Yu et al, 2008) [51]. Both techniques can be used to visualise cohorts of significantly differentially expressed genes. A typical initial step after SAM could be independently revealing groups visualised by PCA and then using these groups for classification which will produce a minimum gene list.

Selections in terms of gene groups for PCA or CoA are currently made by the user and consequently more susceptible to selection variation according to which user was making the choices – the human error factor. However, if such a selection could be performed intelligently by the computer, the chances of variation would be lower as the computer would be more consistent in terms of the rules it applies to a particular decision, due to the absence of human intervention being required. Adopting approaches where the number of groupings in the data is chosen automatically and confirmed with statistics will be desirable in reinforcing results obtained using high throughput analysis.

1.3.8.2 Multidimensional scaling

Multidimensional scaling (MDS) is a set of data analysis techniques that visualises the structure of distance matrices as a geometrical picture. The technique is similar to principal components analysis. MDS pictures the structure of a set of objects from data that approximate the distances

between pairs of the objects which could correspond to difference genes from a microarray experiment. Each object or event is represented by a point in a multidimensional space (Rencher et al, 2003) [52]. The points are arranged in this space so that the distances between pairs of points have the strongest possible relation to the similarities among the pairs of objects – such as genes which could have an underlying functional relationship (Rencher et al, 2003) [52]. Two similar objects are represented by two points that are close together, and two dissimilar objects are represented by two points that are far apart. The space can either be two or three-dimensional Euclidean space, although other distance matrices can be used. MDS is a generic term that includes many different specific types. These types can be classified according to whether the similarities data are qualitative (called non-metric MDS) or quantitative (metric MDS) (Rencher et al, 2003) [52].

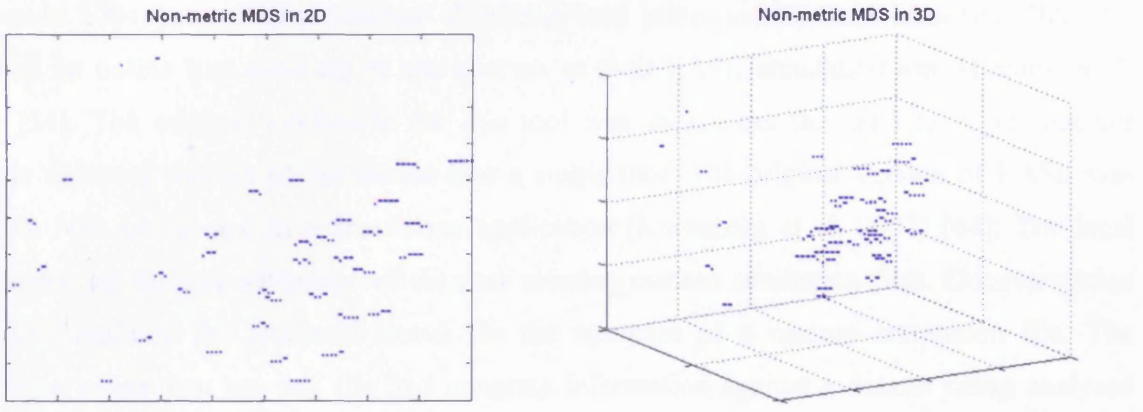


Figure 1.9: An example of how a non-metric Multidimensional scaling example can be plotted in two and three dimensions (Mathworks Inc, 1994-2009) [53].

1.3.9: Class prediction using ontological resources

Ontology is associated with both class prediction and class discovery as initially both processes are driven by candidate gene lists in respect to microarray data. For example, when clustering is performed, it may be of interest to assess where known land mark genes are located and which other genes cluster closely to these known genes. As unknown genes fall into categories close to those which are known, similar ontology could potentially be applied in databases for these

previously undefined genes. Annotation steps have incorporated a mine field of online databases for many years. Typical non-commercial online resources include DAVID (Dennis et al, 2003) [54], FATIGO (Al-Shahrour et al, 2007) [55], NetAffy (Oeder et al, 2007) [56], Biocarta (Mlecnik et al, 2005) [57], Ensembl (Joshua et al, 2008) [58], GenMAPP and MAPPfinder (Dhalquist et al, 2004) [59], Pubmed (NCBI, 1988) [60], Genecard (Weizmann Institute of Science, 2008) [61] and Chilibot (Chen et al, 2004) [62]. In Addition, packages such as Genesifter, and also DMT from Affymetrix, are available with some ontological capabilities (Wang et al, 2006) [63]. Full ontological searching requires the user to visit and integrate data from all of these sites.

1.3.9.1 Database, Annotation, Visualisation and Integrated Discovery (DAVID) resource

In October 2004, the National Institute of Allergy and Infectious Diseases launched (DAVID) launched an online tool building on the success of their EASE annotation tool (Dennis et al, 2003) [54]. The original motivation for this tool was to address the need to bring together multiple database sources of annotation into a single tool. The original version of EASE was available both online and as a standalone application (Korenberg et al, 2007) [64]. The local application had the key advantage of the user creating custom annotation files. Documentation for EASE outlined the accepted format for the structure of a custom annotation file. The application could then use this file and integrate information against a dataset being analysed along with other ontological information already stored in the application. However one of the key problems was the difficulty in creating such files.

Recent versions of the application have moved to an online version of the application. The functional annotation tool, which can be interrogated according to Affymetrix Probe ID, has the benefits of frequent updates whenever the individual databases are changed or new databases are added. This marked a step forward in reliability as new text files were required to be downloaded when updates became available for the original application – EASE. The new online annotation tool offered together with other tools through DAVID, addresses a key limitation of EASE by introducing a new graphical interface complete with graphical pathway information. Typical screen captures showing screen layout of DAVID is shown in figure 1.10.

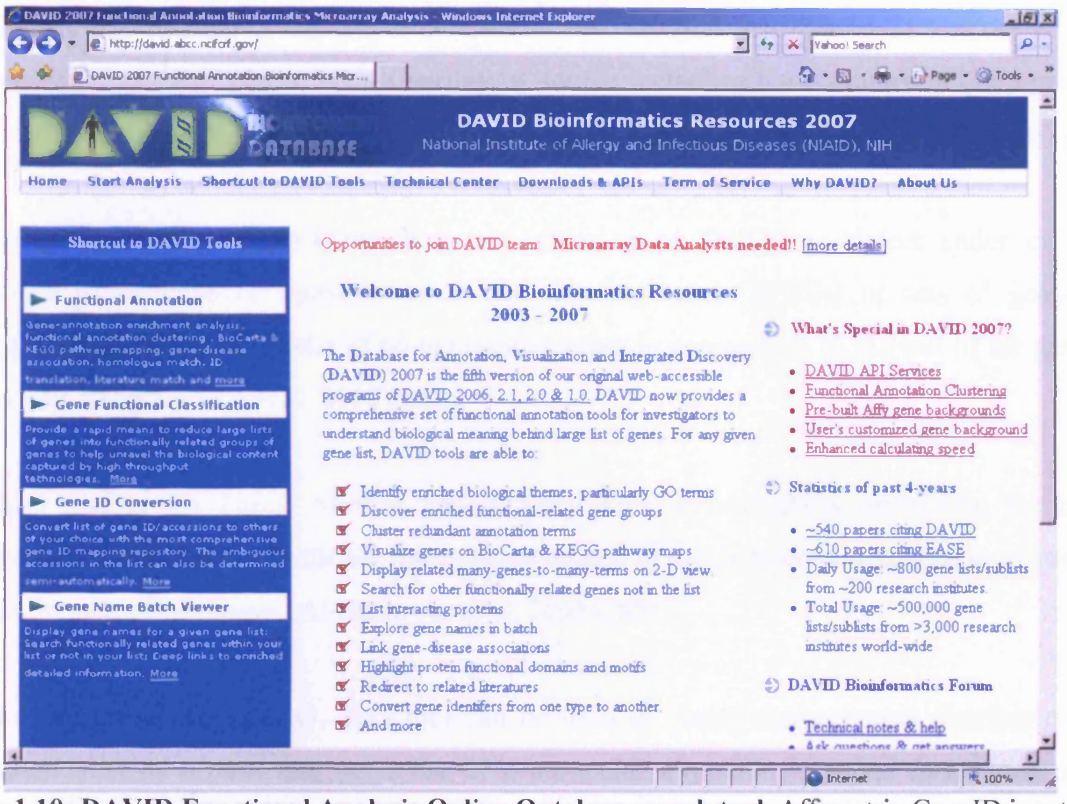


Figure 1.10: DAVID Functional Analysis Online Ontology search tool. Affymetrix GeneID input can be observed to the left of the figure.

1.3.9.2 Babelomics

Babelomics is a suite of web tools for functional annotation and analysis of groups of genes from high-throughput technologies, such as microarray analysis. Named after the tale outlined in the book, 'The Babel library', by the famous Argentinean writer Jorge Luı´s Borges, which outlines that finding a real book among a pile of meaningless texts is an excellent metaphor for the challenge that constitutes the extraction of information from the masses of data which exists in the postgenomic era (Al-Shahrour et al, 2008) [65]. What is real and what is simply an association by chance is a very significant issue when classifying genomic data.

Babelomics was developed in Madrid, Spain at the Centro Nacional de Investigaciones Oncológicas (CNIO). It provides five different tools upon which uploaded gene list analyses can be performed.

The first tool is FatiGO+, which can be used to test unequal distribution of functional terms between two groups of genes from a variety of source including Go ontology, KEGG pathway information (Frohlich et al, 2008) [66] and Swiss-Prot information (Bairoch et al, 1998) [67].

The second tool is TransFAT which is an extension of FatiGO to detect under or over-representation of putative transcription factors binding sites (TFBSs) in sets of genes, by comparing them against a cluster of co-expressing genes in comparison to the rest of the genes in the analysis (Al-Shahrour et al, 2008)[65].

The third tool is the Tissue Mining Tool (TMT). This is particularly interesting for cancer biology in that it extracts significant information related to the differential expression of two sets of genes in particular tissues (Al-Shahrour et al, 2008) [65].

The fourth tool is GenomeGO. This tool can be used to assess chromosomal distribution at a functional level of a particular gene list. If a particular alteration is found in a chromosome region, the functions of the genes therein can be studied (Al-Shahrour et al, 2008) [65].

The final tool is FatScan which allows study of correspondences between phenotypes and molecular roles of genes by analysing ordered lists of genes supplied to the tool. Due to the wealth of information available from different high throughput technologies, this tool can be used to obtain lists of genes ordered according to their different behaviours under different experimental conditions corresponding to different phenotypes (Al-Shahrour et al, 2008) [65]. This tool could be useful where low numbers of highly significant genes were generated from a particular analysis.

Figure 1.11 displays the entrance page to the collection of tools which enable navigation to the Babelomics online software suite.

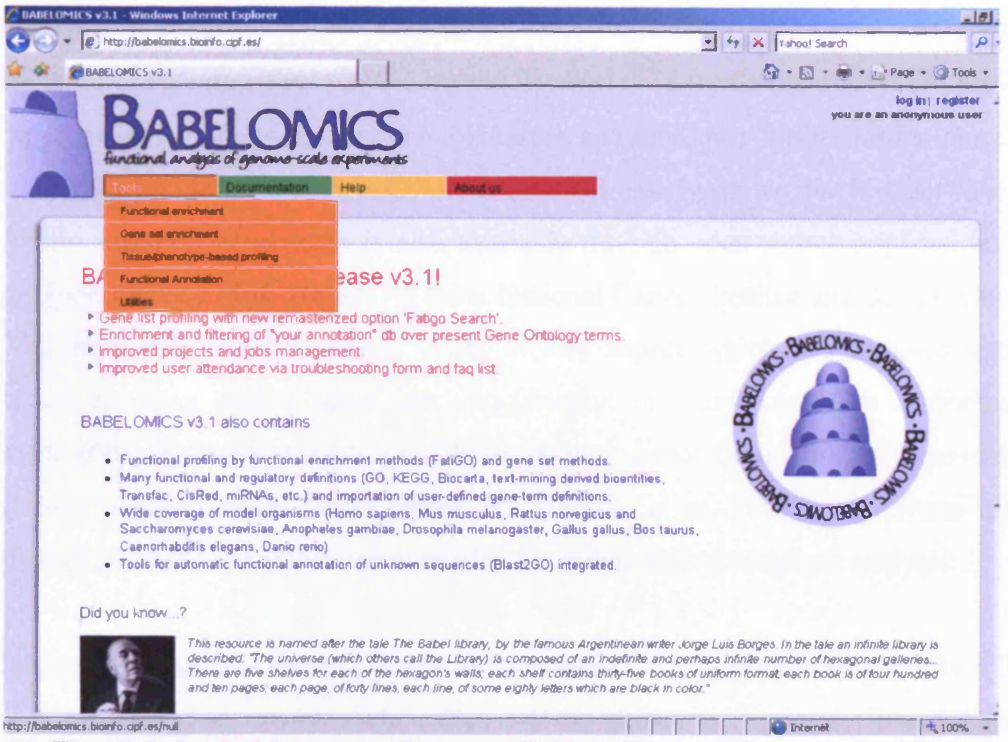


Figure 1.11: The 'Babelomics' homepage listing available online tools for gene classification (Al-Shahrour et al, 2008) [65].

1.3.10. The limitations of existing microarray analysis software – a brief review

Given the different array platforms and biomedical analysis technologies which exist, many different software packages are available to the cancer research community for data analysis. It is therefore useful to review what established applications are widely used. Although there are many commercial packages available the emphasis and pricing structure of such packages has historically been targeted at major pharmaceutical companies and is beyond the scope of most academic research groups. Consequently the review focused on a mixture of applications currently used by Tenovus and also alternatives developed within the research community.

Table 1.3 highlights a summary of how each software application reviewed compares. It should be remembered that Affymetrix MAS5.0 differs to the other application outlined in Table 1.3 in that is primarily used to normalise raw data from Affymetrix scanner to allow analysis.

The commercial microarray analysis packages GeneSifter™ from VizXLabs® (Vizxlabs, 1997-2008) [68] and Data Mining Tool (DMT) from Affymetrix® (Affymetrix, 2008) [69] provide user friendly approaches for analysis, there are cost issues and moreover further limitations including statistical analysis functionality. The software review has clearly shown that broader functionality routine within commercial systems is also available through open source applications such as BRBArrayTools (Simon et al, 2008) [70] from National Cancer Institue and dChip (Cheng et al, 2003) [71], from Harvard University, USA. A key aspect which has allowed such broad flexibility on an open source basis can undoubtedly, in many cases, be attributed to the development of the Bioconductor analysis library (Bioconductor Core, 2002) [72] written using a statistical programming environment called 'R' (Dessau et al, 2008) [73]. These algorithms confer advanced statistical capabilities as needed for robust high-throughput analysis.

The BRBArrayTools suite is one of the few applications exploiting the linking ability of Excel to a statistical programming environment called 'R' as shown in figure 1.12 and figure 1.13. However, as summarised in table 1.3, all the tools perform general array analysis aspects relatively effectively, however some aspects of implementation are inconsistent. It was clear from the review that there was a need for researchers who are aiming to reveal new targets and markers to have access to software with additional graphical functionality including more advanced pattern analysis methods than seen in BRBArrayTools. However, seamless implementation of analysis tools into Excel as seen in BRBArrayTools is a desirable quality when considering ease of use by the researcher. Overall, it is clear from the evaluation that there remains a need to improve upon the software currently offered to facilitate prognostic marker and new target discovery, as applied to *in vitro* experimental material and also potentially applicable to clinical datasets.

Application	Functionality	Advantages	Disadvantages
Affymetrix® Microarray Analysis Suite 5.0 (MAS5.0) (Affymetrix, 2001-2008) [26]	- Raw data processing - Fold change estimates	- One click ease of use	- 'Black box' methodologies unclear - Costly - Poor statistical capability
Affymetrix® Data Mining Tool (DMT2.0) (Affymetrix, 2001-2008) [69]	- t-test - Kruskal-Wallis - Wilcoxon	- Familiar design to MAS5.0 - Links to NetAffy for ontology	- Signal:Noise issues - Very slow - No advanced statistical capability - Costly
Genesifter™ (Vizxlabs, 1997-2008)[68]	- Raw data processing - t-test, ANOVA - PAM & K-Means Wilcoxon test - HCA	- Relatively easy to use - Basic ontological capability	- Limited clustering - Expensive (same capabilities in open source software) - No advanced statistical capability
BRB Array Tools (Simon et al, 2008) [70]	- Raw data processing - t-test - SAM - HCA, MDS clustering - Basic ontology	- Free to academic community - Good interface - Excel add-in - Quick - Uses Bioconductor R-module library	- Glitches during operation - Inconsistent results - Limited pattern analysis
Bioconductor (Bioconductor Core, 2002) [72]	- 'R' scripting basis - Comprehensive array analysis strategies available	- Cutting edge statistics - Fast - Extremely versatile - Free	- Difficult to command for inexperienced computer users.
dChip (Cheng et al, 2003) [71]	- Alternative to MAS5.0	- Improved Signal:Noise ratio handling of Affy data	- Limited in functionality - Lack of advanced statistical methods

Table 1.3: Comparisons of popular microarray software – ranging from raw image processing and analysis

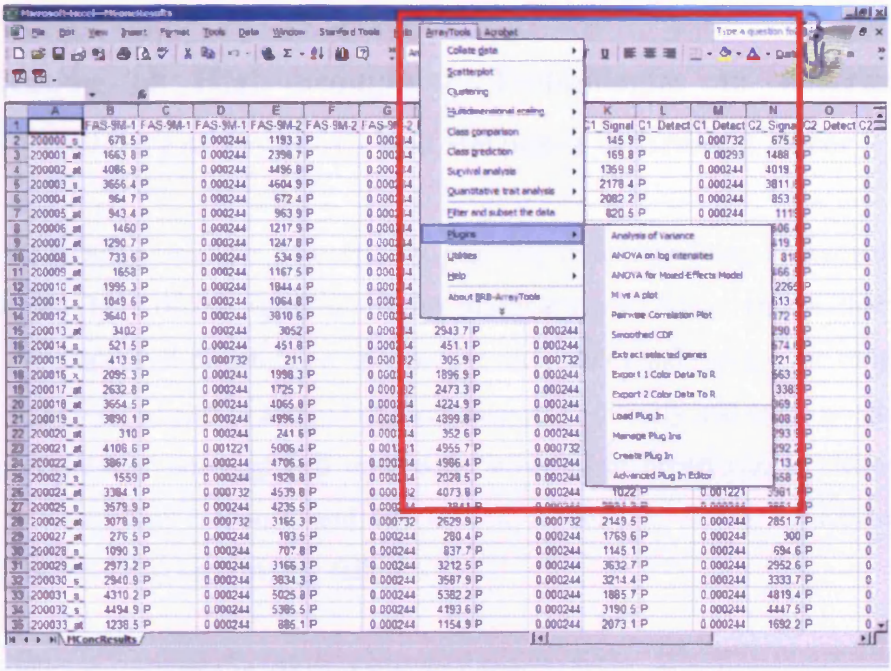


Figure 1.12: BRB Array tool screen shot showing MAS5.0 output in Microsoft Excel format. Red box highlights the analysis choices which the application offers within Excel.

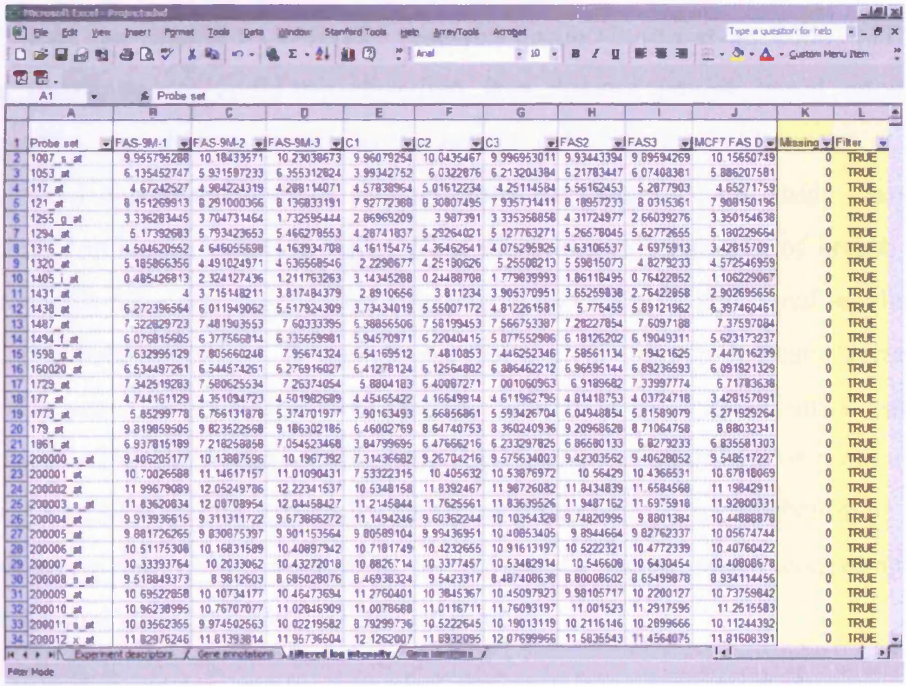


Figure 1.13: BRB Array tool screen shot showing results of transformed MAS5.0 input files after performance of normalisation between the selected groups.

1.4 Application of High-throughput Technologies to determine Gene Signatures Predictive of Prognosis and Response in Breast Cancer

Microarrays are proving increasingly useful for gene expression profiling in many disease states, especially cancer. Two of the applications for microarrays in breast cancer that are proving important in the context of determining new prognostic indices and therapeutic targets are: (i) to use class discovery techniques in arrayed clinical datasets to classify and make predictions, which are revealing previously unrecognised prognostic subtypes in breast cancer extending to their relationship with clinical response and (ii) to cell models, to further understand treatment responses and to reveal factors driving failure.

1.4.1 Class discovery applied to clinical breast cancer

The power of the class discovery approach for breast cancer research was first demonstrated by Perou *et al.* from the Stanford University research group (Perou et al, 2000) [11]. The study showed that breast cancer could be classified into distinct groups based upon their gene expression profiles and their similarity to the normal cell equivalents (Perou et al, 2000) [11]. Using hierarchical clustering methods and an 'intrinsic gene set', the study classified breast cancer into four 'molecular' classes. Apart from the intuitive separation of breast cancers into ooestrogen receptor *ER+* and *ER-* disease (the two main clusters), additional smaller secondary clusters had also been identified. It was shown that the *ER+* group is characterized by higher expression of a panel of genes that are typically expressed by breast luminal epithelial cells ('luminal' cancer). The ER negative group encompassed three subgroups of tumours: one over-expressing *erbB2* (*HER2*); one expressing genes characteristic of basal phenotype and another with a gene expression profile similar to normal breast tissue which consistently clustered together with normal breast samples and fibroadenomas.

Subsequent array studies from this group and others have been able to reproduce the key molecularly-defined groups revealed by Perou *et al* (Perou et al, 2000) [11], although there is some variation with select subsets. Thus, the sub classification of luminal tumours has ranged

from a single group to up to three (A-C) groups. Also, 'normal breast-like' cancers have appeared similar to the *ER-* cluster in most (Sørlie et al, 2001) [74] but not all studies, such as that from Caza *et al* (Calza et al, 2006) [75]. Reverse transcription-polymerase chain reaction (RT-PCR) has been invaluable in verification of selected genes (Mullins et al, 2007) [76] and also immunohistochemistry of tissue sections to assess protein biomarker expression (Makretsov et al, 2004) [77]. These have been able to verify the key luminal, *HER2* and Basal molecular classification previously determined by array studies. A brief overview of key examples is outlined in table 1.4.

The significance of the molecular 'taxonomy' as discussed by Rakha *et al* (Rakha et al, 2008) [3] can be divided into two parts:

(i) The clinical behaviour of each molecular grouping differ, even though the classification system was not developed to predict outcomes

(ii) Genes revealed, or their protein products, could potentially be developed as therapeutic targets as well as diagnostic tools.

Although studies have demonstrated that the oestrogen receptor, oestrogen receptor-related genes and *HER2* are important biological drivers for some of the individual subclasses that they define, the difference between these classes could not be based on single genes or a specific pathway, but on a bank of several groups of genes forming a signature of each class. A study by Charafe-Jauffret *et al.* (Charafe-Jauffret et al, 2006) [78] identified a set of 1233 genes that differentially expressed between basal-like and luminal samples. As such to date, no single gene can be used to identify these classes reliably. Although ER expression is a key factor in these classifications, both *ER+* and *ER-* samples display heterogeneous expression profiles, with the identification of at least two or three subgroups in each category with different behaviour and outcome. These novel molecular subtypes can thus be thought of as being defined by expression of a collection of genes and their associated pathways. A lot of study remains in evaluation of potential new therapeutic targets from within the multi-gene signatures of these sub-types, an avenue which remains important since not surprisingly, not all *ER+* tumours or *HER2+* tumours respond to anti-hormones or Herceptin respectively.

Study	No. Tumours	Gene set (total)	No. of classes	Name, no (%) of different molecular classes
cDNA microarrays				
Perou et al, 2007 [11]	65	496 (8102)	4	Luminal, 36(58) *HER2*, 7 (11) Basal-like, 8 (13) Normal, 11 (18)
Sorlie et al, 2001 [74]	78	456 (8102)	6	Luminal A, 32 (58) Luminal B, 5 (6) Luminal C, 10 (12) *HER2*, 11 (13) Basal-like, 14 (16) Normal, 13 (15)
Sorlie et al, 2003 [79]	115	534 (8102)	5	Luminal A, 28 (36) Luminal B, 11 (14) *HER2*, 11 (14) Basal-like, 19 (24) Normal, 9 (12) (37 unclassified)
Sotirou et al., 2003 [80]				Luminal (classes1-3) etc
RT-PCR				
Perreard et al. 2006 [81]	117	53 genes	4	Luminal *HER2* Basal-like Normal Breast-like
Chanrion et al, 2007 [82]	199	47 genes	12	12 subgroups corresponding to the Luminal A/B, Normal Breast-like, *HER2* and Basal like subsets
Mullins et al, 2007 [76]	124	40 genes	4	Luminal *HER2* Basal-like tumour Normal Breast-like
Immunohistochemistry				
El-Rehim et al, 2005 [83]	1076	25 proteins	6	Luminal-1, 336 (31%) Luminal-2, 180 (17%) *HER2*, 234 (22%) Group 4, 4 (0.4%) Basal-like, 183 (17%) Luminal-3, 139 (13%)
Korsching et al, 2002 [84]	166	15 proteins	3	*HER2* over expressing Basal like (CK5/6)a *ER/PgR+*
Diallo-Danebrock et al, 2007 [85]	236	34 proteins	5	Using 24 of the 34: Luminal A, 61 (27%) Luminal B, 28 (12%) *HER2*, 48 (21%) Basal-like, 29 (13%) 'Multi marker –ve', 63 (27%) characterised by absence of specifying markers.

Table 1.4 – Summary of molecular classes verified using different techniques as summarised by Rahka *et al* (Rakha et al, 2008) [3]

Most of the discriminator genes appear to be involved in cell cycle regulation, cell signalling, cell proliferation, hormone receptors and oncogenic pathways. Table 1.5 shows frequency and some of the genes associated with the different molecular classes of breast cancer described to date, where some may have targeting potential or for example, *EGFR*, a feature of a proportion of basal cancers, is actively being explored as a therapeutic target in the clinic, where treatments for this group are much-needed since tailored therapies are lacking (Feng et al, 2007) [9].

As discovery of the genes and pathways associated with classes of breast cancer continues, it is likely that further clinical sub stratification will occur. For example, the existence of a molecular 'apocrine' breast cancer subtype with increased androgen signalling and frequent *HER2* amplification has been reported (Farmer et al, 2005) [86], while another classification based on the gene signatures of *RAS* and other deregulated pathways has also been described (Bild et al, 2006) [87].

Breast cancer molecular class	**% clinical disease**	**Gene features that could theoretically contribute to prognostic behaviour**
Oestrogen receptor +ve Luminal A Tumours	19-39%	Highest expression of *ER* and *ER* related genes representing best prognosis
Oestrogen receptor +ve Luminal B Tumours	10-23%	Compared to luminal A may have a higher proliferation rate, certain genes shared with basal-like and *HER2* subtypes – less favourable outcome.
Oestrogen receptor –ve Basal like	16-37%	Genes previously identified to be characteristic of basal cells such as *CK5*, *integrin 4*, *EGFR*, *NF-kB* – includes patients with *BRCA1* mutations, poor prognosis and lack of response to hormonal therapy.
HER2+	4-10%	High levels of genes on *HER2* amplicon (17q11) including HER2, *GRB7*, *GATA4*, high levels p53 mutation, poor prognosis and lack of response to hormonal therapy
Normal Breast-like	>10%	High expression of basal epithelium genes, low expression of luminal epothelial genes. Better prognosis than most basal tumours. No responsive to neoadjuvant chemotherapy as those which are *ER-*

Table 1.5: Overview of molecular classes and associated genetic signature (Rakha et al, 2008)[3]

As indicated in table 1.5, some studies (Rakha et al, 2008) [3] have reported that the best prognosis is associated with patients with luminal (*ER+*) tumours, and specifically those of luminal-A (or luminal "1) subtype, with the worst prognosis in *HER2* and basal-like (mainly *ER-*) tumours. However, several further points of clinical importance should be mentioned:

(i) The *ER+* tumours are clearly not a single entity and one subclass (*luminal-B*) is reported to show a poor outcome, comparable to the ER- basal-like and *HER2* sub-types

(ii) Tumours which lack *ER*, progesterone receptor and *HER2* expression (a triple negative phenotype), which are currently difficult to treat, can again be further classified into at least two distinct types, namely basal like and normal breast-like groups, each with a distinct molecular signature and behaviour

(iii) Most basal-like and *HER2* tumours have poor prognostic features as defined by routine pathology methods (e.g. lymph node positive), which could have important implications for outcome and clinical management. However, as a consequence of variation in methodology and defining criteria for each class, while some studies have reported a worse prognosis for the basal-like tumours, this is not always a general consensus (Fulford et al, 2007) [88].

It remains to be determined whether expression arrays provide any additional information to adequate histological grading and simple growth fraction (e.g. *Ki67* marker) indices. Data on comparisons between microarray sub classification of luminal breast cancers and the stratification obtained by means of grade, *Ki67*, and *HER2* (FISH/IHC) expression are few and far between. However, it can already be seen that the frequency of the *HER2+* class (4–10%) is somewhat lower than the percentage of *HER2* amplification and over expression (20–25%) as monitored by FISH and IHC respectively in human breast cancers (Slamon et al, 1987) [6]. This low incidence of representation of the *HER2* class in gene profiling studies may in part be due to differences in the criteria of positive values when compared with the present cut-off in immunohistochemistry (IHC). This is defined as 10% of tumour cells, (Ellis et al, 2004) [89] which may result in sampling bias in expression studies.

While the *HER2+* molecular subtype clusters within the *ER-* group, there are some *ER+/HER2* over expressing tumours. Indeed, in some studies up to 50% of patients with *HER2+* tumours have been classified as steroid hormone receptor positive (Konecny et al, 2003) [90], and 17% of steroid receptor positive tumours show *HER2* positivity. Significant proportion of these *ER+* but *HER2*-expressing tumours has been shown to cluster together with other luminal B cancers. Furthermore, a small proportion of *ER-/HER2+* cancers fall into the basal like cluster (Kapp et al, 2006) [5]. It is thus likely that the difference in the incidence of *HER2* tumours between expression profiling and IHC studies again reflects genuine molecular heterogeneity of *HER2*-expressing tumours, and further investigation may help in further understanding the biology of this class and moreover in predicting response to therapies. Indeed, while *HER2-* overexpressing status can be used to select patients for Herceptin treatment which can enhance anti-tumour effect of chemotherapy (Dahabreh et al, 2008) [14], a study by Harris et al. showed that *HER2*-amplified cancers that express genes pertaining to the basal-like cluster at high levels paradoxically show a poor response to Herceptin plus Vinorelbine (Harris et al, 2007) [15]. The biological significance of some of the additional sub-groups remains to be determined.

1.4.2 Prediction of breast cancer survival outcome using microarray-derived prognosis systems and limitations

Using microarray results to make predictions for outcome would be a very powerful and attractive capability in breast cancer. Although there have been many publications quoting a desire to get to a predictive point from microarray research, no other group have made as bold predictions as in a letter to Nature from The Netherlands Cancer Institute written by Laura van't Veer and Stephen Friend from Rosetta Inpharmatics, USA. In early breast cancer although chemotherapy and antihhormonal therapy reduces the risk of distant metastases by $1/3^{rd}$, 70-80% of patients receiving treatment would have survived without it (Van't Veer et al, 2002) [17].

Consequently, the concept of being able to find gene expression signatures for breast cancer that better select patients who require adjuvant treatment was addressed by this group using microarrays. In total, 98 patient's primary tumour samples were studied, amongst which 34 developed distance metastases, 44 remained disease free after 5 years, 18 had *BRCA1* germline

mutations and 2 were *BRCA2* carries (Van't Veer et al, 2002) [17]. 78 of the patients were sporadic patients, less than 55 years old, lymph node negative and previously untreated. In this instance, cDNA *Cy3/Cy5* 25,000 gene microarray's were chosen with sample material being prepared using 5ug of total RNA from snap frozen tumour material upon which cRNA was derived. Reference cRNA was created by pooling equal amounts of cRNA from sporadic tumour cDNA. As usual with the cDNA system, a dye swap/reversal microarray process was required which utilises two hybridisations. Chips were then scanned and normalised (in an unspecified way) and sample results were 'corrected' in such a way to ultimately reveal transcript abundance determination.

Between tumour and reference samples, 5000 genes were found to be differentially expressed based on the criteria of a two fold change and a p-value of less than 0.001 occurring in more than five tumours in the group of 98. In an attempt to group patients and genes, unsupervised hierarchical clustering was then used for the 98 Tumours over the 5000 differential genes. There were broadly two groups initially revealed. As shown in figure 2.10 (1.3), the genes grouped further into those including landmarks such as *ER* and oestrogen regulated genes. The genes could also be divided into those which are enriched in tumours with lymphocytic infiltrate including then landmark genes associated with B and T cells (Van't Veer et al, 2002) [17].

The 78 sporadic lymph node negative tumours were selected to search in detail for a prognostic signature in gene expression profiles as measured in the primary tumour. This encompassed 44 patients who remained disease free for at least 5 years which acted as a good prognosis group, with the remaining 34 developing distant metastases which equated to a poor prognosis group (Van't Veer et al, 2002) [17]. Subsequently, a supervised three step classification method was used which involved finding significantly associated genes with disease outcome, ordering significant genes by association and creating a 'Prognosis classifier' by adding significant genes to it sequentially (Van't Veer et al, 2002) [17]. The results at each analysis stage were as follows:

(1) Starting with the 5000 genes, Pearson correlation coefficient was calculated between expression values for each particular gene and prognostic outcome category. This resulted in 231 genes associated with disease outcome.

(2) These 231 genes were ordered by rank (using p-value). Subsets of 5 were then taken from this list from the top and sequentially added to create a prognosis classifier gene list. Power for correct classification was subsequently tested finally producing a cohort of 70 genes that comprised the prognostic reporter gene set (the "Amsterdam 70 gene signature").

(3) Using this 'Prognosis reporter' gene set to calculate the average good prognosis expression profile, the correlation for each tumour with average profile was then calculated. Tumours were ranked in order by correlation and significant correlation score was calculated. All tumours with a correlation above a significant level had a good prognosis signature as show in figure 1.14.

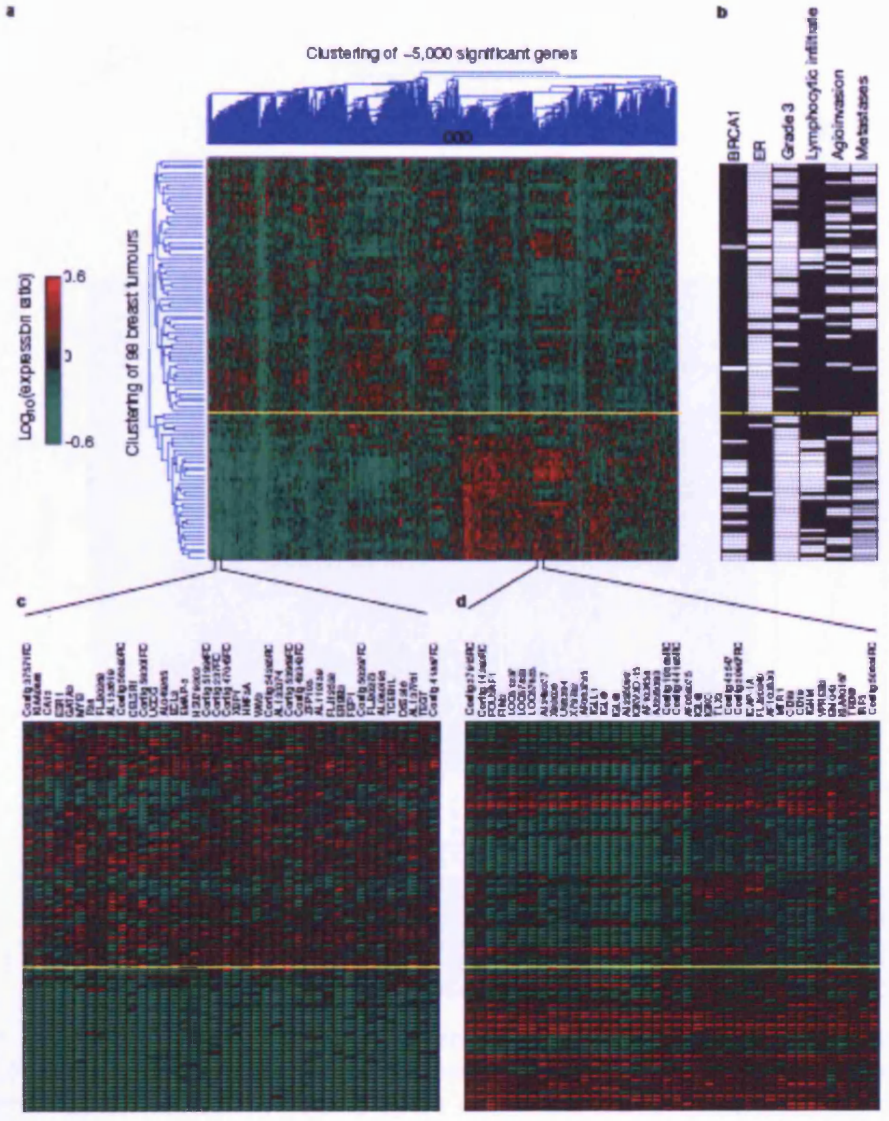


Figure 1.14: Unsupervised two-dimensional cluster analysis of 98 breast tumours from Van't Veer et al (44) showing (a) 4968 significant genes across the group; (b) selected clinical data for the 98 patients in various states – *BRCA1* through to metastases; (c) Enlarged portion from (a) showing genes that co-regulate with *ER* α gene *ESR1*; (d) Enlarged portion from a containing a group of co-regulated genes which are molecular reflection of extensive lymphocytic infiltrate and comprise a set of genes expressed in T and B cells. Black equals negative, white equal positive for figure (b). Intensity has been assessed in the HCA plots on an increasing scale of green to red (Van't Veer et al, 2002) [17].

The prognosis classifier using this reporter set was validated using a subsequent test set comprising of 19 young lymph node negative Breast Cancer patients, of which 7 were disease free after 5 years and 12 which developed metastases within 5 years. Only 2 out of 19 were misclassified resulting in an apparent 89% predictive accuracy (Van't Veer et al, 2002) [17].

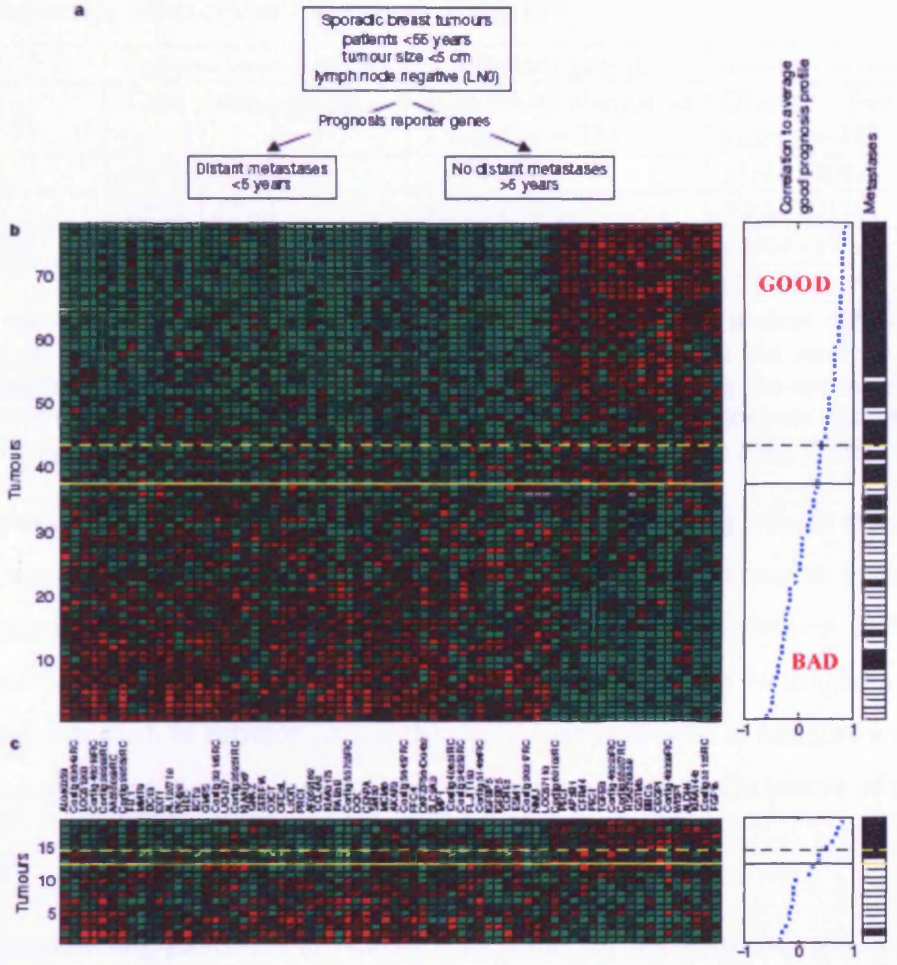


Figure 1.15: Supervised classification on prognosis signature using (a) a prognostic reporter; (b)Tumours ordered by prognostic classifier optimal accuracy – those above dashed yellow line have good prognostic signature, those below is a poor signature; (c) Similar plot as in (a) however with additional patients (Van't Veer et al, 2002) [17].

The study found than in women under 55 diagnosed with lymph node negative breast cancer with a poor prognosis gene signature have a 15 fold odds ratio to develop metastases in 5 years compared with good prognosis signature. It was deemed that the predictive value of the classifier was also superior to current clinical and histopathological factors, with increased power of the predictive Microarray classifier demonstrated using multivariate logistic regression. The St. Gallen, NIH and NPI criterion was also applied to the Van't Veer Microarray classifier group for patients to compare with results, as outlined in table 1.6. The St Gallen and NIH approaches clearly over predicted the number of patients who required treatment since this was

recommended in up to 91% which would remain disease free at up to 5 years, potentially incurring unwanted side effects (Van't Veer et al, 2002) [17].

	←----------	Patient group	----------→
Consensus	Total patient group (n=78)	Metastatic disease at 5 years (n = 34)	Disease free at 5 years (n=44)
St Gallen	64/78 (82%)	33/34 (97%)	31/44 (70%)
NIH	72/78 (92%)	32/34 (94%)	40/44 (91%)
Prognosis profile	43/78 (55%)	32/34 (91%)	12/44 (27%)
NPI			(18/44 (41%)

Table 1.6: The convention consensus is a tumour >= 2cm, ER- , grade 2-3, patient <35 years old (for St Gallen) whereas tumour >1cm (NIH consensus). Prognosis classifier is the number of tumours having a poor prognostic signature using the microarray profile, defined by the optimised sensitivity threshold in the 70 gene classifier. NPI – Nottingham prognostic index – numbers of tumours with a poor prognostic signature in the group of disease free patients (Van't Veer et al, 2002) [17].

The study revealed that the microarray prognostic classifier effectively selects those high risk patients which would benefit from adjuvant therapy but also reduces the number of patients who would be recommended to receive what proves to be unnecessary treatment (27%). It also revealed the microarray approach was somewhat more superior to the Nottingham prognostic index (41%) in this regard. In addition, genes that were over expressed in tumours with the poor prognosis gene profile could comprise potential targets for rational development of new cancer drugs.

This landmark review was published in January 2002 however the study was perceived to have several major flaws which other cancer groups around the world tried to address. The first main issue which is obvious from the review is the complete dependence on hierarchical clustering (HCA); where no other method was used to confirm the classifier gene set. This is problematic as HCA used alone is notoriously difficult to interpret. Two subsequent publications revealed issues with the original research which acted as a warning to others in regard to using microarray HCA to define a predictive set. Firstly, Gruvberger *et al* from Lund University Hospital in Sweden applied the logic shown in the Van't Veer *et al.* paper, however was cautious in predicting clinical variables from the gene expression data as they have found that it was complicated by an interaction between *ER-α* status and the clinical parameters studied (Gruvberger et al, 2003) [91]. The genes used were essentially not believed to be significant independent discriminators for the tumours in the study, rather they were part of the *ER+* or *ER-* phenotypes. Consequently

the group used a different classification method to examine if the Van t'Veer outcome predictor set was able to predict relapse in an independent 44 lymph node negative tumour series, but unfortunately it failed to predict with enough statistical confidence in this set (Gruvberger et al, 2003) [91]. The multidimensional scaling technique plots, as previously introduced, can be seen in figure 1.16. MDS displays the position of each tumour sample in a three-dimensional Euclidean space, with the distance between the samples reflecting their approximate degree of correlation.

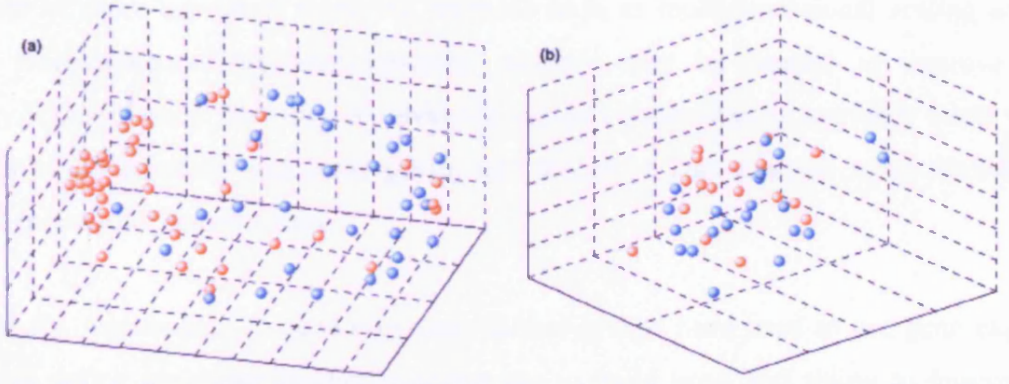


Figure 1.16: Multidimensional scaling (MDS) clustering of gene expression data from breast tumours using 58 out of 231 genes from the outcome predictor gene set identified by Van 't Veer *et al.* The study found that genes retained their predictive value in the data however not in the independent patient sample. (a) Fifty-eight primary breast tumours (training set) from the study by Van't Veer *et al.* and (b) 44 from the Swedish group array study were plotted. Tumours with a poor prognosis (distant recurrences within 6 years) are coloured blue and tumours with a good prognosis (no recurrences within a follow-up period of 5–14 years) are in red (Gruvberger et al, 2003) [91].

The correlation between prognosis and ER status in the Van t'Veer study may have led to the selection of a prognostic set that may not be broadly applicable to other breast tumour samples where no correlation between ER status and prognosis exists (Gruvberger et al, 2003) [91]. Furthermore the usage of MDS is widely acknowledged as a better and more robust classification method and should be considered as an advanced pattern discovery method, rather than examining HCA alone.

A second study published in the Lancet tried a similar technique to the original Van t'Veer study, but in this instance used the Affymetrix platform rather than *Cy3/Cy5* array system. The study identified aggregate patterns of gene expression, "metagenes" (=clusters), which associated with lymph node status and recurrence. These were capable of predicting outcomes in individual patients at up to 90% accuracy. While some of the minor metagenes included 17 out of the 70

genes used in the Van t'Veer predictor set, none of them appeared in their key 'metagenes' that related to recurrence (Huang et al, 2003) [92].

All three papers show that multiple factors are involved to determine a robust, effective predictive set, encompassing analysis strategy and also the microarray conditions. The reviews have been useful to show that more specialised classification methods need to be considered rather than simply hierarchical cluster analysis as heavily focused upon in the Van't Veer study. The power of more advanced clustering methods such as multidimensional scaling and even machine learning/neural network clustering methods may be needed to improve cluster efficiency. Consequently, focusing on advanced analysis capabilities is a priority when working with a set of differentially expressed genes, and should be encompassed when developing an improved microarray analysis suite.

Alongside the Amsterdam 70 gene signature, further groups have tried to use gene expression profiling to define molecular predictive signatures to build upon and trying to improve upon prognostic abilities commercially. There are two main multigene assays that have been verified and are now under evaluation in breast cancer clinical trials in this regard. These are MammaPrint (Netherlands Cancer Institute signature), which aims to address prediction of outcome and need for adjuvant chemotherapy; and polymerase chain reaction measured OncotypeDX. (21-gene signature) that aims to predict relapse after tamoxifen treatment (Ross et al, 2008) [22]. These differ in that the starting test material is different; however both encompass proliferation, *ER* and *HER2* pathway information. Each assay is undergoing different clinical trials. The features of both systems are summarised in table 1.7.

Further markers and assays cover prognosis and response to therapy as summarised in figure 1.20. One of the most well known is the Rotterdam 76 gene signature developed as a pure prognostic assay and most notably has no genes common with the Mamaprint or the oncotypeDX platform and it is based on the Affymetrix HGU-133A array platform – it is validated for predicting outcome in lymph node negative patients independently of hormone receptor status.

	Oncotype DX	MammaPrint
Supplier	Genomic Health, Inc	Agendia
Starting material	Formalin-fixed Paraffin Embed	mRNA
Number of genes	21	70
Rank of pathways assessed by importance	1 Proliferation 2 *ER* 3 *HER2*	1 Proliferation 2 *ER* 3 *HER2*
Current indication	Node negative ER positive	Node negative *ER+* *ER -*
Patient age recommendation	Older patients	Young and old patients
Outcome prediction	Continuous	Dichotomous
ASCO guidelines	Recommended for use	Under study
Clinical trial details	TAILORx ("Trial Assigning Individual Options for Treatment") Node negative, ER positive Design: Who with intermediate risk will benefit from chemotherapy?	MINDACT ("Microarray in Node Negative Disease may Avoid Chemotherapy"). Node negative, ER+, *ER-* Design: Who will have an excellent outcome without chemotherapy?

Table 1.7: Comparison of Oncotype DX and Mammaprint (Ross et al, 2008) [22].

The Rotterdam signature also has promise in revealing recurrence in ER+ patients treated with Tamoxifen (Ross et al, 2008) [22]. NuvoSelect uses a 30-gene set to predict complete response to preoperative paclitaxel (Taxol®), 5-fluouracil doxorubicin (Adriamycin™) and cyclophosphamide (TFAC) chemotherapy with a 200-gene set to predict tumor response after 5 years of endocrine therapy. The Roche Amplichip is essentially a metabolism test – analysing *CYP2D6* and *CYP2C19*, two genes encoding key enzymes from the cytochrome P450 system that greatly influence drug metabolism. It is a step towards individual therapy of a patient as the result directly relates to the phenotype of the individual as to how well they metabolise drugs which, in a cancer setting ultimately has an impact on their effectiveness. Dosing of drugs such as Taxmoxifen can therefore be varied accordingly. A summary of these platforms is summarised in table 1.8.

Review of the gene lists across these multiple expression systems demonstrates little cross-over and has led to some debate as to reproducibility. However, a study has compared the signatures and has shown that despite little concordance in the precise genes identified, such signatures do seem able to uniformly stratify patients according to outcome, suggesting they are defining a common cellular phenotype (Fan et al, 2006) [93] .

Platform	Rotterdam signature	NuvoSelect	Roche Amplichip
Company	Veridex	Nuevera Bioscience	Roche/Merck
Platform	Microarray	Microarray	Microarray
Type of Platform	Affymetrix HG-U133A	Affymetrix HG-U133A	Roche Amplichip
Starting material	Fresh Frozen	Fresh Frozen	Fresh Frozen
No. of genes	76	200	1 (P450)
Indication	Neoadjuvant predictor of TFAC response; prediction of response to hormonal therapy	ER positive; ER negative; LN positive; LN negative	N/A
Guide to therapy	Possible (Tamoxifen)	Yes (neoadjuvant TFAC, Tamoxifen)	Yes (Tamoxifen, CypChip)

Table 1.8: Prognosis and response to therapy commercial assays (Ross et al, 2008) [22].

1.4.3 *In vitro* studies

Several groups are studying factors underlying drug response and resistance through microarray studies *in vitro*. A primary goal of the microarray profiling in these models is to discriminate potential markers that can be used to develop therapies and predictive gene sets applicable to drug failure. For example, Tamoxifen has been shown to negatively influence more than 60 genes regulated by the oestrogen estradiol in the *ER* positive MCF-7 cell line. The gene regulation is mediated via *ERα* and reversed by estradiol. It was found that some of these genes can be used as markers of development of Tamoxifen resistance – namely *YWHAZ* and *LOC441453*- which correlated strongly with disease re-occurrence resulting in the proposal that these should be considered as markers of poor prognosis (Frasor et al, 2006) [20]. Determination of the gene profile underlying Tamoxifen resistance and associated tumour progression is also a key aim of the Tenovus centre for cancer research, in addition to the above described candidate pathway approach. Expression profiling with microarrays is being performed across the TAMR model versus its hormone responsive MCF-7 counterpart (Nicholson et al, 2005) [21]. Analysis of this model in the Tenovus group has progressed from nylon and plastic array formats, to the Affymetrix HGU-133A platform. A recent success of this microarray approach has been the initial discovery of a zinc transporter, *ZIP7*, which is over expressed in the tamoxifen resistant TAMR cells (Taylor et al, 2008) [94]. Levels of intracellular Zinc within TAMR cells have also recently been shown to relate to the subsequent response of anti-hormone resistant cancers to

treatment with agents that target growth factor pathways (Taylor et al, 2008) [94]. Subsequent studies have shown that targeting of *ZIP7* in TAMR will suppress zinc induced events and has a knock on effect of inhibiting signalling through multiple growth factor pathways, increasing the chances of cell death and consequently treating resistance[94]. *ZIP7* is now being examined in clinical material with parallel anti-hormone response data to address its prognostic/predictive capacity.

Microarray technology is clearly facilitating exploration of breast cancer cell models in order to better understand biological behaviour, including the breadth of mechanisms of therapeutic response and resistance, in particular factors underlying growth and cell survival and as well as invasiveness. In doing so, potential prognostic/predictive factors and drug targets are emerging, as exemplified by the TAMR experimental model studies. Larger array platforms are increasingly being employed for such studies, coupled with initiatives emerging that are increasing the complexities of samples analysed (e.g. mRNA preparations comparing multiple cancer cell models and treatments).

In the year 2000, the National Cancer Institute development therapeutics department carried out intensive expression studies of 60 cell lines created from a range of tissues and organs with the research led by Douglas T Ross (Ross et al, 2000) [95]. In relation to cancer agent and chemotherapeutic sensitivity of the 60 cell lines, more than 70,000 different chemical compounds had been tested which ultimately revealed a connection between the pattern of response to a drug and its method of action as measured at the level of gene signature. Due to the connection between the function of a gene and its pattern of expression, the pattern of this gene expression can reveal novel phenotypic aspects of cells and tissues studied (Ross et al, 2000) [95]. The main advantage of applying this concept over the large number of cell lines is that it should result in detailed (and potentially-clinically relevant) understanding of human gene expression and hopefully give indications of the physiological roles uncharacterised genes may perform. This information can then be stored electronically and shared with the research community, particularly with research groups working on the gene signature of individual cell lines without the luxury of a cross spectrum portfolio.

In breast cancer, the various cell lines available to researchers can be broadly categorised according to the clinical breast cancer classes (Neve et al, 2006) [96]. However, where comparison of gene expression signatures from breast tumours and individual breast tissue-derived cell lines has been discussed, caution was a consistent theme whereby careful assessment of the nature of a particular cell line as a model for particular breast tumour subtypes was highlighted as essential (Neve et al, 2006) [96]. Again, the relationship between signatures derived from individual cell lines and clinical disease may be more robust if larger breast cancer cell line microarray databases can subsequently be accessed. The key differences between clinical tumours and cell lines is usually due to differing growth rates however many of the gene expression patterns can be related to normal physiology which distinguishes different cell types *in vivo* (Ertel et al, 2006) [97]. *Dennis Slamon* from UCLA's Johnson comprehensive cancer centre has expanded the mass clustering of different breast cell lines in relation to drug sensitivity. The group have over 100 different cell lines which have been clustered against each other in relation to anti-cancer agent response in addition to known genetic alterations within the cell lines, which is a very powerful asset to the group when trying to better equate with clinical phenotype (Slamon et al, 1987) [6]. With regards specifically to anti-hormone resistance, the Tenovus cancer research group have also recently expanded their study through development of a unique, broader panel of anti-hormone responsive and resistant MCF-7 breast cancer cell lines (encompassing resistance to various anti-oestrogens or oestrogen deprivation), with profiling also extending to encompass response and resistance to anti-growth factors such as *EGFR* and *erbB2* inhibitors (Nicholson et al, 2004) [98]. Genetically, optimisation of procedures for robust identification of differential genes expression across multiple control/treatment/resistance groups is important for analysis success. Having identified genes involved in each different scenario for particular anti-hormonal agents, it would be interesting to apply the gene set revealed against the different therapeutic agents to determine if this is an individual or generic predictive set. This could then determine resistance elements that may be relevant as targets in resistance to multiple types' anti-hormones or potentially as targets for individual anti-hormone resistant states. In addition to revealing therapeutic targets there is key interest in determining predictive sets of genes for resistance to treatment with anti-hormones such as the antioestrogens Tamoxifen and Faslodex. Identified predictive sets could then be applied clinically to test if they improve patient stratification for responses/predict outcome/prognosis.

1.5 Data Mining Large Clinical datasets for Application of Bioinformatics Classification Methods to Reveal Improved Clinicopathological Prognostic Indices in Non-microarray datasets

While results are clearly promising, mainstream use of microarray gene signatures derived from clinical material (or emerging from *in vitro* data) for prognosis and to predict outcome of therapy is still under intense evaluation. Among the various techniques, Oncotypedx remains the furthest advanced; since it has recently been recommended for clinical use in the recent ASCO guidelines for the use of tumour markers in the management of breast cancer (Harris et al, 2007) [99]. However, there also remains (at least in the short term) a need to evaluate if established clinicopathological prognostic models, notably the NPI, can be further improved upon using novel combinations of existing genetic and clincopathological variables addressed through development and implementation of bioinformatics approaches. In this regard, initiatives are emerging to construct very large cancer databases with clinical follow-up and pathological information, which with appropriate analysis should allow exploration to further improve prognostic indices.

Particular interest in this regard is the Surveillance, Epidemiology and End Results (SEER) program from the National Cancer Institute that offers survival and patient parameters between 1975 and 2005 (Ries et al, 2005) [100] for an expanding breast cancer patient series. Although different from the dataset used to calculate the NPI, the same information can nevertheless be extracted and an NPI equivalent calculated and applied to the SEER dataset. This dataset is novel in that it is one of the only available datasets of such size available for analysis stored in a single repository. To date, few data analysis packages utilise this powerful clinical resource particularly in the context of discovery of new covariates which ultimately can be used to discover new prognostic markers. The power of the SEER dataset to improve models for breast cancer prognosis could be exploited if flexible data analysis tools were developed to find relationships common across the thousands of patients by analysis beyond that of simply tumour size, grade, and lymph node status. Moreover, as the SEER dataset has in depth patient information, treatment information can also be analysed in relation to prognostic models in terms of surgery received and radiotherapy.

AIMS AND OBJECTIVES

As a result of the numerous methodologies available for improved understanding of breast cancer using high throughput technologies, the following aims were developed as the main objectives of this project:

- Develop a new user friendly Affymetrix microarray analysis and visualisation suite by :
  (a) Assessing and choosing the user interface to facilitate ease of use of the software.
  (b) Selecting optimised analysis algorithms to enhance Affymetrix array analysis incorporating a means to perform quality control, advanced statistical analysis, multiple clustering with enhanced visualization, differential gene prioritisation and ontological exploration
  (c) Implementing various analysis algorithms into a user friendly application for high throughput analysis (incorporating choice of an effective programming language able to direct these analyses and to interplay with the array database).
- Demonstrate the capability of using this developed software for robust identification of differentially-expressed genes in this instance the context of identifying potential markers to discriminate endocrine resistance from response and potential new therapeutic signalling targets, using microarrayed *in vitro* breast cancer models. This will encompass analysis techniques including class prediction, annotation and target discovery applied through the developed software to two acquired anti-hormone resistant cell lines versus their responsive counterpart. These models were previously derived in the Tenovus Centre for Cancer research, emerging during prolonged exposure of the *ER+* MCF-7 breast cancer cell line to a 10-7M dosage of the SERM Tamoxifen (TAMR cells) or of the pure antioestrogen Faslodex (FASR cells), two clinically-valuable anti-hormonal agents in *ER+* breast cancer management. In order to potentially provide generic markers/therapeutic targets for antihormone resistance, particular focus will be placed on applying analysis approaches that identify differentially-expressed gene sets shared by the two resistant states (and within this revealing potential growth/invasion signalling

elements altered in resistance), alongside identifying those gene cohorts associated with individual classes of antihormone resistance.

- Develop an analysis tool of potential value to explore the effect of multiple clinicopathological parameters in the context of impact on patient survival based on a large published breast (and colorectal) cancer data set. Features to be prioritised include ability for the tool to be able to perform single and also dual cohort patient comparisons, and to incorporate superior visualization of results.
- Subsequently use advanced computational techniques (such as decision tree analysis) to further improve accuracy of survival prediction using existing prognostic factors, comparing the findings with existing prognostic factors relationships such as the NPI score.

EXPECTED OUTCOMES:

- Provide an optimised set of data analysis tools able to advance our understanding of the breadth and patterns of transcriptional impact of therapeutic resistance, and therein also relevant for future marker and target discovery.
- Identify in the thesis molecular marker sets that could be potentially be tested in clinical material in the future to see if they equate with endocrine response/failure, where such measurement could ultimately improve patient stratification for response/failure and prognosis
- Identify a number of novel signalling targets in the thesis which, if verified in the future at the protein level as functional, could potentially be manipulated to treat resistance with its adverse phenotype alongside existing therapies (either by inhibition of the new signalling target where this is induced in resistance, or by its restoration where expression is lost) .
- Provide a clinical cancer query survival tool which (alongside being of value to researchers) could assist oncologists and patients in estimating individual patient outcome and hence aid management decisions, potentially also estimating the potential benefits of different treatment strategies in this context.
- Begin to uncover new combinations of prognostic markers based on exploration of the SEER dataset covariates using advanced statistical modelling techniques.

Chapter 2

Informatics system Tenovus

'I-10' development

Chapter 2 – Informatics system Tenovus – 'I-10' development

2.1 Background

A review of available tools for microarray analysis as outlined in Chapter 1 shows an interesting spectrum of options to the biologist. However no individual tool offers flexibility in terms of future upgrades or even embraces the breadth of available analysis methodologies once initially purchased. It was clear that some applications offered useful strategies and approaches to microarray analysis however the biologist would have to subscribe or purchase multiple products to cover all analysis methodologies. Many of the freely available research driven applications – such as BRB Array tools – demonstrated the power of open source technologies. Therefore it would be highly desirable if the strengths of each individual application could be encompassed in a single application developed using freely available technologies such as in the case of BRB Array tools. This would create a highly desirable tool not only for use within Tenovus but also for the greater cancer research community as a whole and facilitate more rapid discovery of new genetic land marks from microarray experiments.

2.2 – Graphical User Interfaces

Before in depth discussion of how I-10 was successfully created, it is important as a background to appreciate the origins of how current computer technology and user interaction with computers has evolved over the last 20 years. The resultant choice of technologies has led to why I-10 runs on a Microsoft Windows operating system. However it was not the only possible choice. Three key computer operating systems exist – Microsoft® Windows, Apple Operating System X and various forms of LINUX such as Red Hat or Fedora. An operating system is the most important software component a computer uses – it contains all information about the computer, how the user interacts with the computer and how information is returned to the user.

Recent versions of the Apple operating system X (OSX) have an 'open source' LINUX basis, where open source applications refer to a community of users who have developed software in a non-profit, often non-commercial environment. However, in development of an analysis system,

it is important to appreciate aspects users are most familiar with using day to day, and to appreciate that the resources commonly available for development can also often be a constraint. Consequently there is a bias from an end point user in that the vast majority of organisations and Universities use Microsoft® Windows when delivering applications to employees and students. Figure 2.1 highlights that Microsoft® Windows is used by over 90% of the world's personal computers, in the past twelve months alternative operating systems have increased slightly reducing Microsoft's dominance in the market place (W3 consortium, 2008) [101]. However Microsoft® remains far ahead of its rivals in overall market share. This historically has been due to better support being available compared to its rivals. Apple OS X has nearly a 5% of the market share and Linux 2%. The ability of the world's population to use computers is constantly improving with more established users being increasingly more adventurous with software choices – embracing LINUX solutions and applications. This has been facilitated, in part, by hardware manufacturers of computer technology designing and supporting different operating systems which a user can now choose for their computer.

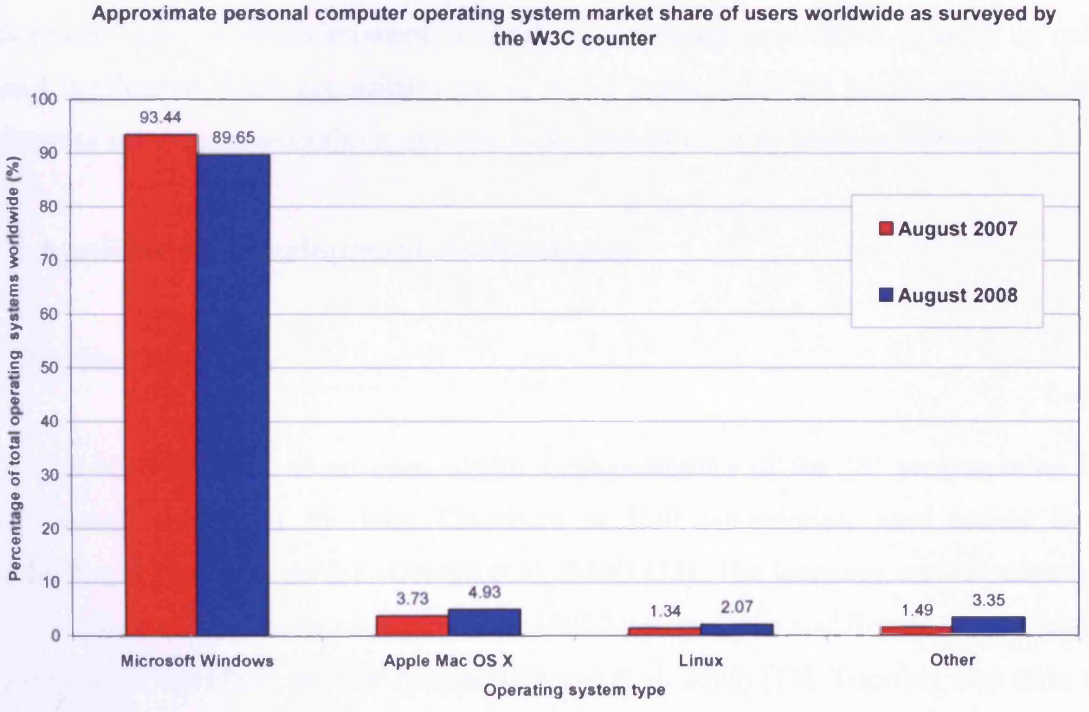


Figure 2.1: Distribution of operating systems in use worldwide as surveyed by over 45 million visits to the W3Counter in the USA. Microsoft's dominance has reduced slightly in a year. 'Other' operating systems include mobile telephones with internet access (W3 consortium, 2008) [101].

The most notable development in the computer market has arisen from Apple's switch to Intel Corporation processors from 2005 onwards instead of Motolora or IBM custom-built processors. The motivation for such a change was the Windows based systems using Intel technology were increasing in speed at a faster rate than the processors Apple had been using (Stratton et al, 2005) [167]. Intel processors use less energy and run cooler than equivalent IBM or Motorola chips – a crucial development which helped Apple develop the 'Mac book Pro' – a laptop instead of desktop computer. This in itself has allowed Apple to gain market share. However, Microsoft Windows based Intel processor machines remain the quickest personal computers available on the market. As a result the development of an array analysis system to run using Microsoft Windows is of great importance to appeal to a wide range of users and the associated performance benefits.

When subsequently focusing upon design of data analysis applications for use in the Microsoft® Windows Operating system, there are three obvious choices of user interface. These include: the Microsoft Excel client/environment; a novel custom design application or lastly an online web based application. Each necessitates use of novel technologies for application development to maximise subsequent analysis capabilities in the developed array analysis software.

2.3 Application Development Technologies

2.3.1 – The 'R' project

'R' can be considered as an open source implementation of the 'S' programming language environment developed by John Chambers at Bell laboratories, now owned by Lucent Technologies, formerly AT&T (Dessau et al, 2008) [73]. The language was reincarnated as the 'R' statistical programming environment in 1997 by Ross Ihaka and Robert Gentleman from the University of Auckland, in New Zealand (Dessau et al, 2008) [73]. Together they form the basis of the R Development Core Team for new implementations. The name 'R' was given to the language as both the original core team have Christian names starting with the letter 'R'. Updates of the application occur frequently with the most recent version being 'R' version 2.7.1 at the

time of writing. 'R' is available in multiple versions for all the three key operating systems Windows, MAC OS X and LINUX (Dessau et al, 2008) [73]. The source code is also available for custom installations such as 64 bit computing environments as used by super computing facilities.

'R' is predominantly a scripting language. Thus it is packaged for installation as a command line application, with scripts entered within a simple 'R' graphical user interface (RGUI; as shown in figure 2.2). To enable 'R' to communicate with other computer applications, a communication interface available only for Microsoft Windows systems is used to remotely command 'R' using scripts embedded in other Windows applications. Microsoft® Excel is a popular choice of such an application that can communicate with 'R', however custom applications to harness the capabilities of 'R' can be written using the programming language Visual Basic.

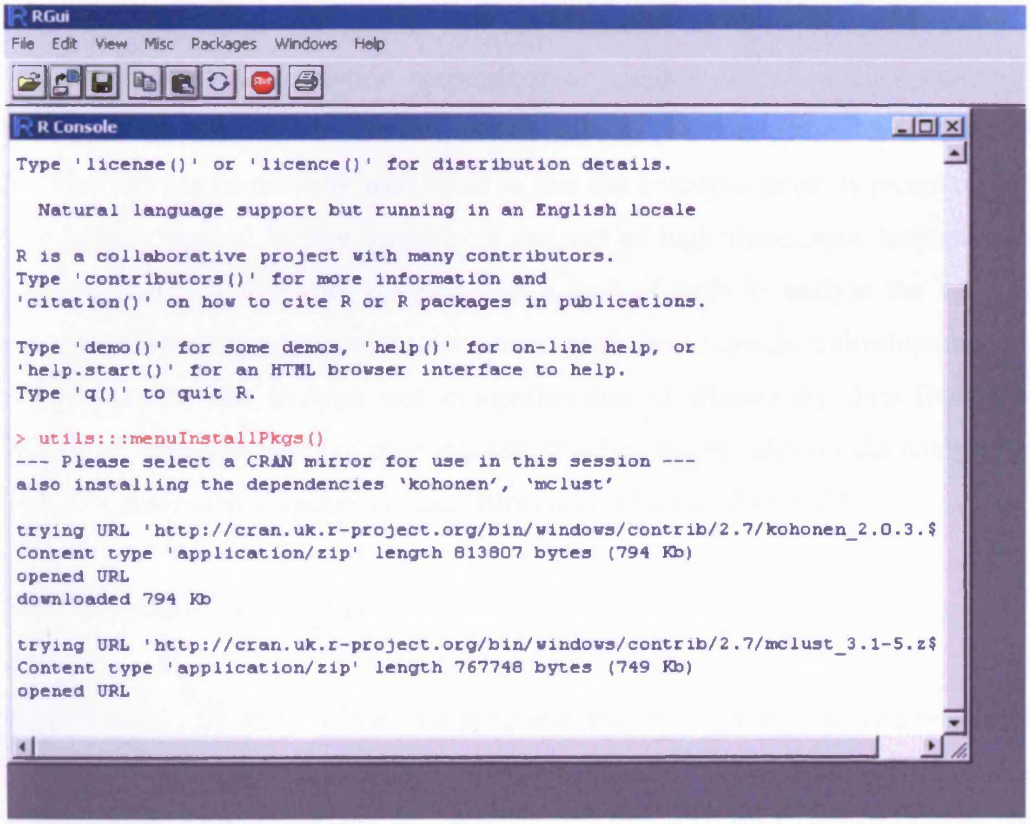


Figure 2.2 – Screen capture of the 'RGui' console window where 'R' can be commanded and functions trialled before potentially being integrated into applications. Installation of packages can also take place here as certain libraries will be required by analysis steps.

The main strengths of 'R' lie within the flexibility it offers as a fully planned and coherent system to allow the user to access comprehensive microarray analysis capabilities rather than an inflexible set of very specific software packages. Additional functionality is feasible through new libraries that are developed to a particular area of study. As outlined in Chapter 1, several clustering methods can be used for data analysis, all of which have been coded as 'R' libraries.

The ability of R to be commanded and seamlessly integrated with other user friendly applications is vital if users unfamiliar with any form of computer programming are to be able to harness the powerful statistical functionality of 'R'. Communication of instructions from Excel to R is typically achieved utilising the Microsoft® office interface (D)-COM, a relationship which is covered in depth in subsequent pages (Baier et al, 2003) [102].

2.3.2 Bioconductor

Bioconductor is a worldwide consortium developing 'R' libraries containing analysis tools developed primarily for microarray data analysis (Bioconductor Core, 2002) [72]. Worldwide contributors are continually welcomed to join the initiative since its inception in late 2001. This was largely fuelled by the increase in the use of high throughput technologies together with genome completion studies coupled with a lack of tools to analyse the large volumes of data generated. The Bioconductor project's main goal was to design a development suite commanded within 'R' for the analysis and comprehension of Microarray data from the various array platforms, although many of the tools can now be used broadly for the analysis of other types of genomic data, such as sequence data (Bioconductor Core, 2002) [72].

2.3.3 Microsoft® Visual Basic (VB)

Visual Basic was derived from the programming language BASIC and enables development of graphical user interface applications for the Microsoft® Windows operating system. Visual Basic was designed to be relatively straightforward to learn and use particularly for those with some computer programming experience and particularly those wishing to enter into the programming arena for the first time. The language not only allows programmers to create simple Windows

applications yet also has the power to enable more complex application development. Programming in VB takes place within a specially designed Microsoft Windows application where components are added to a form and the user specifies attributes and functionality of individual components. Although many options are pre-written, the programmer can write code directly. The application converts the different 'forms' developed into a compiled executable file which makes application development fast. An example of a form is shown in figure 2.3.

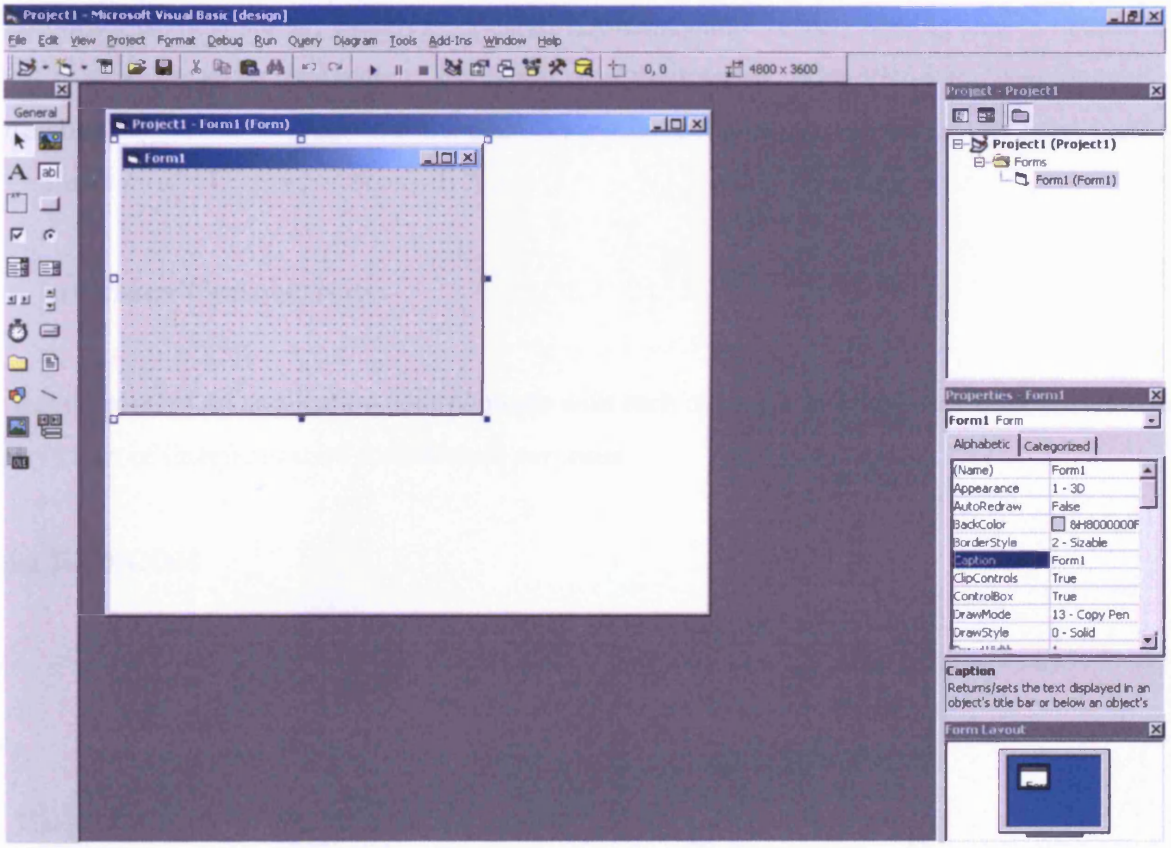


Figure 2.3: A Visual Basic form shown within Visual Basic 6.0 Studio for application development

The last version of Visual Basic was VB 6.0 in 1998 before Microsoft stopped technical support for the application. Due to the rapid growth of the internet, Microsoft launched Visual Basic.net to superceed Visual Basic. Microsoft® offers a newer, freely available application development envirnoment similar to the VB6.0 application called Visual Studio.NET. Due to the rise of open source technologies Microsoft was pressurised to offer a freely available version called Visual Studio.net. Most of the types of application developed in VB 6.0 can be developed in Visual Studio.net. Due to the web based nature of VB.NET technology, it is recommended that

development occurs using a Microsoft® Windows Server. A Windows server allows secure remote desktop connections over which development can take place which has positive security and back-up advantages as no key files reside on any developers machine – they are all kept on the server.

Having a Windows Server is a large investment for VB.NET technology, without even considering the investment required for DNS (Domain name server) hosting that is needed for users to access a given application on a server over the internet. With VB6.0 the development of individual applications which are installed on every user's machine does not require such an investment.

2.4 Interface Connectivity

Different parts of an application communicate with each other via an interface. However there are many types of interfaces used for different purposes.

2.4.1 R-(D)COM

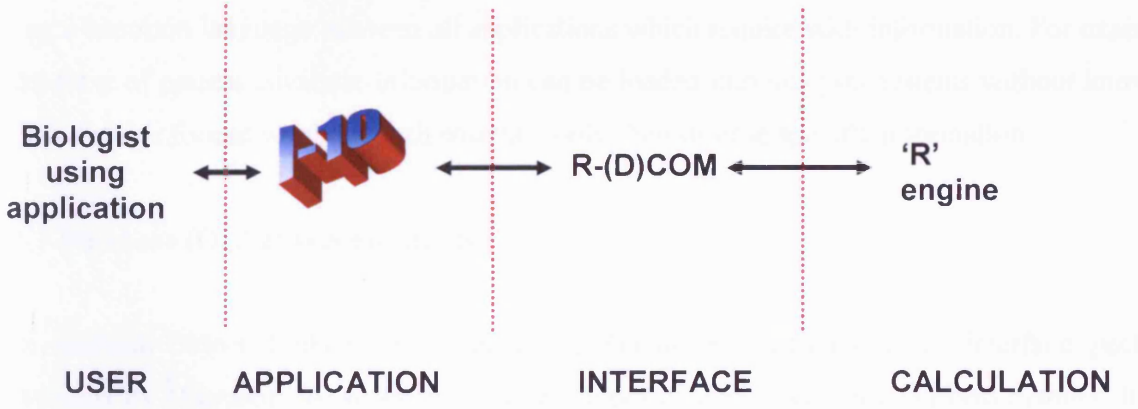


Figure 2.4 – Interaction between I-10 and 'R' using the R-(D)COM interface.

R-(D)COM acts as a programming interface to COM and DCOM (Microsoft distributed object interface) to access the R calculation engine. (Baier et al, 2003) [102]. A typical use for R-(D)COM is in the connection between Visual Basic to 'R' or Microsoft® Excel to 'R' as shown in figure 2.4, allowing commands and data to be passed between the two efficiently, an aspect which has proven to be a central benefit of the I-10 operation approach.

2.4.2 Web services

A Web service as defined by the W3C consortium – responsible for standardisation of web applications – is "a software system designed to support interoperable Machine to Machine interaction over a network" (W3Consortium, 2004) [103]. Web services provide a way of linking systems which perform unique tasks, each offering something different yet desirable in any given application. Web service technology utilising multiple protocols such as 'Simple Object Access Protocol' (SOAP) allow many different computational objects to communicate seamlessly with each other, interchanging data in multiple directions (W3Consortium, 2004) [103]. Web services can add a layer of data security in that data is never exposed and exchanged within applications and avoids manipulation by the user in any way.

Web services are based on XML technology for encoding of information so that it can be shared using a common language between all applications which require such information. For example, a database of patient covariate information can be loaded into analysis systems without knowing any in depth information about each patient – only their disease specific information.

2.4.3 Database (OLEDB) connectivity

The method Object Linking and Embedding Database (OLEDB) is an interface package developed by Microsoft for accessing different types of data stored in a uniform manner. It is a set of interfaces implemented using the Component Object Model (COM) previously introduced in the DCOM section. The system was designed to be a higher-level replacement for Open Database connectivity (ODBC) having additional features to support different types of non-

relational databases, this can included object databases and spreadsheets that do not necessarily implement SQL.

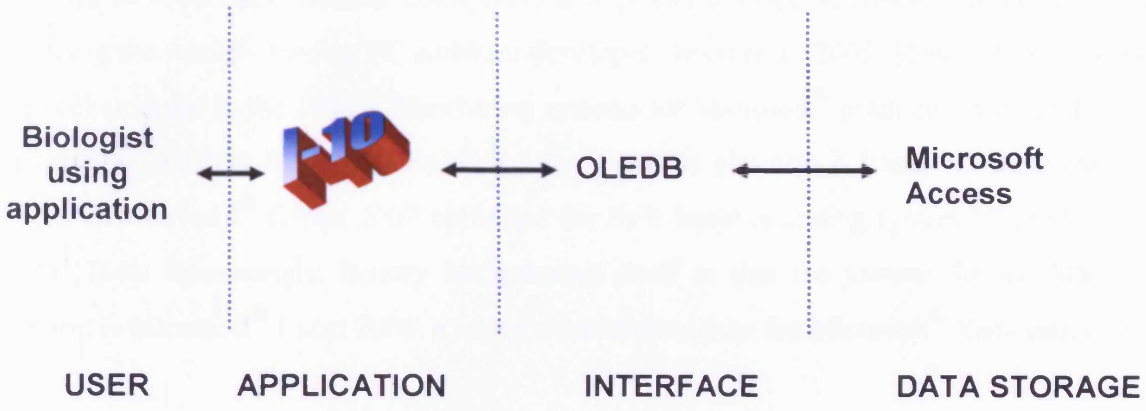


Figure 2.5 Interaction between I-10 and the Affymetrix array database stored in Access which also stores analysis results.

OLE DB separates the data store from the application that requires access through a set of configuration steps that include the data source, for example a Microsoft® Access database, logon information and the query to retrieve information from the database. The technology was developed as different applications may require access to different sources of data without themselves being able to retrieve the information. OLE DB is conceptually divided into consumers and providers. The consumers are the applications that need access to the data, and the provider is the software component that implements the interface and therefore provides the data to the consumer. This component is vital in any data analysis system where the data to be stored is in a database and needs to be called from the analysis application as seen in figure 2.5.

2.5 Microsoft® Excel

Microsoft® Excel started life under a different name – largely known as Multiplan in 1982. It had major competition from Lotus 1-2-3 which itself was based on the very first spreadsheet application – VisiCalc developed by Software Arts that was later bought by Lotus (where Lotus itself was later bought by IBM corporation) (Jelen et al, 2005) [164]. Ironically for a Microsoft®

development, the first version of Excel was released for the Apple Macintosh in 1985 and the first Microsoft® Windows version released in 1987. It is widely believed the downfall for Lotus was the slower development of the company of its applications for the Microsoft® Windows platform. In 1988 Excel outsold Lotus 1-2-3 and greatly assisted Microsoft® in its success as to becoming the world's leading PC software developer (Jelen et al, 2005) [164]. Multiple versions of Excel emerged in the 1990's. Numbering systems for Microsoft® products ended at the turn of the century; however the latest version for the Windows platform is Excel 14, more commonly known as Microsoft® Office 2007 optimised for their latest operating system Vista (Jelen et al, 2005) [164]. Interestingly, history has repeated itself in that the version for the Mac OS X platform is Microsoft® Excel 2008, a newer development than the Microsoft® Vista equivalent.

Microsoft® Excel is important in data analysis in terms of the way novel applications can be written to harness the Excel interface. Since the early 1990's, Excel has included Visual Basic for Applications (VBA), a programming language based on Visual Basic which adds the ability to automate tasks in Excel. The term 'Macro' was developed as a way to automate commands which users performed routinely. However from an application development position, VBA is a powerful addition to the application which in the latest versions features an integrated development environment (IDE). The IDE is in essence a way in which interfaces such as R-(D)-COM can communicate with Excel. Therefore powerful applications using Excel functionality can be designed to communicate with externally developed applications such as 'R' which in turn can be used to create powerful data analysis suites.

2.6 Microsoft® Access and Microsoft® SQL Server

Microsoft® Access was launched in 1992 and was Microsoft's first entry into the database market. Microsoft® Access is a relational database management system which is powered by the relational Microsoft® Jet Database Engine. Recent versions of Access ship as part of Microsoft® Office which also incorporates Microsoft® Excel, as previously introduced. The functionality for biomedical data with Access is a relatively straightforward data management example. With regards to this project and considering the needs of a data analysis platfrom, it can provide a way

of storing Affymetrix array information on an individual array basis, with database rows containing probe expression information. This interaction was shown previously in figure 2.5. The application can also be used to form 'projects' for comparison in a multiple experimental model setting in a user friendly way. For example, the user can specify which models are required to form the analysis comparison experiment. Database tables can be inserted for archiving directly into Microsoft® Access from Microsoft® Excel spreadsheets, which is a particularly useful function as Affymetrix Array results are received in a spreadsheet format produced from the Affymetrix chip normalisation application MAS5.0.

If database technology is required on a larger scale to store larger arrays or larger comparison projects, Microsoft® offer the MS SQL server application. It is preinstalled as an integral component of the Microsoft® Windows Server 2003 operating system. It is also a relational database management system. SQL server is a far more complex database. Unlike Access which has a core Jet database engine, the SQL server database has three parts – an operating system (SQLOS), a relational engine and a protocol layer. By division of 'labour' into different sections the product is more reliable in day to day use. As SQL server is found on server technology platforms, additional issues become apparent which may never be observed in Microsoft® Access. Such issues include concurrency and 'locking' – where multiple users are accessing the same information. Each could potentially change the database and therefore get different results back. SQL server has procedures in place to prevent such an occurrence or to assign who has priority. Such functionality is not as advanced in Microsoft® Access, yet is required for large applications. Additionally, more advanced SQL queries of the data can be implemented in SQL server in comparison to Microsoft® Access, although it is more difficult to use due to the nature of the product.

Microsoft® Access can be remotely commanded using OLE DB connections previously described embedded into a data analysis application as shown previously in Figure 2.3. Microsoft® SQL server can also be commanded using OLE DB; however it is more suited to web service technology due to the scale on which applications are developed although can also use VB.NET function calls.

2.7 Three Dimensional Visualisation Technologies – OpenGL and Direct3D

To facilitate three dimensional (3D) clustering and visualisation of results returned from a developed analysis platform, a set of programming procedures to draw a 3D image are required. Remarkably, there are few 3D application programming language standards – the two main choices are OpenGL and Direct3D (Shreiner et al, 2005) [104]. However they both offer extensive capabilities and as such have diverse implementations ranging from computer gaming to mathematical modelling.

OpenGL is essentially an interface with over 250 function calls based on the 'C' and Visual Basic programming languages, which can be used to create 3D 'scenes' from simple geometric primitives (Shreiner et al, 2005) [104]. It was developed by Silicon Graphics in 1992 and is supported by all operating system incarnations previously described (Shreiner et al, 2005) [104]. Mesa3D is the nearest equivalent to OpenGL but is not identical as OpenGL does have some licensing implications. However Mesa3D coding is fully compatible with OpenGL.

Microsoft was originally part of the OpenGL architecture review board until 1992 when they left the project to pursue their own standard – Direct3D. Although Direct3D was originally less intuitive and more convoluted to programme than OpenGL, Direct3D inherently remains more of a hardware interface technology whereas OpenGL is more a 3D rendering system with hardware acceleration. More support is given to OpenGL and the documentation is more extensive and therefore a logical choice for 3D modelling capabilities of a data analysis platform.

2.8 Affymetrix Microarray Data Analysis Strategy

Figure 2.6 shows an overview of the generalised microarray data analysis process. It shows a typical strategy which can be used to determine significant differentially-expressed genes that could ultimately provide potential biomarkers or targets of interest, for example, in the context of endocrine resistance in breast cancer.

Initial Experimental Design in this instance focuses on Affymetrix (HGU-133A) microarrays The appearance of a typical Affymetrix scan has been shown previously in figure 1.1 from Chapter 1.

When the information has been collated from the Affymetrix scanner using MAS5.0, the significant differentially-expressed probes need to be determined. An initial step is to perform normalisation of the arrays chosen to address a particular experimental model comparison, for example a control group versus other arrayed groups to be compared. Typically there will be three replicates for each experimental group. A particular array experiment compares at least two sets of three individual array replicates. This set of 6 arrays in this combination needs to then be normalised by log base 2 transformation and typically median centring. This allows the whole 'experiment' to be comparable as individual probe intensities can be very broad-ranging with any very large values potentially adding bias to results to the detriment of other less highly expressed probes. As seen in Figure 2.6, the next steps after normalisation are potentially the most informative – differential gene expression determination being particularly important.

As previously introduced, all Affymetrix HGU-133A microarrays contain a very high number or probes – over 23,000. The complete array is too large to assess each probe individually in turn. Most of the probes in a particular group experimental comparison will have no change in expression between a control group, for example, and what the array is compared against. Consequently the unchanged probes can initially be removed ("feature selection") from analysis at this point leaving only the differentially expressed probes. This can be performed in a number of ways. Two main approaches for filtering include, as previously outlined in section 1.3 Chapter 1, ANOVA (Analysis of variance) and SAM (Significant Analysis of Microarray). Literature suggests a false discovery rate of 10% is acceptable (Tusher et al, 2001) [38].

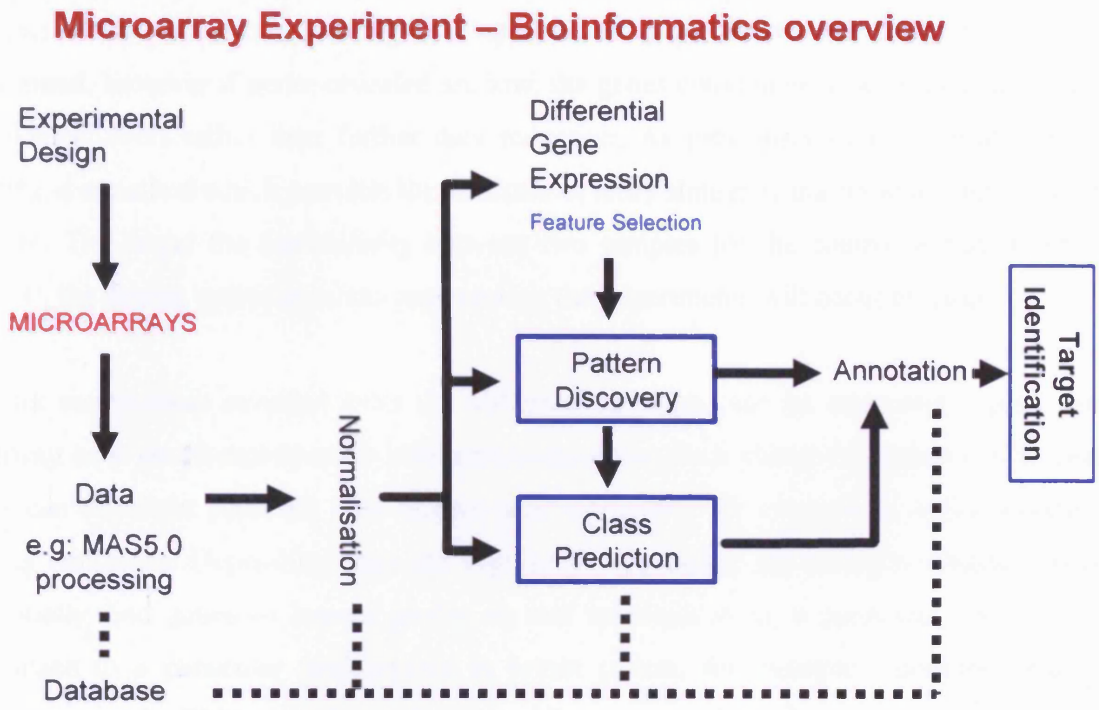


Figure 2.6: Overview of microarray analysis steps regardless of array platform.

Once a data reduction method has been chosen, the significant gene list can be explored by pattern discovery or class prediction. Although shown as two distinct areas in figure 2.6, these aspects closely overlap in reality. The significant gene list, for example 800 probes revealed by SAM analysis, will often firstly be clustered to observe dominant expression patterns in the data using an unsupervised clustering method such as hierarchical clustering as previously outline in section 1.3.8.1, Chapter 1. This shows the relationship between the probes in the significant gene list using their normalised intensity values. Bands of red or green are often evident to indicate regions of similarity of expression profile.

It is good practice to perform many clustering algorithms and compare their results. Alternatives as previously outlined in Chapter 1, include self organising maps (SOM) or partitioning around medoids (PAM). A hierarchical clustering heat map can be used as a guide to determine how many dominant clusters are thought to exist in the dataset. SOM will then force the probes which have most similar profiles into the estimated number of clusters. The effectiveness of this process

will vary based on the number of probes allocated to each cluster i.e. dependent on total cluster number. Multidimensional scaling is a type of class prediction and could be subsequently performed, however if genes revealed are low, the genes could simply be annotated using online ontological tools rather than further data reduction. As previously outlined, multidimensional scaling is a method which converts the structure of array similarity matrix to a simple geometrical picture. The larger the dissimilarity between two samples (of the control versus the treatment arrays), the further apart the points representing the experiments will occur in space.

Cluster membership revealed from the different algorithms can be annotated – gene function, pathway or even disease-specific information regarding each cluster of probes can be revealed. This can highlight potential new targets/gene signatures, for example in anti-hormone breast cancer resistance. Depending upon the experimental question, the biologist/clinician could also potentially find genes of known profile in, and relationship to, a particular outcome such as resistnace to a particular antihormone in breast cancer, for example resistance amongst the clustering results. This not only validates the effectiveness of the clustering techniques (since out of 24,000 probes (23K) initially, one or two established landmarks have been accurately revealed) yet also produces the possibility of identifying new targets as a result of sharing a similar or dissimilar profile to a known landmark gene.

2.9 Technology Selection

After careful evaluation of technologies available to facilitate development of a microarray analysis platform the following technologies were used for system development:

1. Microsoft® Windows to develop the application – Microsoft® Visual Basic
2. Mathematical processing engine to calculate results – 'R' scripting language - selection of appropriate components (e.g. for feature selection, pattern discovery, class prediction and annotation) to interlink within the developed software.
3. Microsoft® Excel to create a workspace to view and collate results.
4. Microsoft® Access for data storage to retrieve and record analysis results.
5. DCOM and ODBC connectivity to communicate between application areas.
6. Open GL scripting language to provide 3D graphical capabilities.

Using these six separate yet unique components, a comprehensive high throughput analysis system was developed.

2.10 Overview of Informatics Tenovus – 'I-10'

2.10.1 The rationale for I-10:

I-10 has been developed to harness the symbiotic relationships between 'R', Excel and D-COM as described previously, enabling I-10 to perform powerful microarray analysis in a user-friendly manner. Figure 2.7 shows this relationship in terms of how each part of this system interacts with each other, where importantly I-10 has been developed such that the user is removed from all knowledge of how this system works.

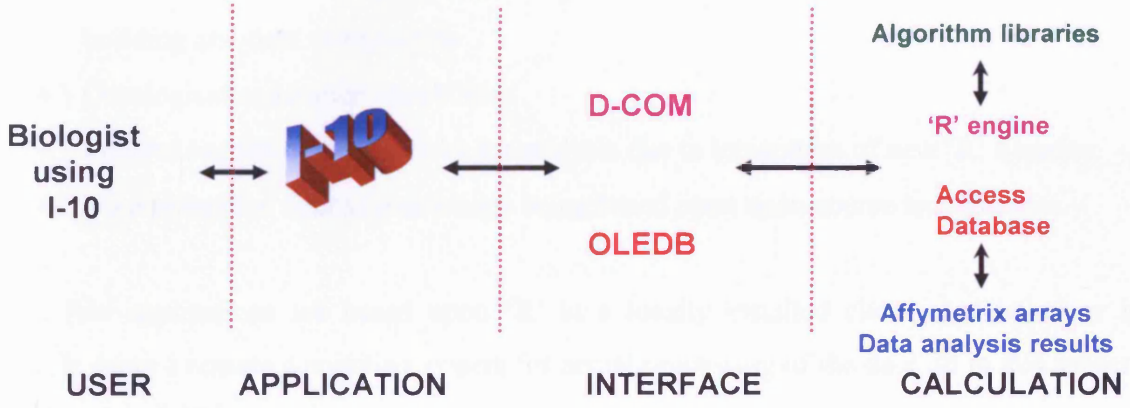


Figure 2.7: Relationship between the user and the components of I-10. Note from the previous technological evaluation section how difference interface protocols – D-COM and OLEDB are used to communicate with the separate parts of the application – 'R' (shown in purple) communicates in turn to installed algorithm libraries for clustering (shown in green) and the Access database (shown in red) retrieves Affymetrix array information and stores data analysis results (show in blue).

The Tenovus research group had previous extensive experience with the commercial application Genesifter software from Vizxlabs which has been used to analyse endocrine response/resistance microarray data. Although Genesifter itself has evolved slightly over the course of the project, the

tool still has missing capabilities notably in its capabilities for advanced statistical and graphically-based clustering, where these have been developed within I-10 particularly from the 3D graphical point of view. An initial goal of the project was thus to encompass within I-10 all the capabilities of Genesifter yet go beyond the features currently available to the Tenovus group. However, in practice the developed I-10 software evolved to have the following abilities that fulfil much broader functionality requirements for the microarray analysis community as overviewed in figure 2.8:

- A user friendly, graphical menu driven interface through Microsoft Windows environment
- Speed and Versatility
- Microarray data analysis
- Basic and advanced statistical analysis
- Multiple unsupervised and supervised clustering facilities
- 2D/3D displays to enhance diverse aspects of data exploration, statistical analysis, model building and data visualisation
- Ontological/annotation capabilities
- Endless expansion possibilities for analysis due to integration of new 'R' libraries.
- No commercial license restrictions being based upon open-source technologies.

Very few applications are based upon 'R' in a locally installed client application or indeed hosting using a remote computing system for actual processing of the data, so in this aspect of its development I-10 is unique.

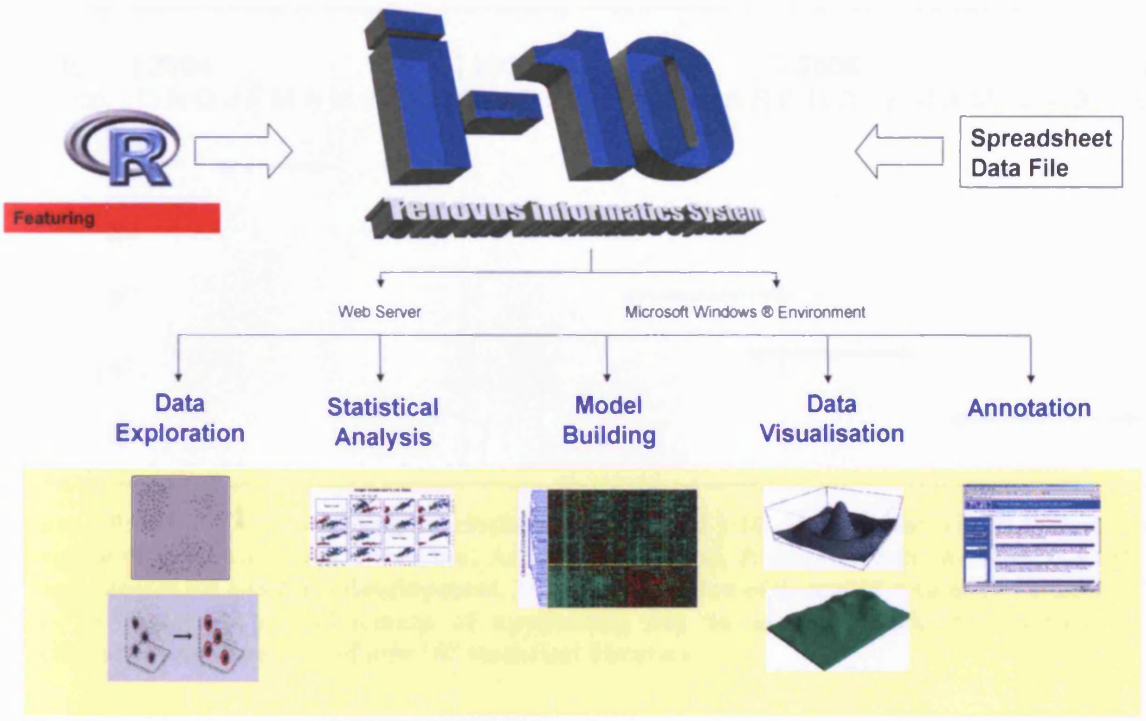


Figure 2.8: Functionality requirements, and overview of I-10 as either a standalone application or future web deliverable.

2.10.2 Design of I-10

For any new system to be successful, ease of use is likely to be a primary consideration from the very beginning of development. When a system is first beta tested, inevitably some changes are required; however if they can be minimised during early development it will give the user a better idea of what is possibly and hopefully generate new ideas. Figure 2.9 outlines the development schedule of I-10.

When developing the Windows application for I-10, the concept of using multiple windows each representing the different analyses was found to be a user-friendly feature that would be highly-desirable to incorporate into I-10, akin to other Microsoft Windows applications. This has been successfully achieved in I-10, where using this approach the user only has to focus on an option for any given analysis step, be it viewing the query generated in Access upon which data will be analysed or choosing hierarchical clustering, for example.

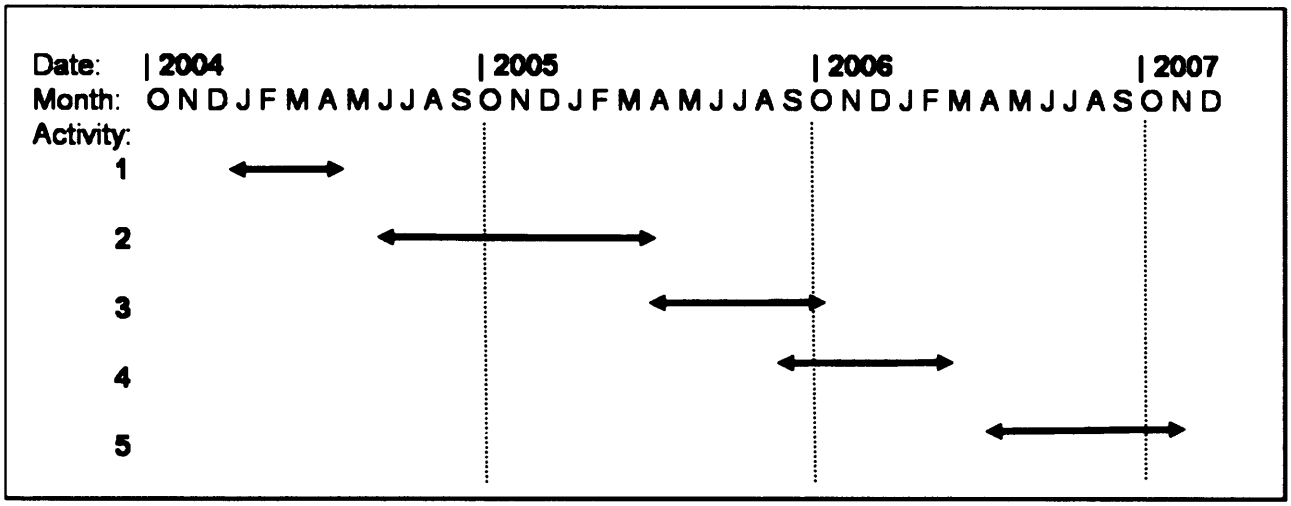


Figure 2.9: Gantt chart outlining development schedule of I-10. Month abbreviated to first letter of corresponding month. e.g: O: October, A: April or August, J: June or July. Activity 1: Evaluation of technologies for platform development. 2: Implementation of design of system. 3: Initial trial with users in Tenovus. 4: Refinement of application due to user feedback. 5: Addition of new functionality due to release of new 'R' statistical libraries.

Results from the multitude of options can be minimised within the application for instant comparisons. Moreover, I-10 also utilises a user-friendly "flow-chart" approach for its main menu that the user accesses to perform the various steps in microarray analysis. Early discussions regarding menu design with users utilised a flow chart outlining all steps from start to finish for microarray analysis as shown previously in figure 2.6. This model proved very easy to understand, and so was mirrored directly within the application, presenting selectable options at each step clearly to the user, as seen in the final software main menu as shown in figure 2.10.

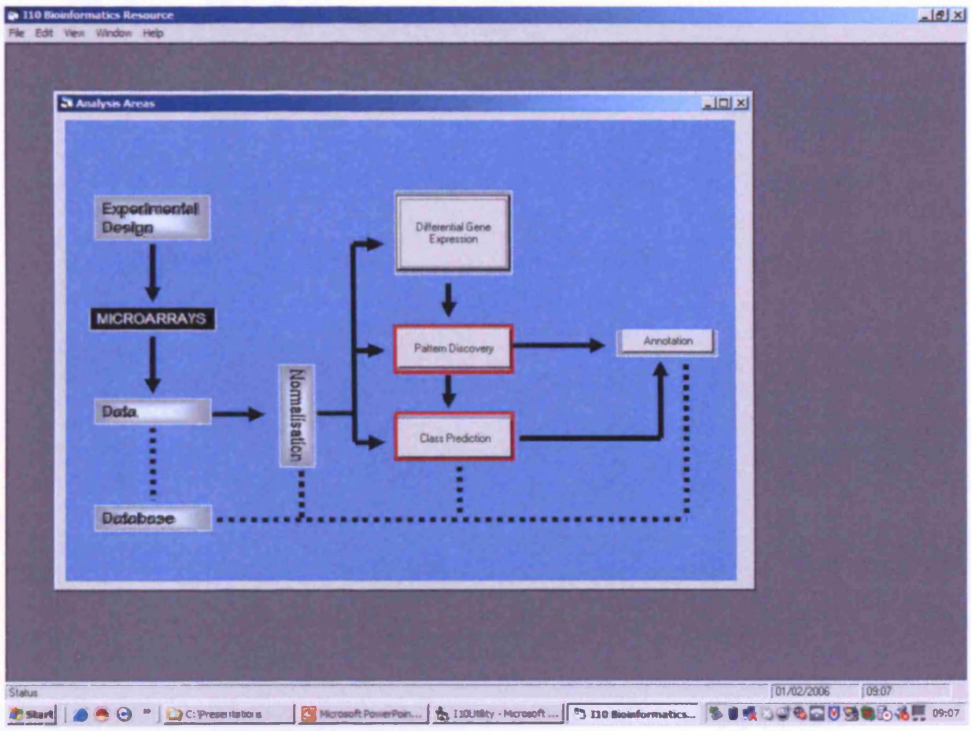


Figure 2.10: Main Menu of I-10 based on a flow-chart microarray analysis approach.

There is a wealth of analysis options available to the user through I-10. They range from hierarchical clustering through to multi-dimensional scaling, fuzzy analysis, PAM, to name only a few of the available options. Classical 2D representations of clustering can now be displayed in 3D where appropriate, where these latter capabilities are generated by calling the OpenGL API, a fundamental improvement over the capabilities of many existing analysis microarray analysis packages. Examples of 2D and 3D comparisons of the same data in I-10 can be seen in figures 2.11 and 2.12. It is also possible for the user to interact with the 3D visualisations (for example, rotating the display, clicking on its components).

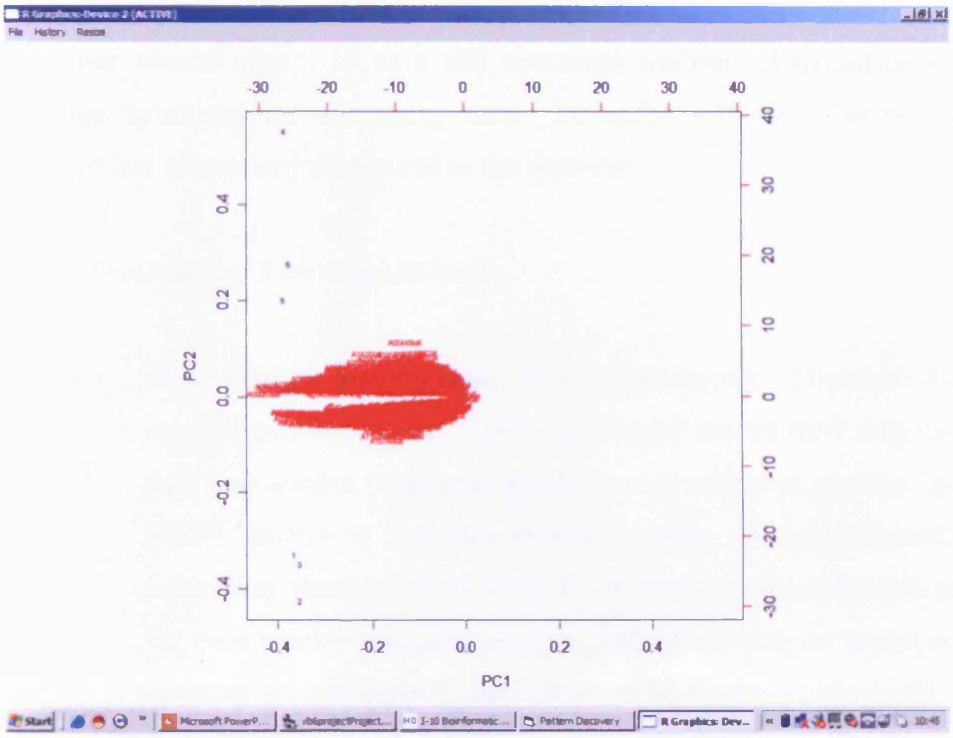


Figure 2.11: 2D plot: Principal Components Analysis in I-10

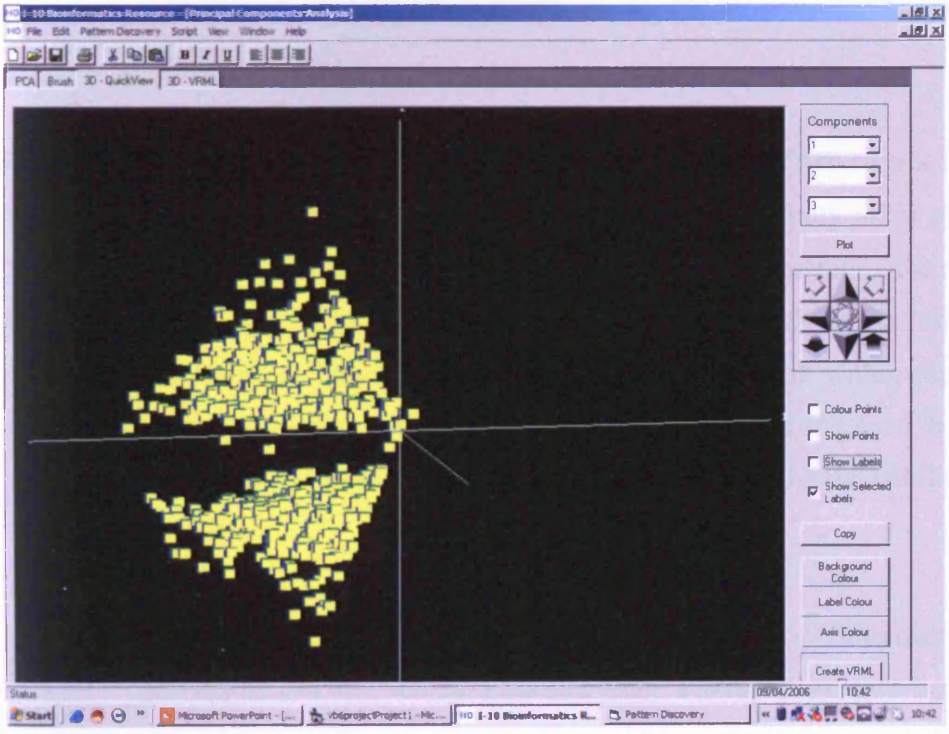


Figure 2.12: 3D plot: Principal Components Analysis in I-10 showing controls to the right of the figure where the plot can be rotated and certain points selected or deselected.

The user should view I-10 as a self contained analysis, visualisation and data management package optimised for microarray data. However, I-10 can also be used to visualise any multivariate biomedical data stored in the database.

2.10.3 Overview of I-10 capabilities:

(i) **Data storage management:** In conjunction with Microsoft Access, I-10 gives the user the power to store, search and display any of their Affymetrix Microarray data. Any new queries (searches) which the user wishes to perform, for example comparing MCF7 control vs Faslodex resistant arrays, can be requested and the appropriate microarray data delivered ready for immediate data reduction using SAM. Likewise for even quicker analysis, previous SAM output can be stored, and re-called from the database for subsequent visualisation within I-10, chosen in this instance specifically to address the needs of Tenovus yet also of obvious value to the wider research community who requires access to advanced analysis techniques within a user-friendly context.

In the Tenovus group, MAS 5.0 results are returned from the Affymetrix facility as set of CD-ROMS, however subsequent storage in a dedicated database only takes place currently within GeneSifter. However storage is limited by cost issues. When generating the database integral to I-10, the column naming conventions returned from the Affymetrix microarray application MAS5.0 were rather ambiguous, and working with researchers within the Tenovus group has allowed more informative naming conventions to be developed and used in the Access database. Migration to such a database storage system has improved the availability of what is a very valuable data resource.

(ii) **Excel to check data pre-analysis:** Using elements of Microsoft Excel, the user can quickly check the returned results from Microsoft Access before continuing with differential gene expression analysis. Although not an essential step, it confirms the array combinations to be analysed are correct.

As previously outlined, most Microsoft Windows users are very familiar with the interface that Excel provides, where scientists may already be using Excel to perform simple statistical analysis such as a t-test. Therefore it was logical for I-10 to utilize Excel's spreadsheet functionality rather than an alternative as it allows simple visualisation of what the database has returned for subsequent analysis as shown in figure 2.13. Furthermore, an Excel 'sheet' view automatically gives I-10 added formatting features which are integral to Excel such as sorting by Ascending/Descending, Formatting, colouring particular rows, and many more useful features the biologist may want to aid in interpreting the raw data. Printing and saving of the data in an Excel format – for example significant gene lists- can also be performed within I-10 for re-insertion into the database for storage or for discussion in research group meetings.

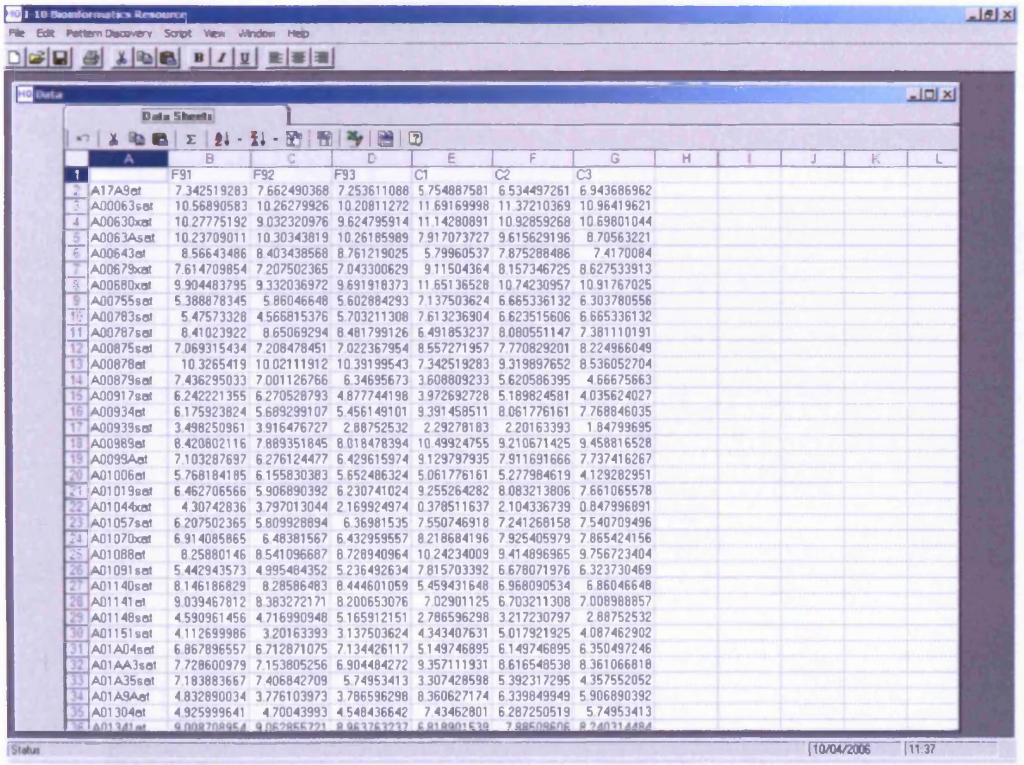


Figure 2.13: Overview of the Microsoft Excel spreadsheet interface to view data generated from the database subsequent to an initial query for analysis.

(iii) **Differential Gene Expression:** SAM can be performed within I-10. All results created can be saved using the Access database so they can be re-called for future pattern exploration in I-10.

As previously outlined in detail, a fundamental goal of any microarray experiment is to identify genes with significant differences between groups in any given array experiment. The large numbers of probes present on Affymetrix arrays thus need to be initially reduced to a core significant list prior to any subsequent analysis. One of the most popular ways of filtering for differentially expressed genes has been according to fold change in expression. However this has been proven unreliable in a number of studies as there is no information to back up what is a true change and what is occurring due to random variation (Mariani et al, 2003) [105]. This process can be influenced by inherent replicate variation, a feature that is addressable by initial MVA plot assessment (performed prior to sample addition to the Access database). Subsequent statistical testing of gene lists can then provide more confidence in identifying differentially-expressed genes. Self organising maps functionality can also be found under this section. The SAM option available to the user can be accessed in the I-10 menu from the differential gene Expression methods box (figure 2.14).

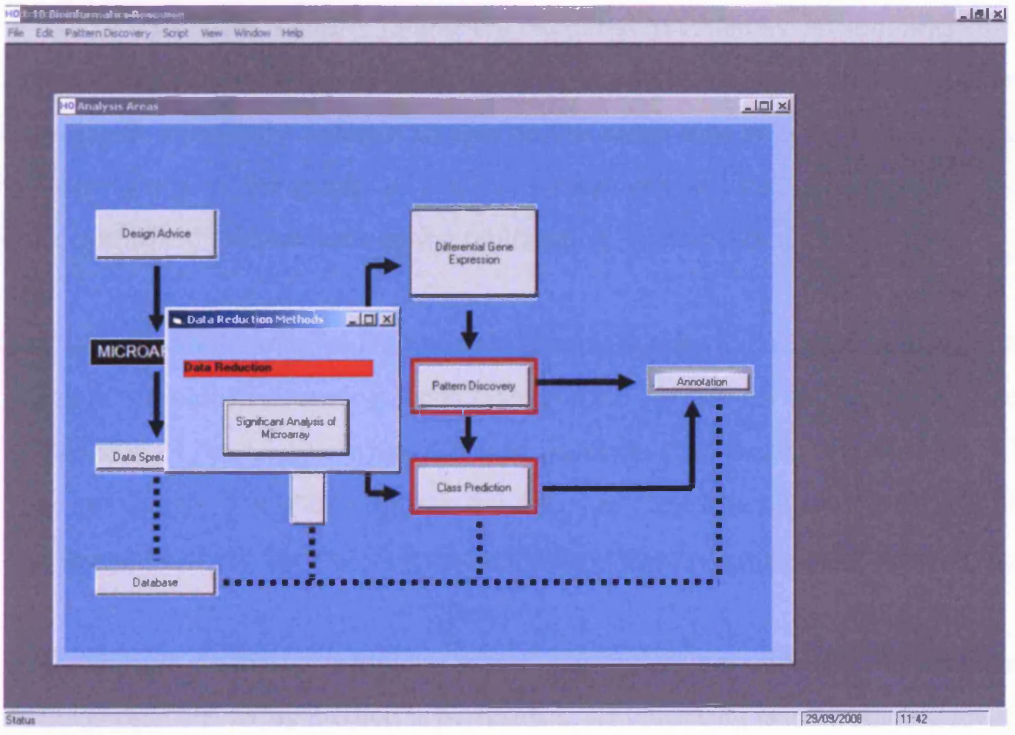


Figure 2.14: The available Differential Gene Expression Methods from the I-10 menu.

(iv) **Pattern discovery:** One of the first clustering routines commonly performed on a dataset is hierarchical clustering (HCA) with visualisation using a heat map representation. This is available in I-10 within the pattern discovery options. Although often difficult to interpret, especially with large significant gene lists and complex patterns, the key aspect of HCA is to give an indication as to the number of dominant 'groupings' within the gene profiles. In I-10, more advanced clustering techniques are available such as PAM and K-Means. However the HCA steps of analysis are highly recommended as these latter approaches require an indication of the total number of groupings in the data which can be determined using hierarchical clustering. The novel aspect of I-10 to aid gene discovery is the 3D representations for these functions, as opposed to the more usual 2D plots which 'R' can generate. As a consequence of the 'z' dimension axis in 3D scatter plots, the user can rotate the plot in 3D which will reveal hidden clusters or patterns not as obvious as observed with a 2D representation.

An overview of the pattern discovery options can be seen in figure 2.15. Hierarchical clustering, as well as being useful towards detailed pattern discovery as described above, can also be used as a classical indicator to further confirm (alongside the initial MVA analysis) similarity of arrays of a particular set of replicates, especially when using a reduced list of genes. If individual replicates are associating with completely different sample groups rather than with their own replicates, this is an early warning sign of the quality of a particular replicate and that subsequent analysis steps should not be performed without considering replacement with an alternative sample.

Ideally the situation should arise where multiple advanced pattern discovery compare clustering results. Multiple individual clustering techniques can be performed within the I-10 software; moreover, inclusion of the relatively new PVclust packages ("automated cluster comparison" in the menu) as previously described removes the decision regarding clustering choices from the user by comparing the clustering results from the multiple tests and presenting these to the user.

The 3D plots of each clustering technique where appropriate are accessed via a tabbed menu when analysis of each clustering method is performed. 3D clustering is not available for HCA.

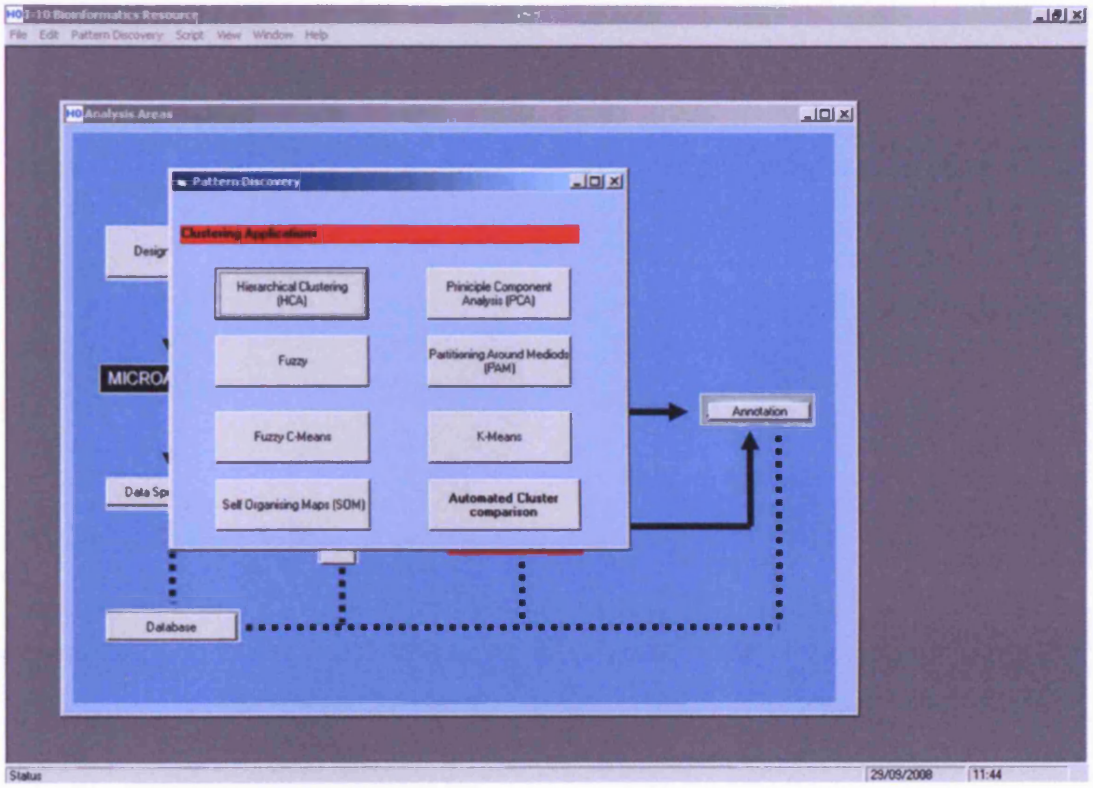


Figure 2.15: Clustering methods available under Pattern Discovery within I-10.

(v) **Class Prediction:** refers to the assignment of particular genes into previously identified classes, as previously outlined. This can be performed by ontological searching and is one aspect which ClValid module in I-10 harnesses.

Class prediction can be considered as an optional step in microarray analysis. It will depend upon what classifiers exist for a given data set and depend upon how many genes have been revealed and what expectations were set regarding the data. Figure 2.16 shows an overview of options available for class prediction detailing PCA, MDS and a link to online classification tools such as DAVID and Babelomics, previously described in Chapter 1.

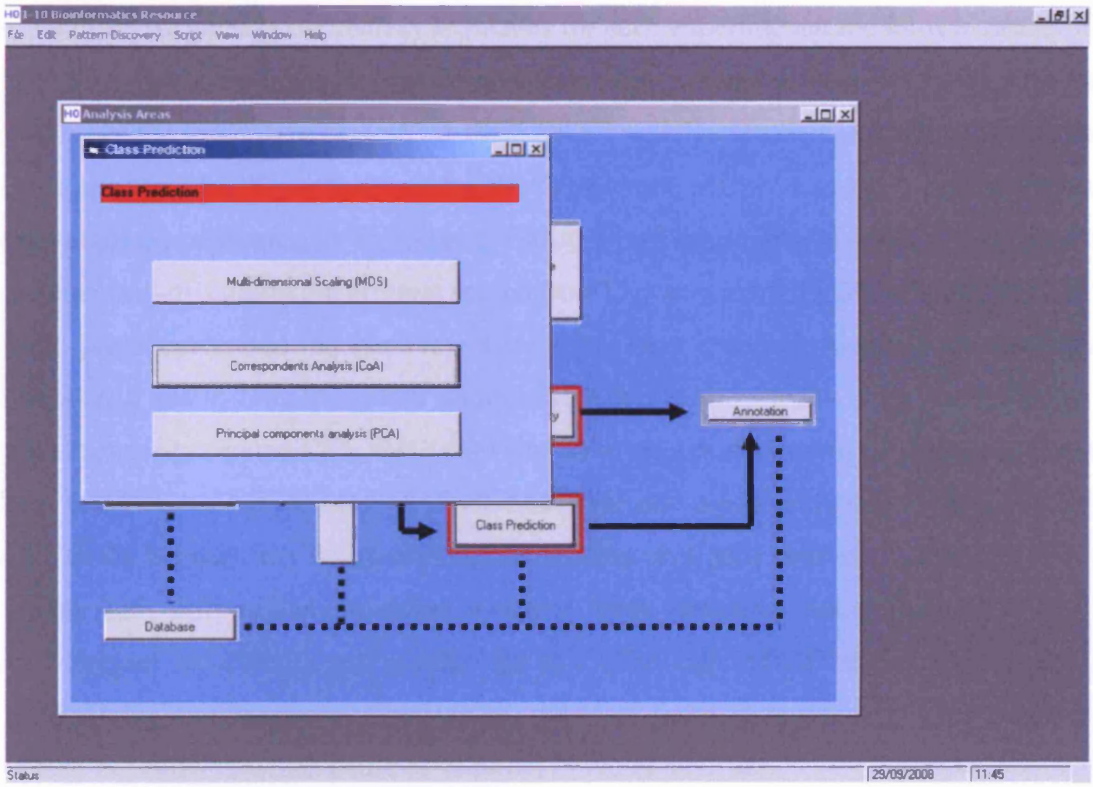


Figure 2.16: Options available for class Prediction within I-10

2.10.4 Database development

Microsoft Access databases can be created using the inbuilt graphical user interface, akin to other Windows applications. Initially a blank Microsoft Access database is created and for the purposes of I-10 this was given the name 'TenovusAffy.mdb' where *.mdb is the file type used by Access for storing database information. One of the requirements of the Tenovus group was to not only analyse data in I-10, however to also be able to export datasets in an interchangeable format for associated research collaborations. Before the development of I-10 there was no easy way to extract raw information from Genesifter once it had been uploaded. Microsoft Access enables this export function, due to the flexible way queries can be structured and results returned as shown in figure 2.17.

Each of the three Affymetrix microarray replicates for each experimental model was stored in an individual table with an appropriate title for added flexibility. Tenovus have extensively invested in their Affymetrix arrays covering all of their MCF7 cell line models, including the parental endocrine responsive MCF7 cells (termed here "Resistance MCF-7 Control") and related sub-lines with acquired resistance to Faslodex ("FASR") or Tamoxifen ("TAMR"). The database tables were created by editing the original spreadsheet files generated by the Affymetrix MAS5.0 application. These comprised the same raw dataset that have been uploaded and are undergoing normalisaion and analysis by the group within Genesifter. However raw chip normalisation of Affymetrix scanner generated CEL files could also have been performed in 'R' before addition to the Access database instead of using MAS5.0. Each replicate was stored using Affymetrix ID for each probe as the primary key which allows comparisons on a gene level with other arrays in the query, along with information on signal intensity level, detection call assigned by MAS5.0 (Present, Marginal or Absent) and p-value for this. Although AffyID and the corresponding signal level are the most important for analysis, the other fields were retained in the database so as not to lose information which could be required in future analyses.

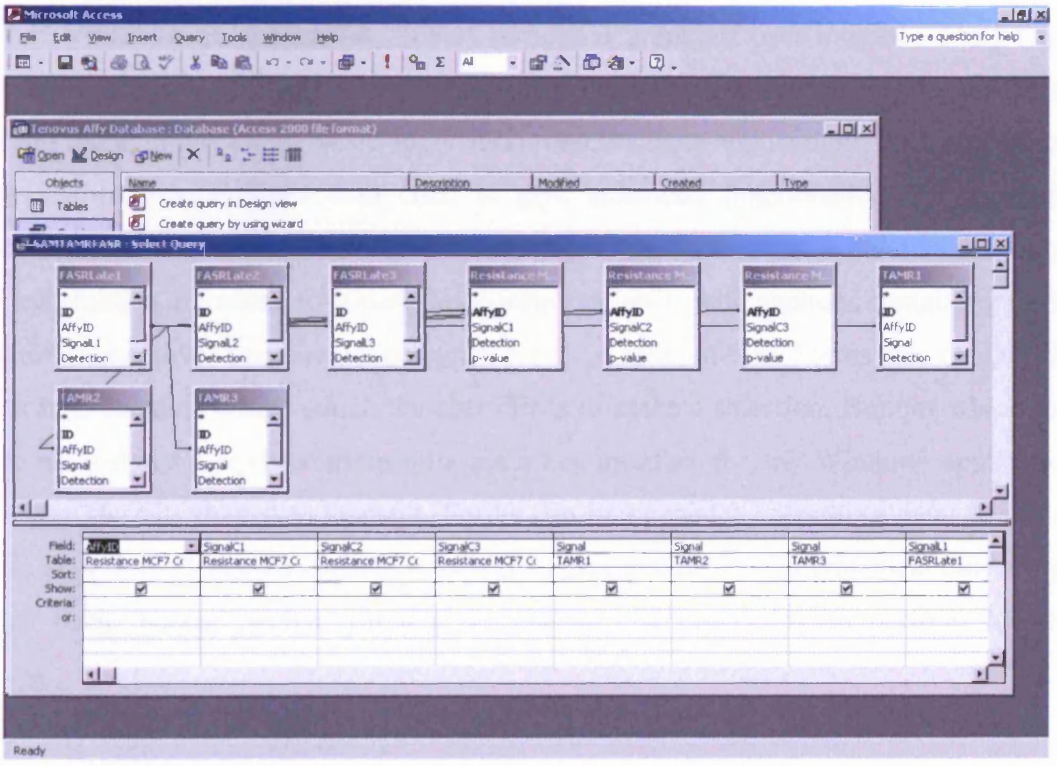


Figure 2.17 – Generating a model comparison from the Microsoft Access database of Affymetrix microarray models for data analysis.

The query (comparison of models) is visible in I-10 by saving the query as 'I10Q'. This automatically results in the Excel database sheet view in I-10 'seeing' the project created and ready for subsequent analysis. However if normalisation has already been performed, existing pre-analysed datasets can be recalled from the database for immediate analysis by changing the query returned so it can be 'seen' in I10.

2.11 I-10 Coding Development

2.11.1 Visual Basic general principles

Visual basic is used to produce the interface which the user interacts with I-10. It also binds together and commands 'R' and the Access database to perform their respective functions when required.

Coding of Visual Basic applications occurs through a graphical user interface using forms, as previously introduced. Although main components to generate applications can be chosen from a component menu in the Visual Basic application used to create applications, each component can then be further supplemented with code to give enhanced functionality. For example, the development of I-10 involved adding 'R' scripting to perform a particular function. Items of textual information are added to Visual Basic forms as labels with captions containing the text to be displayed – however they are not designed for large areas of text. Forms also contain choices which in turn contain buttons which the user clicks to make a selection. Buttons which the user clicks to run a statistical procedure in I-10, are a key interface for any Windows application and Visual Basic permits their development. Forms can be opened by attaching them to buttons as shown in code 2.1. Clicking Command1 (which represents the label hierarchical clustering) will open the form 'hcacat' which contains the coding and processing to perform hierarchical clustering.

```
Private Sub Command1_Click()
Dim frmBP As hcacat
    Set frmBP = New hcacat
    frmBP.Show
End Sub
```

Code 2.1: Inserting a button onto a form in Visual Basic. Coding to show initialisation of the button and the form to show - hierarchical clustering in this example. The private 'sub' statement informs visual basic of the start of a function and 'end' sub the end of code for the function.

Each form can be coded to include any function call to 'R', and also to manipulate data loaded into I-10 from the database. This versatility allows I-10 to be infinitely upgradeable to feature the most up to date 'R' libraries for data analysis. Indeed, I-10 continually evolved in this manner as the project progressed, with the latest functionality included being the clValid module which compares clustering method performance.

2.11.2 Syntax alterations from Visual Basic to 'R'

At the start of each form, the connection to the R-(D)COM interface to 'R' needs to be initiated from Visual Basic as shown in code 2.2

```
Dim sconn As StatConnector
Set sconn = New StatConnector
sconn.Init "R"
```

Code 2.2: Initialising a connection to 'R' via the R-(D)COM interface.

Commands to 'R' are passed as strings of text from Visual Basic. Early in the development of I-10 it was clear that certain 'escape' characters are also required to build the syntax which 'R' requires in its commands. An example character is the quotation mark. These are characters which cannot be passed via an interface as they have special meaning within the computer system. However they can 'protected' using the ASCII code Chr(34) which represents the quotation mark. Commands sent directly to 'R' are always processed using the 'sconn.EvaluateNoReturn' command. This will become apparent throughout the following pages.

2.11.3 Excel sheet component data handling

When data has been imported into Excel from Access using an OLEDB interface to Visual Basic, the worksheet needs to be sent to 'R' for processing. This is performed using various steps before analysis using 'R' libraries can begin. As shown in code 2.3, the datasheet form in I-10 is called, and that all rows and columns are looped through and stored in 'dat' until complete. The labels of the headings from the first row of the Excel spreadsheet and first column are stored and sent to 'R'.

```
Set dat = sp1.ActiveSheet

i = 2
Do Until dat.cells(1, i) = ""
i = i + 1
Loop
cl = i - 1
i = 2
Do Until dat.cells(i, 1) = ""
i = i + 1
Loop
rw = i - 1
cexval = 0.1

  Dim R1(10, 5) As Double
  Dim r6 As Variant
  vars = rw

cname = Chr(34) & "labs" & Chr(34)

    tmp = "labs<-c("
    For j = 2 To cl
        r5 = Chr(34) & "i_" & dat.cells(1, j) & Chr(34)
        If j < cl Then tmp = tmp & r5 & ","
        If j = cl Then tmp = tmp & r5
    Next j
    tmp = tmp & ")"
    Text1.Text = tmp
    sconn.EvaluateNoReturn (tmp)

'create data matrix
    For i = 2 To vars 'mf1.LastCol
        cname = Chr(34) & "i_" & dat.cells(i, 1) & Chr(34)
        tmp = cname & "<-c("
            For j = 2 To cl
                r2 = dat.cells(i, j)
                If j < cl Then tmp = tmp & r2 & ","
                If j = cl Then tmp = tmp & r2
            Next j
        tmp = tmp & ")"
        sconn.EvaluateNoReturn (tmp)
  Next i

  tmp2 = "Car<-data.frame(labs," '
    For i = 2 To vars
        If i < vars Then tmp2 = tmp2 & "i_" & dat.cells(i, 1) & ","
        If i = vars Then tmp2 = tmp2 & "i_" & dat.cells(i, 1)
    Next i
       'tmp2 = tmp2 & "row.names = " & Chr(34) & 1 & Chr(34) & ")"
       tmp2 = tmp2 & ")"
       sconn.EvaluateNoReturn (tmp2)
      sconn.EvaluateNoReturn ("row.names(Car) = labs")
```

Code 2.3: Transfering spreadsheet data from the Excel component of I-10 to 'R'.

The dataset is transferred to 'R' (to create a data matrix) using the R-(D)COM interface as previously described.

2.11.4 Profile viewer

A popular feature with users of I-10, also present in Genesifter, is a profile viewer that allows display of expression of a particular gene across the experimental arms of interest. Using part of the Excel sheet functionality in I-10, individual probes can be analysed with the profile viewer. The individual spreadsheet cell containing the value of normalised intensity of the probe changes colour according to degree of increased or decreased expression level. For this to be effective and meaningful, the experimental arms to be compared are viewed in relation to a control arm which remains black throughout. The code to allow the colour ranging for the profile viewer is shown in code 2.4.

```
Sub ColourCells()

Dim tCell As Object
Const cBlack As Byte = 1
Const cWhite As Byte = 2
Const cRed As Byte = 3
Const cGreen As Byte = 4
Const cDRed As Byte = 5
Const cDGreen As Byte = 6
Dim fColour
Dim bColour
For Each tCell In ActiveSheet.UsedRange.Cells
If Not IsNumeric(tCell.Value) Then GoTo NextOne
Select Case tCell.Value
Case -12 To -2
fColour = cBlack
bColour = cGreen
Case -2 To -0.25
fColour = cWhite
bColour = cDGreen
Case -0.25 To 0.25
fColour = cWhite
bColour = cBlack
Case 0.25 To 2
fColour = cWhite
bColour = cDRed
Case 2 To 12
fColour = cWhite
bColour = cRed
End Select
tCell.Font.ColorIndex = fColour
tCell.Interior.ColorIndex = bColour
tCell.Interior.Pattern = xlSolid
NextOne:
Next tCell

End Sub
```

Code 2.4: Profile viewer. Code defining the colour to change a cell corresponding to a particular range of normalised intensity value to form the profile viewer in I-10

2.11.5 Three-dimensional plotting using OpenGL

Using OpenGL, clustering results resulting from 'R' scripts for pattern discovery can be visualised in 3D. The fundamental method calls crucial for operation are detailed in code 2.5. 2D plots are returned directly for the 'R' programming environment via the R-(D)COM interface and displayed in a separate window which is opened by 'R'.

```
EnableOpenGL glView1.hdc
 glShadeModel smSmooth                                  ' Smooth Shading
 glEnable glcDepthTest                                  ' Depth Testing
 glDepthFunc cfLEqual                                   ' The Type Of Depth Test
 glHint htPerspectiveCorrectionHint, hmNicest           ' Adds Perspective Calculations

   'Set the light settings
  Dim aflLightAmbient(4) As GLfloat
  Dim aflLightDiffuse(4) As GLfloat
  Dim aflLightPosition(4) As GLfloat

   'Ambient settings
   aflLightAmbient(0) = 1
   aflLightAmbient(1) = 0.5
   aflLightAmbient(2) = 0.5
   aflLightAmbient(3) = 1#

'Diffuse settings
   aflLightDiffuse(0) = 1#
   aflLightDiffuse(1) = 1#
   aflLightDiffuse(2) = 1#
   aflLightDiffuse(3) = 1#
  'Position settings
   aflLightPosition(0) = 0#
   aflLightPosition(1) = 0#
   aflLightPosition(2) = 2#
   aflLightPosition(3) = 1#

  'Set up light in OpenGL and the direction (1pm)
   glLightfv ltLight1, lpmAmbient, aflLightAmbient(0)
   glLightfv ltLight1, lpmDiffuse, aflLightDiffuse(0)
   glLightfv ltLight1, lpmPosition, aflLightPosition(0)

  'Enable the light
   glEnable glcLight1
 BuildFont Me

glClearColor Red, Green, Blue, 0

    ReDim mat(rw + 2, rw + 2)
    ReDim lab(rw + 2)


    maxx = 0: maxy = 0: maxz = 0

    For i = 2 To rw + 1
        If Abs(maxx) < Abs(mf2.cells(i, 2)) Then maxx = Abs(mf2.cells(i, 2))
        If Abs(maxy) < Abs(mf2.cells(i, 3)) Then maxy = Abs(mf2.cells(i, 3))
        If Abs(maxz) < Abs(mf2.cells(i, 4)) Then maxz = Abs(mf2.cells(i, 4))
    Next i
```

…Code 2.5 continued overleaf

```
...Code 2.5 continued from previous page

    For i = 2 To rw + 1
        mat(i - 1, 1) = (mf2.cells(i, 2) / maxx) * 0.5
        mat(i - 1, 2) = (mf2.cells(i, 3) / maxy) * 0.5
        mat(i - 1, 3) = (mf2.cells(i, 4) / maxz) * 0.5
        lab(i - 1) = mf2.cells(i, 1)

 sp3.cells(i - 1, 1) = mf2.cells(i, 1)
    Next i

    Me.Show

    glMatrixMode mmProjection        ' Select The Projection Matrix
    glLoadIdentity                   ' Reset The Projection Matrix

    glMatrixMode mmModelView         ' Select The Modelview Matrix
    glLoadIdentity                   ' Reset The Modelview Matrix


    Do
        DoEvents

        If (Not DrawGLScene Or Keys(vbKeyEscape)) Then
         Unload frm
        Else
            SwapBuffers (glView1.hdc)
                DoEvents
        End If
    Loop
```

Code 2.5: Initialising plotting function in OpenGL. Code highlighting the plotting characteristics of the plot and also the datasource to use to draw the 3D plot using results returned from 'R'.

The glView1 method has several functions. Each function is called successively to enable different aspects of functionality. To enable the user to use the keyboard to spin and turn the 3D scatter plot which is created instead of the mouse, the functions shown in code 2.6 is required.

```
If KeyCode = vbKeyRight Then       ' If Right Arrow Pressed
      yrot = yrot - 1.5            ' Rotate The Scene To The Left by 1.5
      End If
      If KeyCode = vbKeyLeft Then  ' If Left Arrow Pressed
      yrot = yrot + 1.5         ' Rotate The Scene To The Right by 1.5
End If
    If KeyCode = vbKeyPageDown Then  ' If Up Arrow Pressed
        scaler = scaler + 0.2
End If
            If KeyCode = vbKeyPageUp Then    ' If Down Arrow Pressed
               scaler = scaler - 0.2
            End If

            If KeyCode = vbKeyUp Then      ' If pagedown key being pressed
            If lookupdown = 50 Then        ' If already up 50 degrees no action
            lookupdown = lookupdown
                Else
             lookupdown = lookupdown + 1   ' if less than 50 keep adding 1
                End If
            End If

            If KeyCode = vbKeyDown Then    ' Is the pageup key being pressed
            If lookupdown = -50 Then  ' If looking down 50 degrees no action
            lookupdown = lookupdown
                Else
                    lookupdown = lookupdown - 1 'if more than 50 keep subtract 1
                End If
            End If

            If KeyCode = vbKeyL And Not lp Then  ' L Key Being Pressed Not Held
                lp = True                        ' lp Becomes TRUE
                light = Not light                ' Toggle Light TRUE/FALSE
                If Not light Then                ' If Not Light
                    glDisable glcLighting        ' Disable Lighting
                Else                             ' Otherwise
                    glEnable glcLighting         ' Enable Lighting
                End If
            End If

            If Not KeyCode = vbKeyL Then         ' Has L Key Been Released?
                lp = False                       ' If So, lp Becomes FALSE
            End If

            If KeyCode = vbKeyF And Not fp Then  ' Is F Key Being Pressed?
                fp = True                        ' fp Becomes TRUE
                mFilter = mFilter + 1            ' filter Value Increases By One
                If (mFilter > 2) Then            ' Is Value Greater Than 2?
                    mFilter = 0                  ' If So, Set filter To 0
                End If
            End If
DrawGLScene
```

Code 2.6: Keyboard control for 3D plot rotation. Code required enabling the user to rotate the 3D model using the keyboard instead of the onscreen buttons which can be chosen using the mouse.

The flexibility of OpenGL allows the user to specify the direction of light and perspective for the model drawn to enhance viewing. Once the direction of 'light' shone against the model is created, the data to be plotted can be loaded together with the three dimensional axis.

```
Public Function DrawGLScene() As Boolean

    glClear clrColorBufferBit Or clrDepthBufferBit  ' Clear Screen/Depth Buffer
    glLoadIdentity                                  ' Reset The Current Matrix

      Red = backcol And &HFF&
    Green = (backcol And &HFF00&) \ &H100&
    Blue = (backcol And &HFF0000) \ &H10000

    glClearColor Red, Green, Blue, 0


    Dim x_m As GLfloat
    Dim y_m As GLfloat
    Dim z_m As GLfloat
    Dim u_m As GLfloat
    Dim v_m As GLfloat        ' Floating Point For Temp X, Y, Z, U And V Vertices
    Dim xtrans As GLfloat
    Dim ztrans As GLfloat
    Dim ytrans As GLfloat
    Dim sceneroty As GLfloat
    Dim scenerotx As GLfloat

    xtrans = -xpos              ' Used For Player Translation On The X Axis
    ztrans = -zpos              ' Used For Player Translation On The Z Axis
    ytrans = -walkbias - 0.25   ' Used For Bouncing Motion Up And Down
    sceneroty = 360# - yrot     ' 360 Degree Angle For Player Direction
    scenerotx = 360# - xrot

    Dim numtriangles As Integer ' Integer To Hold The Number Of Triangles

    glMatrixMode GL_PROJECTION
    glLoadIdentity
    glMatrixMode GL_MODELVIEW
    glLoadIdentity
    glScalef scaler, scaler, scaler

    glRotatef lookupdown, 1, 0#, 0#     ' Rotate Up And Down To Look Up And Down
    glRotatef sceneroty, 0#, 1, 0#      ' Rotate Depending On Direction of object
    glRotatef scenerotx, 1, 0#, 0#      ' Rotate Depending On Direction of object

    Red = axiscol And &HFF&
    Green = (axiscol And &HFF00&) \ &H100&
    Blue = (axiscol And &HFF0000) \ &H10000
    glColor3f Red, Green, Blue
```

…Code 2.7 continued overleaf

```
…Code 2.7 continued from previous page

'draw x axis
    glRasterPos3f 1, 0, 0
    axlab = " 1 "
    glPrint axlab                ' Print GL Text To The Screen
    glBegin bmLineStrip
    glVertex3f 0#, 0#, 0#
    glVertex3f 0.7, 0#, 0#
    glEnd

    glBegin bmLineStrip
    glVertex3f 0#, 0#, 0#
    glVertex3f -0.7, 0#, 0#
    glEnd

'draw y axis

    glRasterPos3f 0, 1, 0
    axlab = " 2 "
    glPrint axlab               ' Print GL Text To The Screen
    glBegin bmLineStrip
    glVertex3f 0#, 0#, 0#
    glVertex3f 0#, 0.7, 0#
    glEnd

    glBegin bmLineStrip
    glVertex3f 0#, 0#, 0#
    glVertex3f 0#, -0.7, 0#
    glEnd

'draw z axis

    glRasterPos3f 0, 0, 1
    axlab = " 3 "
    glPrint axlab               ' Print GL Text To The Screen
    glBegin bmLineStrip
    glVertex3f 0#, 0#, 0#
    glVertex3f 0#, 0#, 0.7
    glEnd

    glBegin bmLineStrip
    glVertex3f 0#, 0#, 0#
    glVertex3f 0#, 0#, -0.7
    glEnd

    cubes mat, lab, rw

    DrawGLScene = True          ' Draw till complete

End Function
```

Code 2.7. Drawing of the 3D object from the dataset. Code enabling the plotting of the 3D data object.

As shown in code 2.7, there are key methods within each statement which perform the drawing operations, using the data process in the 'R'-generated data matrix.

A further method was required for text handling enabling individual gene labels in the bitmap to be created for the 3D plot as shown in code 2.8.

```
Public Sub BuildFont(frm As Form)

    Dim hfont As Long                          ' Windows Font ID

    base = glGenLists(96)                      ' Storage For 96 Characters ( NEW )

    hfont = CreateFont(-12, 0, 0, 0, FW_BOLD, False, False, False, _
    ANSI_CHARSET, OUT_TT_PRECIS, CLIP_DEFAULT_PRECIS, ANTIALIASED_QUALITY,
    FF_DONTCARE Or DEFAULT_PITCH, "Courier New")

    SelectObject glView1.hdc, hfont            ' Selects The Font Created above

    wglUseFontBitmaps glView1.hdc, 32, 96, base ' Builds 96 Characters from 32

End Sub
```

Code 2.8: Function to enable text to be displayed together with the plotted object.

The 3D objects on the plotting surface are displayed as cuboid points, each with different colour surfaces to enhance the 3D effect of a single light. The 'sp3' variable indicates the output vector from 'R' of the clustering result for any given analysis method which has returned results as shown in code 2.9.

```
Public Sub cubes(mat() As Single, ByRef lab() As String, rw As Integer)

  Dim loop_m As Integer
    For loop_m = 1 To rw - 1
    If ch2.value = 1 Then GoTo L3 Else GoTo L2 L3:
    If sp3.cells(loop_m, 3) = "1" Then GoTo L2 Else GoTo L1 L2:
    glColor3f 0.5, 0.5, 1#              ' Set The Color To Blue initally
    xval = mat(loop_m, 1):  mat(loop_m, 1) = xval
    yval = mat(loop_m, 2):  mat(loop_m, 2) = yval
    zval = mat(loop_m, 3):  mat(loop_m, 3) = zval

    glColor3f 0#, 1#, 0#
    glBegin bmQuads                     ' Draw A Quad
        If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 1#, 0.5, 0.5
        End If

        End If

        glVertex3f xval, yval - 0.02, zval - 0.02' Bottom Left Of The Quad  Back)
        glVertex3f xval - 0.02, yval - 0.02, zval - 0.02  ' Bottom Right Of Quad)
        glVertex3f xval - 0.02, yval, zval - 0.02  ' Top Right Of The Quad  Back)
        glVertex3f xval, yval, zval - 0.02         ' Top Left Of The Quad  Back)

        glColor3f 0#, 0#, 1#                    ' Set The Color To Blue
         If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 0.5, 0.5, 0.5

        End If

' Set The Color To Green
      glVertex3f xval, yval, zval               ' Top Right Of The Quad  Top)
      glVertex3f xval - 0.02, yval, zval        ' Top Left Of The Quad  Top)
      glVertex3f xval - 0.02, yval, zval - 0.02 ' Bottom Left Of The Quad  Top)
      glVertex3f xval, yval, zval - 0.02        ' Bottom Right Of The Quad  Top)
```

…code 2.9 continued overleaf…

…code 2.9 continued from previous page

```
            glColor3f 1, 0.5, 0# ' Set The Color To Orange

 If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 0.5, 0.5, 0.5
       End If

  glVertex3f xval, yval - 0.02, zval           ' Top Right Of The Quad  Bottom
  glVertex3f xval - 0.02, yval - 0.02, zval    ' Top Left Of The Quad  Bottom
  glVertex3f xval - 0.02, yval - 0.02, zval - 0.02 'Bottom Left Of The Quad
  glVertex3f xval, yval - 0.02, zval - 0.02    ' Bottom Right Of The Quad
  glColor3f 1#, 0#, 0#                         ' Set The Color To Red
       If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 0.5, 0.5, 0.5
      End If

  glVertex3f xval, yval, zval                   ' Top Right Of The Quad  Front)
  glVertex3f xval - 0.02, yval, zval            ' Top Left Of The Quad  Front)
  glVertex3f xval - 0.02, yval - 0.02, zval ' Bottom Left Of The Quad
  glVertex3f xval, yval - 0.02, zval        ' Bottom Right Of The Quad

       glColor3f 1#, 1#, 0#                     ' Set The Color To Yellow
       If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 0.5, 0.5, 0.5
       End If

glVertex3f xval, yval - 0.02, zval - 0.02  ' Bottom Left Of The Quad  Back
glVertex3f xval - 0.02, yval - 0.02, zval - 0.02 'Bottom Right Of The Quad Back
glVertex3f xval - 0.02, yval, zval - 0.02      ' Top Right Of The Quad Back
glVertex3f xval, yval, zval - 0.02             ' Top Left Of The Quad  Back
```

…code 2.9 continued overleaf

```
…code 2.9 continued from previous page

        glColor3f 0#, 0#, 1#                    ' Set The Color To Blue
         If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 0.5, 0.5, 0.5
        End If

glVertex3f xval - 0.02, yval, zval    ' Top Right Of The Quad  Left
glVertex3f xval - 0.02, yval, zval - 0.02  ' Top Left Of The Quad  Left
glVertex3f xval - 0.02, yval - 0.02, zval - 0.02 ' Bottom Left Of The Quad  Left
glVertex3f xval - 0.02, yval - 0.02, zval     ' Bottom Right Of The Quad  Left

        glColor3f 1#, 0#, 1#                    ' Set The Color To Violet
         If ch1.value = 1 Then
            glColor3f 1#, 1#, 1#
            If sp3.cells(loop_m, 2) = "1" Then glColor3f 1#, 0#, 0#
            If sp3.cells(loop_m, 2) = "2" Then glColor3f 0#, 1#, 0#
            If sp3.cells(loop_m, 2) = "3" Then glColor3f 0#, 0#, 1#
            If sp3.cells(loop_m, 2) = "4" Then glColor3f 1#, 0#, 1#
            If sp3.cells(loop_m, 2) = "5" Then glColor3f 0#, 1#, 1#
            If sp3.cells(loop_m, 2) = "6" Then glColor3f 1#, 1#, 0#
            If sp3.cells(loop_m, 2) = "7" Then glColor3f 0.5, 0.25, 0
            If sp3.cells(loop_m, 2) = "8" Then glColor3f 0.5, 0.5, 0.5
        End If

glVertex3f xval, yval, zval - 0.02        ' Top Right Of The Quad  Right
glVertex3f xval, yval, zval               ' Top Left Of The Quad  Right
glVertex3f xval, yval - 0.02, zval        ' Bottom Left Of The Quad  Right
glVertex3f xval, yval - 0.02, zval - 0.02 ' Bottom Right Of The Quad  Right
    glEnd
    Red = labelcol And &HFF&
    Green = (labelcol And &HFF00&) \ &H100&
    Blue = (labelcol And &HFF0000) \ &H10000
    glColor3f Red, Green, Blue
    glRasterPos3f xval, yval, zval
    labz = lab(loop_m) 'mf2.Cells(loop_m + 1, 1)
    If ch3.value = 1 Then glPrint " " & labz & " "  ' Print GL Text To The Screen
    If ch3.value = 0 And ch4.value = 1 And sp3.cells(loop_m, 3) = "1" Then
glPrint " " & labz & " "
L1:
    Next loop_m
End Sub
```

Code 2.9: Selection of probes and colours available. Code to facilitate individual probes in clusters of interest to be highlighted in a particular colour.

The associated coding required to implement clustering of the source data using the available 'R' statistical programming language will be examined in depth over the following sections.

2.11.6 Principal components analysis

One of the first steps to permit cluster analysis using PCA is passing the 'Activesheet.spl' data from the Excel spreadsheet component into 'R' where it is created as a data matrix as outlined in section 2.11.3. Principal components analysis is performed on the data matrix as follows, initially returning analysis results yet also facilitated availability for 3D-viewing as shown in code 2.10

```
sconn.EvaluateNoReturn ("library(MASS)")

sconn.EvaluateNoReturn ("library(lattice)")

str7 = "pc.cr <- prcomp(Car)"

sconn.EvaluateNoReturn (str7)

sconn.EvaluateNoReturn "print(pc.cr)"

sconn.EvaluateNoReturn ("cols<-dim(pc.cr$ rotation)")
```

Code 2.10: Creating a principal components analysis plot. Code required to be sent to 'R' to generate the plot from I-10.

Within the PCA form, there is also the option to generate a scree plot allows the user to observe the individual fractions of total variance in the data as represented by each principal component. This is performed as outlined in code 2.11.

```
sconn.EvaluateNoReturn "windows()"
str7="y<-pc.cr$sdev^2": sconn.EvaluateNoReturn (str7)
str7="prop<-cumsum(pc.cr$sdev^2/sum(pc.cr$sdev^2))":sconn.EvaluateNoReturn (str7)
str7="m<-max(y)": sconn.EvaluateNoReturn (str7)
str7="fc<-m*0.05": sconn.EvaluateNoReturn (str7)
str7="x<-barplot(y, col=" & Chr(34) & "grey" & Chr(34) & ",
ylim=c(0,m+(m*0.2)))": sconn.EvaluateNoReturn (str7) '
str7="text(x,y+fc, round(prop, 2), cex = 0.6, adj = NULL)":
sconn.EvaluateNoReturn(str7)
```

Code 2.11: Generation of the scree plot for PCA analysis. Code required to be sent to 'R' from I-10.

The classical representation within 'R', which is generated in an 'R' window, is the biplot which shows the directions of the principal components in a two dimensional representation which can be difficult to visualise. The two dimensional representation is created as highlighted in code 2.12

```
sconn.EvaluateNoReturn "windows()"
str1 = "biplot(pc.cr, choices = 1:2 , cex = 0.5, scale = 1)"
sconn.EvaluateNoReturn (str1)
```

Code 2.12: Creation of a 2D PCA plot. Code required to be sent to 'R' to generate the plot from I-10.

OpenGL can be used to plot a 3D representation of PCA results using the previously outlined OpenGL functions. This adds value to the way in which the results are interpreted by 'R' using the R-(D)COM interface.

2.11.7 Self Organising Maps

Self organising maps forces the data into a preset number of groups based on which profiles within the data are similar, as previously outlined in Chapter 1. The option for Self Organising Maps is found under the Data Reduction menu in I-10. The data matrix is created as described previously for any given dataset loaded into I-10. However for Self Organising Maps, the x and y component relating to the number of groups the user wants the data to be forced into for the clustering needs to be specified. This is achieved by passing the value of the 'Combo' drop down box from the I-10 form options for Self Organising Maps into the string command script which is ultimately sent to 'R'. Note the initialising of the 'R' connection and the automatic loop from 1 to 20 to set up the combo1 and combo2 box which is performed upon loading of the form as outlined in code 2.13.

```
Private Sub Form_Load()
  Set sconn = New StatConnector
  sconn.Init "R"
  For i = 1 To 20
    Combo1.AddItem (i)
    Combo2.AddItem (i)
  Next i
  Combo1.ListIndex = 1
  Combo2.ListIndex = 1
```

Code 2.13: Creating the user defined cluster number for SOM. Code to create the drop down boxes in I-10.

When the form is loaded, the data is retrieved from the active Excel sheet and then sent to 'R' for subsequent SOM analysis as outlined in code 2.14.

```
sconn.EvaluateNoReturn "library(som)"

str2 = "sresult <- som(Car, xdim=" & Combo1.Text & ", ydim=" & Combo2.Text & ",
topol=" & Chr(34) & "hexa" & Chr(34) & " ,neigh=" & Chr(34) & "gaussian" &
Chr(34) & ")"

sconn.EvaluateNoReturn (str2)

str3 = "plot(sresult, ylim=c(" & mini & "," & maxi & " ))"

sconn.EvaluateNoReturn (str3)
```

Code 2.14: Performing SOM analysis with a dataset. Code sent to 'R' to perform the analysis from I-10.

A typical display of results from Self Organising Maps analysis in 'R' is show in figure 2.18.

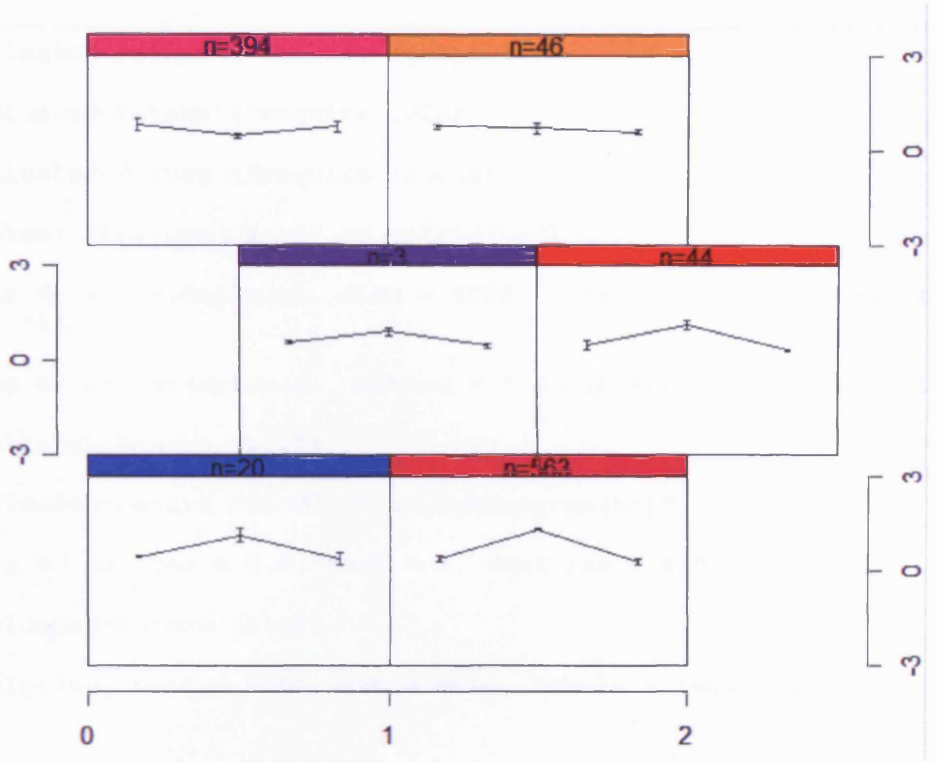


Figure 2.18: Self organising maps (SOM) – the dataset is divided into six profiles – the numbers of probes in each can be seen at the top of each box for each cluster of profiles.

2.11.8 Hierarchical clustering

After passing the contents of 'Activesheet.sp1' to 'R', hierarchical clustering can be performed. Note the requirement as outlined in code 2.15 for loading of the graphics, cluster and 'R' stats libraries to perform hierarchical clustering and results to be drawn as a heatmap.

```
sconn.EvaluateNoReturn ("library (graphics)")

sconn.EvaluateNoReturn ("require (cluster)")

sconn.EvaluateNoReturn ("require (stats)")

sconn.EvaluateNoReturn "x  <- as.matrix(Car)"

str1 = "hc <- agnes(daisy(x), diss = TRUE, , method = " & Chr(34) & "complete" &
Chr(34) & ")"

str1 = "hc <- hclust(daisy(x), method = " & Chr(34) & "average" & Chr(34) & ")"

sconn.EvaluateNoReturn (str1)

sconn.EvaluateNoReturn "dend1 <- as.dendrogram(hc)"

str1 = "op <- par(cex = 0.4, font = 1, font.lab = 4)"

sconn.EvaluateNoReturn (str1)

str1 = "plot(hc, main = NULL, sub = NULL, labels = labs, horiz = TRUE)"
```

Code 2.15: Generation of a hierarchical clustering heat map. Code sent to 'R' to generate the heat map.

2.11.9 Fuzzy analysis

After the data matrix has been created and sent to 'R' as outlined earlier, fuzzy clustering is performed on the dataset. To force the data into a given number of clusters, a combination box is again used to choose the cluster number and pass it into the string which is sent to 'R' via the R-(D)COM interface as the variable 'clus' as outlined in code 2.16.

```
clus = Combo1.Text

sconn.EvaluateNoReturn ("library(cluster)")

str7 = "fannyx <- fanny(Car," & clus & ")"

sconn.EvaluateNoReturn (str7)
```

Code 2.16: Performing fuzzy clustering in 'R'. Code sent to 'R' to generate clustered results from I-10.

Once 'str7' has been sent to 'R' to process using fuzzy analysis, the output can be stored. For example I-10 creates a file listing all the Affymetrix probes and the clusters they have been assigned as a *.csv file into the root installation directory of I-10. It is a two column file with Affymetrix probe ID in one column and the cluster number to which it has been assigned in the second column. This can then either be added into the database or ontological searching performed of the genes within each cluster.

```
sconn.EvaluateNoReturn ("fannyc <- fannyx$clustering")

sconn.EvaluateNoReturn "setwd('" & "C:/Development/" & "')"

sconn.EvaluateNoReturn "getwd()"

sconn.EvaluateNoReturn ("write.csv(fannyc, " & Chr(34) & "Fuzzyoutput.csv" &
Chr(34) & ")")

str7 = "fannyx"
sconn.EvaluateNoReturn (str7)

str7 = "summary (fannyx)"
sconn.EvaluateNoReturn (str7)

str7 = "plot (fannyx, labels = 2, lines = 0, stand = TRUE, which.plots = 1, cex =
1, color = TRUE, plotchar = TRUE, col.p = " & Chr(34) & "dark green" & Chr(34) &
")" 'col.clus = if(color) c(2, 4, 6, 3) else 5)"

sconn.EvaluateNoReturn (str7)
For i = 1 To clus
    For j = 1 To rw - 1
       tmp3 = "fannyx$ membership [" & j & "," & i & "]"
        xv = sconn.Evaluate(tmp3)
        mf2.cells(j + 1, i + 1) = xv
        tmp3 = "fannyx$ clustering [" & j & "]"
        xv = sconn.Evaluate(tmp3)
        mf2.cells(j + 1, 5) = xv
            On Error GoTo error_handler
        mf2.cells(1, i + 1) = i
        mf2.cells(j + 1, 1) = dat.cells(j + 1, 1)
    Next j
    cb1.AddItem (i)      'fill in combo boxes for 3D scatterplot
    cb2.AddItem (i)
    cb3.AddItem (i)
  Next i

error_handler:
cb1.SelText = "1"
cb2.SelText = "2"
cb3.SelText = "3"
```

Code 2.17: Plotting the Fuzzy clustering results and passing returned results to OpenGL for plotting.

A plot of the clusters and associated Affymetrix IDs from fuzzy clustering are drawn using the code shown in code 2.17, however the generated CSV file is also useful in that it fully defines cluster membership whereas some probes can overlap in the plot view in large datasets making interpretation difficult. The CSV file can be renamed and imported back into the Affymetrix database for future comparison with other analysis methods. Furthermore the cluster members can subsequently be plotted in 3D using the output from 'R. Cluster membership is also displayed alongside the sample results column for each Affy ID as well as written to a file for the user to view or email to colleagues.

2.11.10 Partitioning around medoids (PAM)

Similar in syntax to Fuzzy clustering, an active worksheet 'Sp1' is generated together with Affymetrix probe information according to cluster membership as shown in code 2.18.

```
clus = Combo1.Text

str7 = "cl<-pam(Car," & clus & ")"

sconn.EvaluateNoReturn (str7)
sconn.EvaluateNoReturn ("pamlist <- cl$clustering")
sconn.EvaluateNoReturn "setwd('" & "C:/Development/" & "')"
sconn.EvaluateNoReturn "getwd()"
sconn.EvaluateNoReturn ("write.csv(pamlist, " & Chr(34) & "PAMoutput.csv" &
Chr(34) & ")")

sconn.EvaluateNoReturn ("plot(cl, labels=3)")
```

Code 2.18: Clustering using the PAM algorithm. Code passed to 'R' from I-10.

The plot, also called using the syntax in the box above, is returned within an 'R' window to I-10 for viewing or saving however a list of the PAM Cluster membership is also included which could, for example, be used to compare to the Fuzzy-derived cluster list. The returned results are also available for visualisation in 3D using OpenGL in 'R'.

3.11.11 K-Means

As an alternative to PAM, K-Mean can also be implemented through I-10, again choosing cluster number where appropriate as outlined in code 2.19.

```
clus = Combo1.Text

str7 = "cl<-kmeans(Car," & clus & ")"

sconn.EvaluateNoReturn (str7)
sconn.EvaluateNoReturn ("kmeanslist <- cl$clustering")
sconn.EvaluateNoReturn "setwd('" & "C:/Development/" & "')"
sconn.EvaluateNoReturn "getwd()"
sconn.EvaluateNoReturn ("write.csv(kmeanslist, " & Chr(34) & "Kmeansoutput.csv" &
Chr(34) & ")")

sconn.EvaluateNoReturn ("plot(x, col = cl$cluster)")
```

Code 2.19: Clustering using K-Means in I-10. Code passed to 'R' to perform K-Means from I-10.

2.11.12 Multidimensional Scaling (MDS)

MDS is located within the Class Prediction menu options. The MDS plot is generated within a panel in the form within I-10, instead of a separate window as shown in code 2.20. This was required so that the size of the Affymetrix probe ID text displayed against an MDS plot could be altered for clarity to as the probes can overlap making interpretation difficult.

```
sconn.EvaluateNoReturn ("library(MASS)")
sconn.EvaluateNoReturn ("library(stats)")

str7 = "ds<-dist(Car)"
'sconn.EvaluateNoReturn ("ds<-dist(Car)")
sconn.EvaluateNoReturn (str7)
sconn.EvaluateNoReturn "ct<-cmdscale(ds, k = 3)"
sconn.EvaluateNoReturn ("loc <- cmdscale(ds, k = 3)")
sconn.EvaluateNoReturn ("x <- loc[,1]")
sconn.EvaluateNoReturn ("y <- -loc[,2]")

str5 = "plot(x, y, type=" & Chr(34) & "n" & Chr(34) & ", cex = " & cexval & ")"

sconn.EvaluateNoReturn (str5)

str5 = "text(x, y, labs, cex = " & cexval & ")"

sconn.EvaluateNoReturn (str5)
```

Code 2.20: Performing multidimensional scaling in I-10. Code passed to 'R' from I-10.

The size of the Affymetrix probe ID's are plotted in a font which can be decreased and increased in size using two buttons above the plot output. The larger and smaller text for the probes only differ by either subtracting (to make the font smaller) or adding (to make the font larger) a value of 0.1 before the plot is re-drawn to reflect the changes. Code 2.21 highlights method which is used to achieve this functionality.

```
cexval = cexval + 0.1

str5 = "plot(x, y, type=" & Chr(34) & "n" & Chr(34) & ", cex = " & cexval & ")"

str5 = "text(x, y, labs, cex = " & cexval & ")"

sconn.EvaluateNoReturn (str5)
```

Code 2.21: Enabling alteration of text size in a Multidimensional analysis plot.

2.11.13 Correspondence Analysis

In comparison to previous methods, implementation of Correspondence analysis is very similar however involves additional libraries as highlighted in code 2.22. It should be apparent at this stage that 'R' uses a very similar syntax to perform each of the clustering methodologies. The individual functions and algorithms are hidden in their respective libraries. The user therefore does not need to manipulate any mathematics to return a result.

```
sconn.EvaluateNoReturn ("library(MASS)")
sconn.EvaluateNoReturn ("library(stats)")
sconn.EvaluateNoReturn ("library(grid)")
sconn.EvaluateNoReturn ("library(lattice)")

str7 = "ct<-corresp(Car[,2:" & stmp & "], nf = 3)"
sconn.EvaluateNoReturn (str7)
str7 = "cl<-trellis.par.get()"
sconn.EvaluateNoReturn (str7)
str7 = "cl$col<-" & Chr(34) & "white" & Chr(34)
sconn.EvaluateNoReturn (str7)

sconn.EvaluateNoReturn ("plot(ct, cl$col)")
```

Code 2.22: Code passed to 'R' in I-10 to perform correspondence analysis.

2.11.14 Clustering technique comparison

Before a statistical basis for comparing cluster techniques was developed, a way of comparing clustering output using colour schemes over layed alongside results was implemented in I-10 under the menu 'Advanced'.

However, recent incorporation of pvClust into I-10 facilitates bootstrap analysis to reveal which clusters (according to a specific technique e.g. HCA), and within this ultimately which genes, are most significant by setting a confidence level, for example p value = 0.05 or p=0.001. Typically a p value of 0.05 is chosen. Significant clusters by bootstrap analysis will be in revealed in red. This was implemented in I-10 as highlighted in code 2.23.

```
sconn.EvaluateNoReturn (Nwdata <- t(scale(t(Car))))

sconn.EvaluateNoReturn (hr <- hclust(as.dist(1-cor(t(Nwdata), method="pearson")),
method="complete"))

sconn.EvaluateNoReturn  (hc <- hclust(as.dist(1-cor(Nwdata, method="spearman")),
method="complete"))

sconn.EvaluateNoReturn (heatmap(Car, Rowv=as.dendrogram(hr),
Colv=as.dendrogram(hc), col=my.colorFct(), scale="row"))

sconn.EvaluateNoReturn (htree <- cutree(hr, h=max(hr$height)/1.5)

sconn.EvaluateNoReturn (mytreehc <- sample(rainbow(256)))

sconn.EvaluateNoReturn  (mytreehc <- mycolhc[as.vector(htree)]
heatmap(Car, Rowv=as.dendrogram(hr), Colv=as.dendrogram(hc), col=my.colorFct(),
scale="row", RowSideColors=mycolhc))
```

Code 2.23: Revealing hierarchical clustering probes at a predetermined level using 'R' in I-10.

The library pvClust can perform validation using bootstrap analysis using any of the previously outlined clustering techniques as detailed in code 2.24.

```
sconn.EvaluateNoReturn  (library(pvclust))

sconn.EvaluateNoReturn  ((pv <- pvclust(scale(t(Car)), method.dist="correlation",
method.hclust="complete", nboot=100))

sconn.EvaluateNoReturn  (plot(pv, hang=-1); pvrect(pv, alpha=0.95))

sconn.EvaluateNoReturn  (clsig <- unlist(pvpick(pv, alpha=0.95, pv="au",
type="geq", max.only=TRUE)$clusters))

sconn.EvaluateNoReturn  (dend_colored <- dendrapply(as.dendrogram(pv$hclust),
dendroCol, keys=clsig, xPar="edgePar", bgr="black", fgr="red", pch=20))

sconn.EvaluateNoReturn  (heatmap(Car, Rowv=dend_colored, Colv=as.dendrogram(hc),
col=my.colorFct(), scale="row", RowSideColors=mycolhc))
```

Code 2.24: Assessing significance of clusters revealed through hierarchical clustering in I-10.

A typical dendrogram plotted for HCA results for an individual cluster is shown here. The significant genes by bootstrap analysis are revealed and indicated by red boxes as highlighted in figure 2.19.

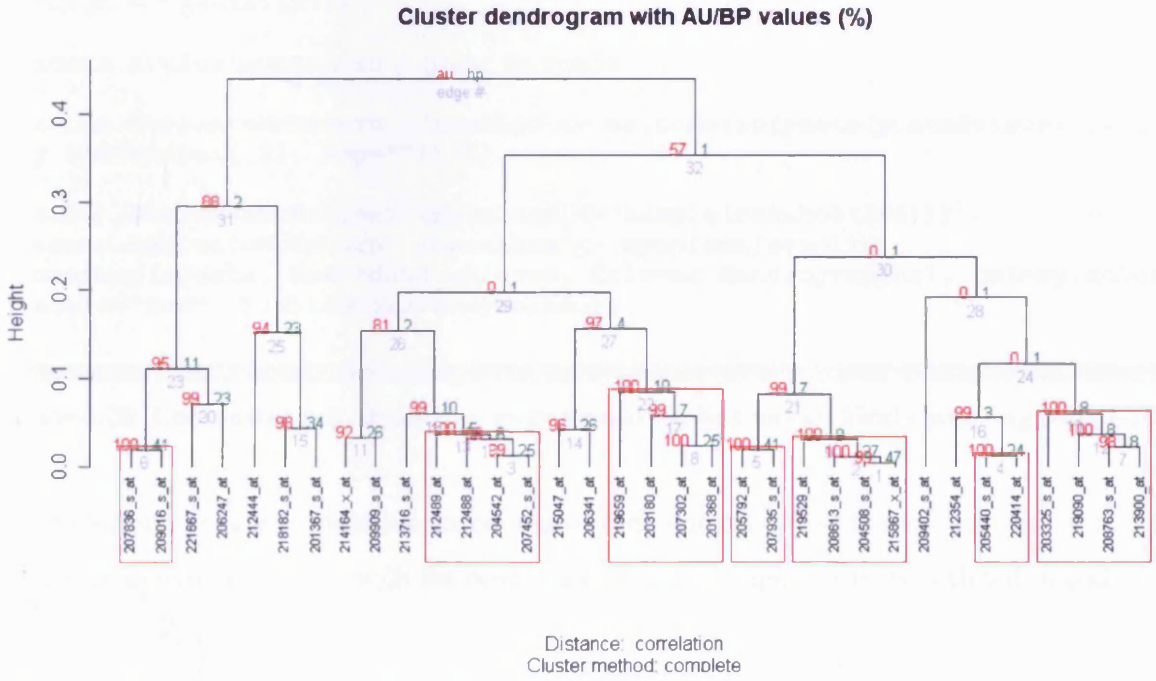


Figure 2.19 Dendogram of an individual cluster revealed by hierarchical clustering at a significance level of p value=0.05 using the pvClust library in 'R' from I-10.

Each clustering method can be compared against each other. For example, PAM can be compared against hierarchical clustering. This is highlighted in code 2.25.

```
sconn.EvaluateNoReturn  (Nwdist <- as.dist(1-cor(t(Car), method="pearson")))

sconn.EvaluateNoReturn  (pamy <- pam(Nwdist, max(mycl)))

sconn.EvaluateNoReturn  (mycolkm <- sample(rainbow(256)))
sconn.EvaluateNoReturn  (mycolkm <- mycolkm[as.vector(pamy$clustering)]
heatmap(mydata, Rowv=dend_colored, Colv=as.dendrogram(hc), col=my.colorFct(),
scale="row", RowSideColors=mycolkm))
```

Code 2.25: Comparing PAM results against hierarchical clustering using 'R' from I-10.

Self organising maps can also be compared against hierarchical clustering as highlighted in code 2.26.

```
sconn.EvaluateNoReturn  (library(som))

sconn.EvaluateNoReturn  (y <- t(scale(t(mydata))))

sconn.EvaluateNoReturn  (y.som <- som(y, xdim = 2, ydim = 3, topol = "hexa",
neigh = "gaussian"))

sconn.EvaluateNoReturn  (plot(y.som))

sconn.EvaluateNoReturn  (somclid <- as.numeric(paste(y.som$visual[,1],
y.som$visual[,2], sep=""))+1)

sconn.EvaluateNoReturn  (mycolsom <- sample(rainbow(256)))
sconn.EvaluateNoReturn  (mycolsom <- mycolsom[somclid]
heatmap(mydata, Rowv=dend_colored, Colv=as.dendrogram(hc), col=my.colorFct(),
scale="row", RowSideColors=mycolsom))
```

Code 2.26: Comparing self organising maps results against hierarchical clustering from I-10.

Furthermore, it is also possible to compare principal components analysis with self organising maps as shown previously with the other clustering techniques. This is outlined in code 2.27.

```
sconn.EvaluateNoReturn  (pca <- prcomp(Car, scale=T))

sconn.EvaluateNoReturn  (summary(pca))

sconn.EvaluateNoReturn  (library(scatterplot3d))

sconn.EvaluateNoReturn  (scatterplot3d(pca$x[,1:3], pch=20, color=mycolsom))
```

Code 2.27: Comparing principal components analysis with self organising maps from I-10.

Although the 'scatterplot3d' library is available in 'R' in conjunction with pvClust, it is not as powerful as that which can be produced by openGL in I-10. However it is an interesting module to allow plotting and comparison of the clustering techniques in a relatively small plot. The 'canvas' can be set to compare multidimensional scaling, hierarchical cluster, self organising maps and PAM as outlined in code 2.28.

```
sconn.EvaluateNoReturn  (loc <- cmdscale(Car, k = 3))

sconn.EvaluateNoReturn  (x11(height=8, width=8, pointsize=12); par(mfrow=c(2,2)))

sconn.EvaluateNoReturn  (plot(loc[,1:2], pch=20, col=mycolsom, main="MDS vs SOM
2D"))

sconn.EvaluateNoReturn  (scatterplot3d(loc, pch=20, color=mycolsom, main="MDS vs
SOM 3D"))

sconn.EvaluateNoReturn  (scatterplot3d(loc, pch=20, color=mytreehc, main="MDS vs
HC 3D"))
```

Code 2.28: Plotting results comparison of all four clustering techniques using I-10.

The associated comparative plot is produced in an 'R' graphics window which can be saved to PDF or as an image file for use in presentations or publications. An example of how typical results are presented can be seen in figure 2.20.

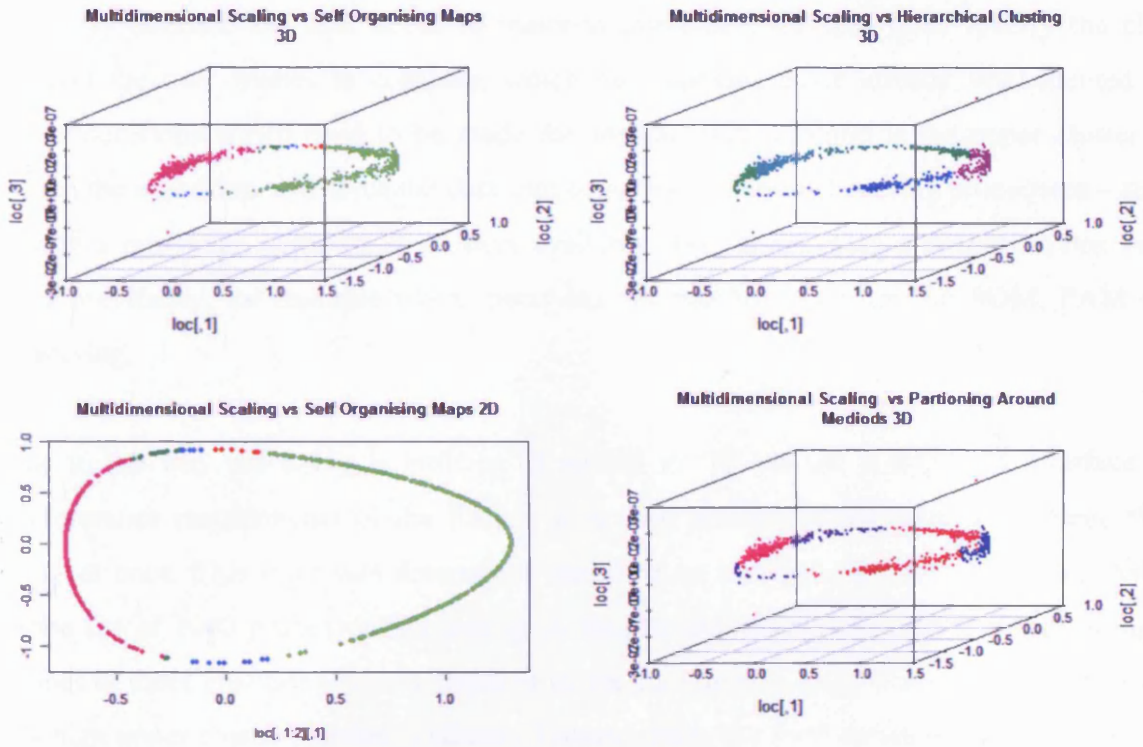


Figure 2.20 – Comparing results of each clustering method by overlaying the colour scheme of clusters determined using each method using the scatterplot3d library in 'R' from I-10.

2.11.15 ClValid

ClValid was one of the final modules added to I-10. It was released to the 'R' project community in January 2008 to assess and clarify the best clustering strategy for any given dataset. Although it had been sometime since a final version of I-10 was released and testing within Tenovus had already begun, ClValid was deemed to be powerful enough to warrant its addition to I-10 during the final stages of this project. Furthermore, its inclusion reinforces the key benefit of I-10 in that I-10 is able to have new 'R' library modules inserted as required, using a new Form created in Visual Basic to control the different parameters supplied for any given library. This was achieved in the same way as the other functions in I-10 were created as previously outlined.

ClValid contains functions for validating the results of cluster analysis. This is achieved by three different measures – 'Internal', 'Stability' and 'Biological'. An in depth example is shown in Chapter 3 where ClValid is used in the context of exploring endocrine resistance, where this section focuses upon the coding actually needed to allow the user to access the ClValid methods.

An early decision the user needs to make to implement ClValid is to specify the clustering method the user wishes to compare, which encompasses those already implemented in I-10. Other decisions which need to be made for analysis with ClValid is the upper cluster number which the algorithm will force the data into using the different clustering procedures – starting at 1 with a maximum value of 20 clusters available. This is set using a dropdown box in I-10 as used previously, for example when specifying the number of groups for SOM, PAM or fuzzy clustering.

Due to the way the string is built to be passed to 'R' via the R-(D)COM interface and the performance requirements of the library, it is only possible to compare up to three clustering types at once. This limit was determined based on an initial testing of a significant Affymetrix probe list of 1000 probes during testing. A dataset comprising of larger numbers of significant probes or more clustering types compared in one run can take many hours to complete, especially if a high upper cluster number is chosen. Consequently, the form developed in this thesis has two buttons, depending upon whether 2 or 3 clustering techniques are to be compared, for example, hierarchical clustering and PAM or hierarchical clustering, PAM and self organising maps. There are drop down box selections in I-10 for each as shown in figure 2.21, within the cluster comparison tool in class prediction. Results are output to the installation directory for I-10 in a text file in a similar way as returned from the clustering procedures previously outlined. This approach was chosen as opposed to displaying the results within I-10, as the extended processing time required I-10 to hang while waiting for over an hour for a result to be returned. Creating the files in a directory had no such issues and allows the user to use other windows applications while I-10 runs in the background. However it should be noted that ClValid is a computing intensive library and performs best on high specification computers.

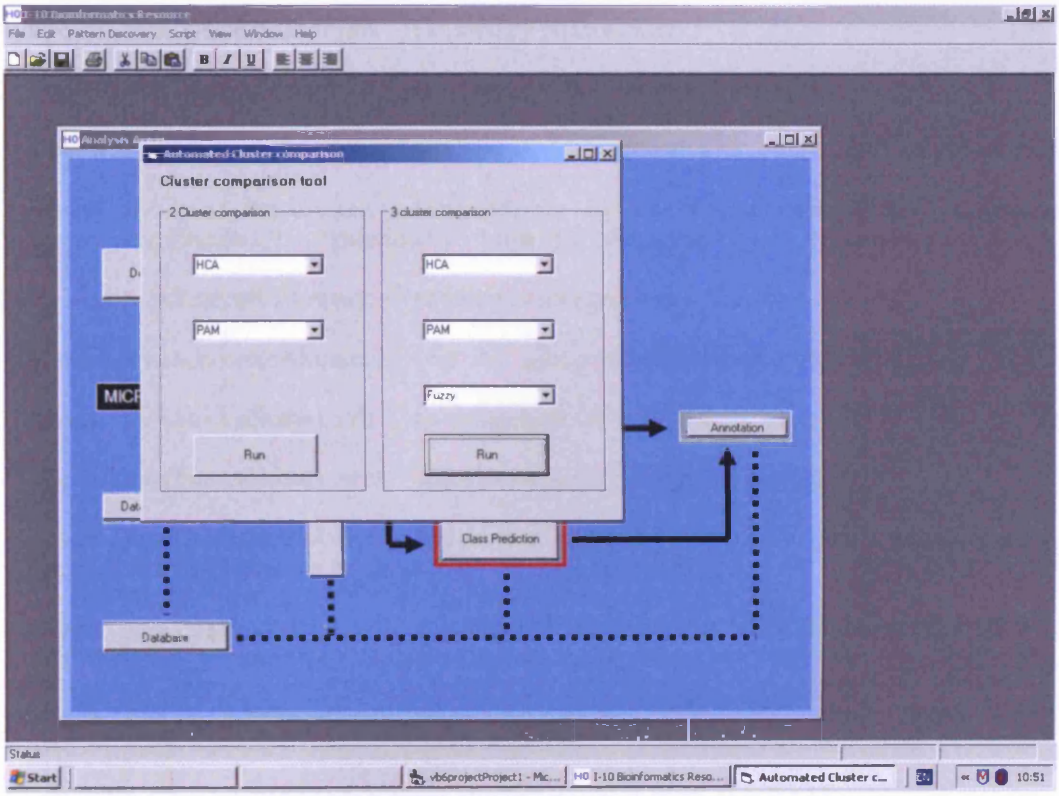


Figure 2.21 – Available user selections to select automatic clustering and comparison of clustering results within I-10 using CIValid.

The functions highlighted in code 2.29 shows examples of comparing hierarchical clustering, K-Means and PAM as applied to a 3-way group microarray experiment. For clarification, as shown previously, code 2.29 outlines method statements which populate the drop down boxes. All three types of validation available in CIValid ('Internal', 'Stability' and 'Biological') are performed in one step, however it would also be possible to run them separately.

(a) For Internal validation, the Library is loaded, and the Excel data frame 'Car' is ensured to be a matrix and given experimental arm headings. The headings can be typed into text boxes in visual basic and sent to 'R'. The number of columns to process is then indicated and the clustering methods specified (again from drop down boxes). Validation method is set to be internal. The summary of results is written to a file. Then the results are plotted through the 'R' window. This can be saved as a PDF or an image.

```
sconn.EvaluateNoReturn  (library(clvalid)

sconn.EvaluateNoReturn  (myresult <- as.matrix(Car[,-1]))

sconn.EvaluateNoReturn  (express <- myresult[, c("CON","TAMR","FASR")])

sconn.EvaluateNoReturn  (intern <- clValid(express, 2:12, clMethods =
c("hierarchical", "kmeans", "pam"), validation = "internal")

sconn.EvaluateNoReturn  (write.csv(intern,"intern.csv")

sconn.EvaluateNoReturn  (op <- par(no.readonly = TRUE))

sconn.EvaluateNoReturn  (par(mfrow = c(2, 2), mar = c(4, 4, 3, 1)))

sconn.EvaluateNoReturn  (plot(intern, legend = FALSE))

sconn.EvaluateNoReturn  (plot(nClusters(intern), measures(intern, "Dunn")[, , 1],
type = "n", axes = F, xlab = "", ylab = ""))

sconn.EvaluateNoReturn  (legend("center", clusterMethods(intern), col = 1:9, lty
= 1:9, pch = paste(1:9)))

sconn.EvaluateNoReturn  (par(op))
```

Code 2.29: Comparing hierarchical clustering, K-means and PAM using the 'R' library Clvalid.

(b) The stability validation is also performed with parameters sent to the algorithm as outlined in code 2.30. The data frame has already been created at this point as shown previously. The same clustering techniques are compared for stability as shown in code 2.30.

```
sconn.EvaluateNoReturn  (stab <- clValid(express, 2:6, clMethods =
c("hierarchical", "kmeans", "pam"), validation = "stability"))

sconn.EvaluateNoReturn  (par(mfrow = c(2, 2), mar = c(4, 4, 3, 1)))

sconn.EvaluateNoReturn  (plot(stab, measure = c("APN", "AD", "ADM"), legend =
FALSE))

sconn.EvaluateNoReturn  (plot(nClusters(stab), measures(stab, "APN")[, , 1], type
= "n", axes = F, xlab = "", ylab = ""))

sconn.EvaluateNoReturn  (legend("center", clusterMethods(stab), col = 1:9, lty =
1:9, pch = paste(1:9)))

sconn.EvaluateNoReturn  (par(op))
```

Code 2.30: Assessing stability of the clustering algorithms hierarchical clustering, K-means and PAM.

(c) The biological validation step is a very powerful feature of the library. It uses Biobase, Annotate and the Affy133A modules to annotate and explore the clustering results based on ontology. Using all Go ontological functional categories, they are applied to the clusters generated by hierarchical, K-Means or PAM algorithm as highlighted in previous steps. A graphical output is generated as outlined in figure 2.22 with the resulting 'biological validation' plot highlighted in code 2.31.

```
sconn.EvaluateNoReturn  (if (require("Biobase") && require("annotate") &&
require("GO") && require("hgu133a")) { bio <- clValid(express, 2:6, clMethods =
c("hierarchical", "kmeans", "pam"), validation = "biological", annotation =
"hgu133a", GOcategory = "all"))

sconn.EvaluateNoReturn  (if (exists("bio")) optimalScores(bio))

sconn.EvaluateNoReturn  (if (exists("bio")) plot(bio, measure = "BHI", legendLoc
= "topleft"))

sconn.EvaluateNoReturn  (if (exists("bio")) plot(bio, measure = "BSI"))
```

Code 2.31: Producing biological validation of clustering results to assess ontologically the best performance of each clustering technique.

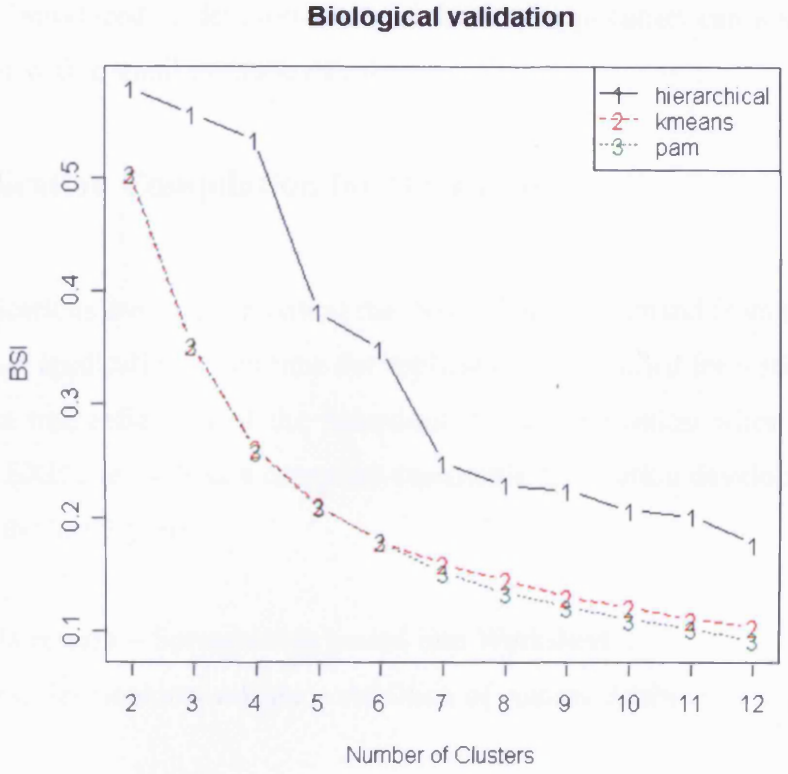


Figure 2.22 – Example biological validation output produced in 'R' displayed via I-10. The effect of increasing cluster number against BSI score according to individual clustering technique can be observed.

Individual cluster membership information produced by clvalid can be extracted and written to a file as outlined in code 2.22.

```
sconn.EvaluateNoReturn  (hc <- clusters(bio, "hierarchical")

sconn.EvaluateNoReturn  (write.csv(hc,"Biohcclustering.csv")
```

Code 2.22: Writing a file of hierarchical clustering results from the library clvalid.

The CSV spreadsheet files can also be easily added into the database as created and named appropriately for future reference from the analysis.

A full listing of all visual basic code which comprises I-10 can be found on the accompanying CD-ROM attached to the rear cover of this thesis. Although I-10 is continually being refined as

new libraries are introduced, a demonstration of the I-10 application can also be found on the CD-ROM together with a small example dataset.

2.12 I-10 Application Compilation for Distribution

Visual basic applications are compiled using the 'Make EXE' command from the file menu in the Visual Basic design application. Each time the application is compiled for testing using the 'play' function, this is a true reflection of the behaviour of the application when compiled and run directly from the EXE file. I-10 as a compiled executable application developed in Visual Basic has evolved over the last 3 years.

Version 1.0 – Beta release – Spreadsheets pasted into Worksheet
Version 1.1 – First development release – Addition of Access database to supply data, addition of pvClust
Version 1.2 – Integration of new cluster comparison library – clValid

When compiled, the EXE file is approximately 3MB in size. However its function is dependent on certain libraries, the Affymetrix database and the R-(D)COM interface component being installed as well. All information needed is provided with the installation distribution as a text file accompanying the CD-ROM. Visual basic gives the EXE icon for I-10, designed early in development,t which is also used stylistically throughout the application as shown in figure 2.23.

Figure 2.23: Icon logo chosen to represent the program launch icon for Informatics Tenovus (I-10).

A 'splash screen' was also developed as commonly found on Windows applications when first launched. This was added to I-10 to give the application a more professional appearnce (including the I-10 version number), and complemented the icon development as shown in figure 2.24.

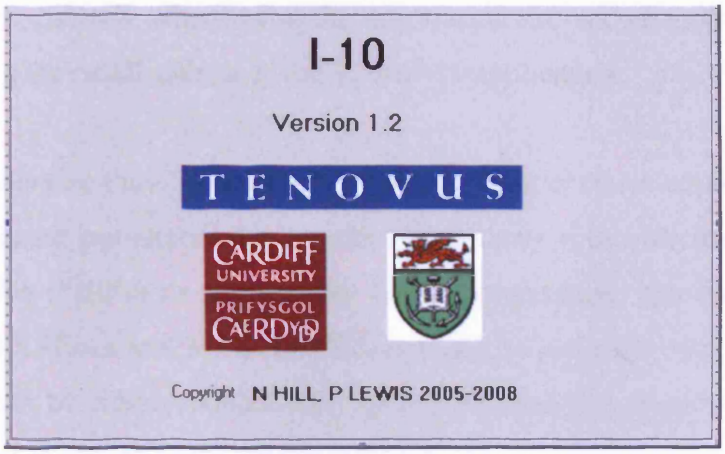


Figure 2.24 – I-10 'splash screen' which is shown to users upon entering I-10 when the application is loaded.

2.13 – I-10 Installation

Installation of I-10 is a straight forward process. Once 'R' and the D-COM interface have been installed, the I-10 executable file together with the database file is copied to a directory of the user's choice is the only additional installation files required. Initial installations were performed in this proeject on a dedicated, relatively high specification Windows XP computer devoted to data analysis within Tenovus. During initial testing, a Windows XP computer with at least a Pentium III processor and 512MB of RAM is required for the application to run at a reasonable speed. However as computer technology gets quicker, the application will run faster. During the process of the evaluation, members of the Tenovus Gene discovery team were shown the installation process step by step. Several demonstration sessions of how to use the system were performed. Users who attended the sessions preferred to make their own notes on how to use the I-10 as they were guided through the system rather than a user manual.

I-10 can be launched by double clicking I-10.exe in the installation directory, or from the start menu or desktop icon if a link has been created. The only installation anomalies encountered on some systems were those which could result in four system extension files being missing from certain versions of Windows XP depending upon the level of updates on the individual machines.

Any anomalies users discover when using the application are encouraged to be emailed for a swift resolution using the email address given in the I-10 application.

Since the release of Service Pack 2 for Windows XP, the way in which applications developed in Visual Basic 6.0 execute has altered due to potential security risks. Microsoft re-issued certain system extension files (*.dll's) to address this issue. Consequently the files **MSCBCM60.dll**, **IMM32.dll**, **RICHTX32.ocx** and **MSCOMCTL.ocx** will be included with all distributions of I-10. The four files can be simply copied into the system directory found within the Microsoft Windows directory. It is safe to agree for the included copies to be written over the existing files, at the time of writing within the system directory.

2.14 Discussion

A key aim of this project was to develop a platform to improve and increase the speed and ease of use in revealing potential new genetic targets from Affymetrix array portfolios. Although Genesifter was already being used for basic analysis of this dataset, advanced data reduction and clustering techniques were not being explored. Consequently no system being routinely used was optimised to reveal new differentially-expressed genes from the Affymetrix arrays that could provide future interesting biomarkers/targets in antihormone resistance, for example, which would otherwise have been overlooked. Development of I-10 had to offer more flexibility in terms of what can be analysed beyond that available in Genesifter, incorporating advanced data reduction techniques, advanced multiple class discovery and class prediction techniques, and an ability to more efficiently obtain comprehensive gene annotation to generate robust data. The system had to be user-friendly (graphically-driven) yet also MIAME compliant for publication purposes.

Before development of I-10 began, existing technologies and approaches in these regards were evaluated in detail before system architecture choices were finalised. At all stages, the requirements of the end user were considered in detail. In part to fulfil this aim, a new versatile analysis platform has been developed which also has potential to analyse data beyond in vitro microarray data. Indeed, potentially any type of multivariate biomedical data could theoretically

be analysed using this I-10 however the primary focus was to analyse Affymetrix microarray data. Critically, the developed platform fulfilled the important aim of being easy to use. No analysis tool would have been powerful if the end user was not able to generate results from the system. However, working with key members of the Tenovus research team in workshop sessions using I-10, the user experience was refined and the application's design proven user friendly.

I-10 has also been developed in such a way as to be continually upgradeable and expandable. Continued addition of up to date clustering libraries such as 'clValid', have added considerable value to the software. Alternative applications not using the 'R' frame work would have resulted in a need for complete redevelopment of the application to accommodate such upgrading. However due to the way in which the new platform has been designed it can be continually expanded by adding Visual Basic forms containing the necessary 'R' scripting. The versatility of the 'R' project statistical programming environment together with Microsoft Visual Basic user interface design has proven a powerful combination.

One of the hardest decisions to make during development was choosing a name for the application. After much consideration, it was decided to use a combination of the name of the discipline of study (Informatics) and the name of the sponsoring charity of the research group – Tenovus. Consequently the name 'Informatics Tenovus' was born – or 'I-10' for short.

I-10 has constantly evolved over the 3 years of the project. The ability of 'R' to be expanded with libraries for diverse statistical and clustering analyses as applied to microarray data, the 3D-graphics and annotation capabilities incorporated into I-10, and the framework which I-10 provides to accept new features has satisfied the primary aims of the project. Importantly, I-10 has also been developed to be non-commercial in its availability allowing the greater cancer research community as a whole to access I-10.

Chapter 3

Application of I-10 to *in vitro* endrocrine response and resistance microarray data

Chapter 3 – Application of I-10 to in vitro endrocrine response and resistance microarray data

3.1 Background

One of the motivations of I-10 development was to allow greater understanding of genetic events occurring within the Tenovus MCF7 *in vitro* models. This was achieved by generating Affymetrix microarrays of the various models which Tenovus had performed. Appendix 1 outlines the methodology which was adopted to prepare the models for Affymetrix Microarray analysis. To test the ability of I-10, a three way comparison of *in vitro* anti-hormone resistance models where the MCF7 model replicates antioestrogen response versus the acquired Tamoxifen and Faslodex resistant cell lines was proposed, having been treated with 10-7M Tamoxifen and 10-7M Faslodex (*Fulvestrant*) This comparison had proven difficult to achieve in Genesifter which had previously been used by Tenovus. However the development of I-10 facilitated such a comparison to be made.

Hierarchical clustering, K-Means, Partitioning around medoids (PAM) and Fuzzy Analysis algorithms available through I-10 were applied to the Affymetrix data generated from the models and subsequently cross compared in their effectiveness to reveal dominant genetic profiles. This was achieved by comparing the internal, stability and biological cluster validation measures using the library clValid available within I-10. clValid, as previously outlined in Chapter 2, allows the user to simultaneously evaluate multiple clustering algorithms while varying the number of clusters. These measures help numerically and graphically determine the most appropriate clustering method and optimal number of clusters for the dataset of interest. Biological validation was assessed in clValid by comparing Go ontology classification of the probe set for each clustering method.

Consequently, it was hoped that the analysis would demonstrate confidence that I-10 could be used to reveal new potential genetic targets which could be studied in depth in a future project.

3.2 Phenotype of MCF7 Models

Before high throughput target identification was performed, a phenotypic study based on published classifiers was performed through I-10 using the Affymetrix database compiled across the responsive and resistant cell lines. I-10 has the ability to add known classification lists to the database and apply them during analysis. This can then be used to assess if the MCF7 models have lost or regained a particular phenotype based on present or absent call.

Sorlie et al in 2001 published a key breast cancer study which determined that the previously characterised luminal epithelial/oestrogen receptor-positive group could be further divided into at least two subgroups, each with a distinctive expression profile (Sørlie et al, 2001) [74]. Clustering studies showed the subsequent classification into *errb2+*, basal, normal and the luminal categories, as determined by the study, which correlated well with patient outcome. Survival analysis on sub cohorts of patients showed different outcomes when assigned to the different classifiers. For example, a basal phenotype was associated with a poor prognosis. There was a significant difference in outcome for the two oestrogen receptor-positive groups – luminal being superior to the others (Sørlie et al, 2001) [74].

The same classifiers can be applied to the *ER+* luminal MCF7 model versus TAMR and FASR cell lines to assess any phenotypic shifts which have developed in resistance when treated with both Tamoxifen and Faslodex. Appendix 2 outlines a summary of the presence or absence of the genes and corresponding Affymetrix probes applied to the various MCF7 derived models. The MCF7 cell line is a luminal cell line which is a phenotype not lost in resistance as shown in the luminal table in appendix 2.

However it is also interesting to note that the models are not shifting to a basal phenotype in resistance as key probes associated with a basal phenotype are absent in control, TAMR and FASR models.

3.3 Confirmation of Quality of Affymetrix Samples Through I-10

The way in which RNA samples are initially prepared for array hybridisation can influence replicate consistency and robustness of identifying the differentially expressed genes. The Microarray Variance Analysis (MVA) is modification of a simple quality plot of repeat samples in the array results. Before detailed array analysis began, it was revealed by comparing each of the three replicates against each other that some discrepancy existed in third control MCF7 replicate. It was also confirmed when the array was reproduced by a lower IQR value of the new re-arrayed sample. Ideally as small as possible a difference is expected between the arrays for them to accurate replicates.

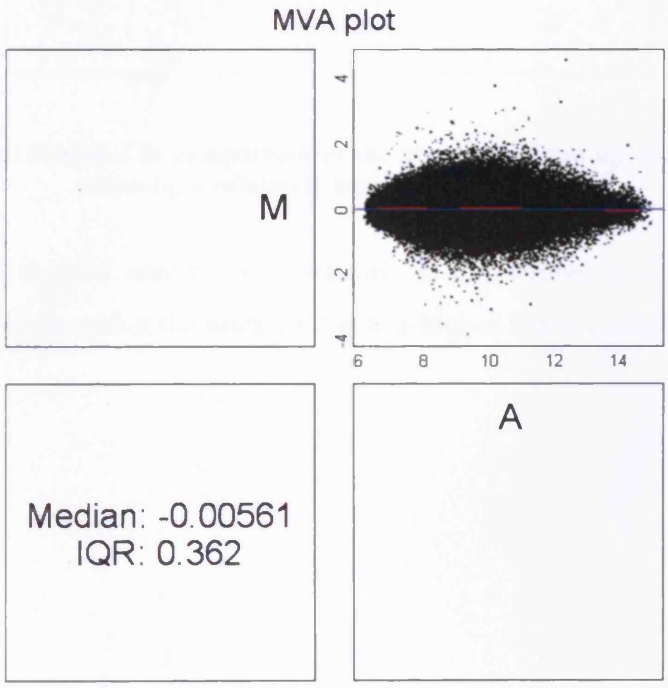


Figure 3.1: Good control sample 1 compared to the old control 3 – relatively high IGR showing a poor association between control 1 and control 3.

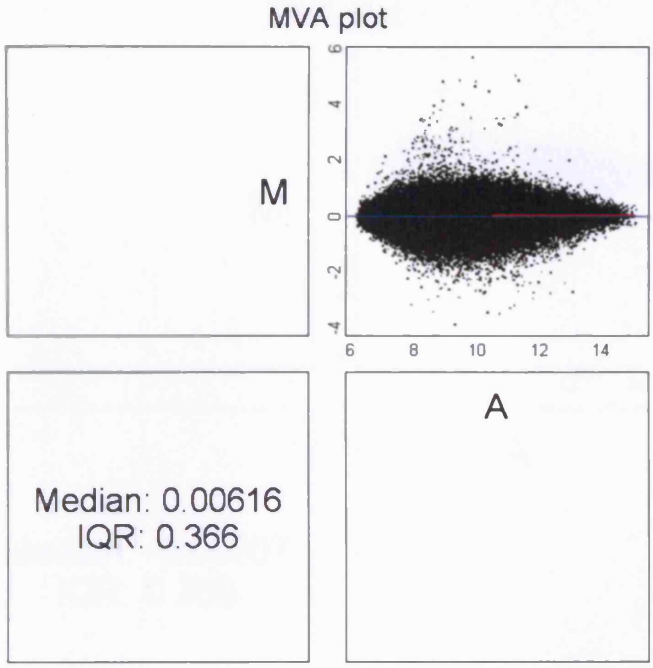


Figure 3.2: Good control sample 2 in comparison to the poorly performing control 3 sample, again showing a relatively high IQR sample.

A newly prepared and arrayed control 'C3' sample is also shown in Figure 3.5 where the differences observed are less within the samples due to a higher linear correlation.

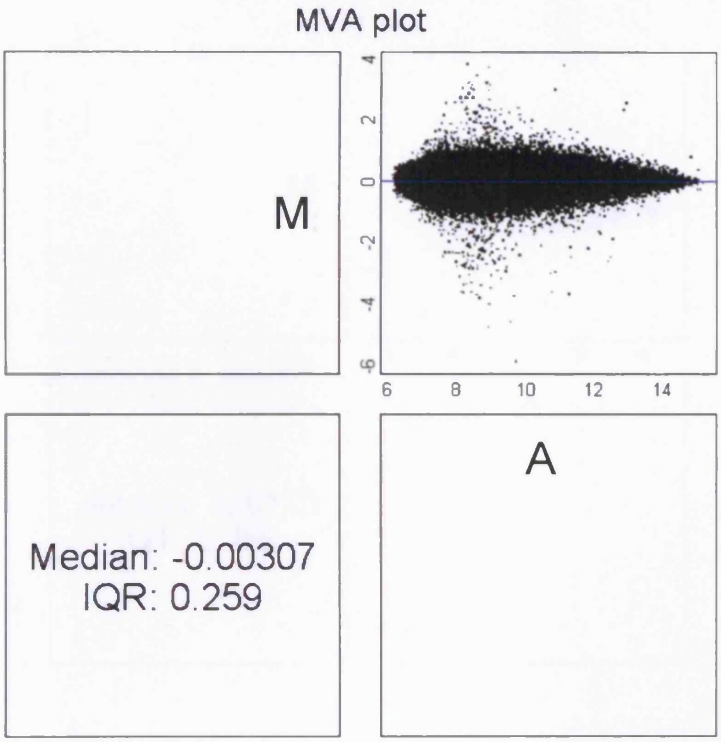


Figure 3.3: Good control 1 versus good control 2 from first batch of arrayed samples. These perform well having a lower IQR value.

Rigorous quality control at early stages of analysis ensures more robust results. Consequently, all biological work involved the same technician (R.M) who extracted both the original RNA from the control samples and performed the RNA extraction for the new control samples. The sample was also re-arrayed by the same operator.

MVA plots produced in I-10 to evaluate the array quality can be generated as shown in Figures 3.1 through 3.5. A lower IQR value and smaller deviation in the mean value indicates the sample which was re-arrayed and more accurate in the second batch than the first samples arrays.

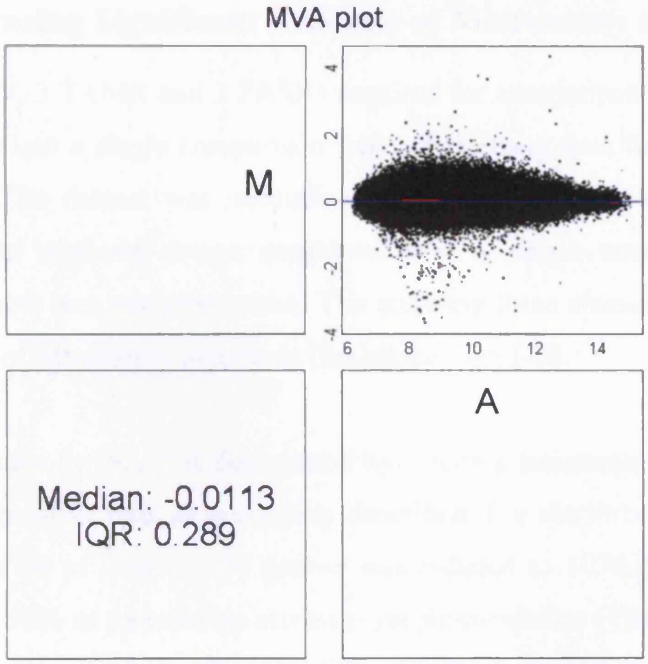


Figure 3.4: Newly prepared control 1 in comparison to the older control 1 which performed to a satisfactory level.

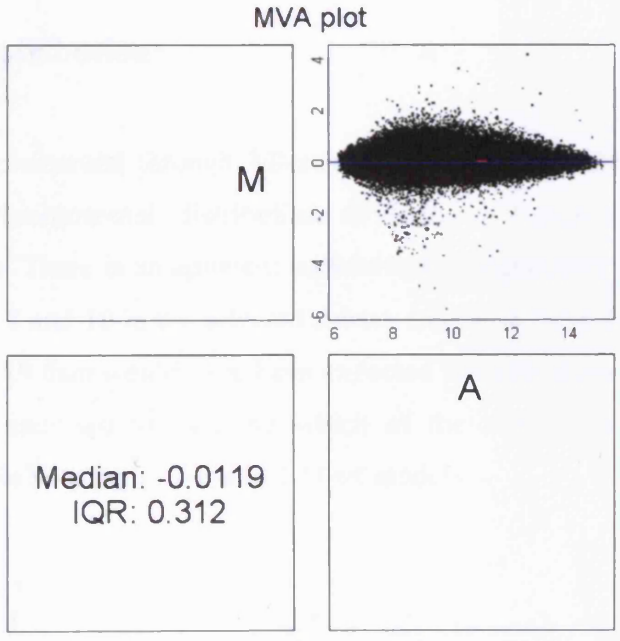


Figure 3.5: Newly prepared control in comparison to the satisfactory original control 2 showing improved IQR results.

3.4 Data Reduction using Significant Analysis of Microarray through I-10

The nine arrays (3 MCF7, 3 TAMR and 3 FASR) required for comparison were generated from the Affymetrix database into a single comparison table of intensity data for each of the 22,000 probes under analysis. The dataset was normalised for each probe by log 2 transformation, median centred and the triplicate arrays combined into a single normalised mean value representing each treatment arm per gene probe. The resulting three classes were reduced using the Significant Analysis of Microarray algorithm (SAM) through I-10.

The cut off for significance for SAM is determined by a tuning parameter delta, chosen by the user based on the false positive rate, as previously described. For the three way analysis in this project, the entire probe set of over 22,000 probes was reduced to 1070 probes setting a false discovery rate (FDR) of 10% as directed by literature recommendation (Tusher et al, 2001) [38]. This represented the master set of significantly differentially expressed probes for this project. The probes are rank ordered in I-10 according to p-value. The complete list of differential probes with mean normalised intensity values can be found on the accompanying CD-ROM.

3.5 Chromosomal Distribution

The SAM algorithm implemented through Microsoft Excel in I-10 can also give associated information regarding chromosomal distribution as seen in Figure 3.6 – comparison of chromosomal distribution. There is an apparent enrichment of significant differential expression of genes on chromosome 7 and 10 in the selected subset. Likewise there seems less deregulation of genes on chromosome 19 than would have been expected from the gene set. The chromosomal distribution was further analysed to examine which of the significant genes revealed were induced or shared events in both the FASR and TAMR models.

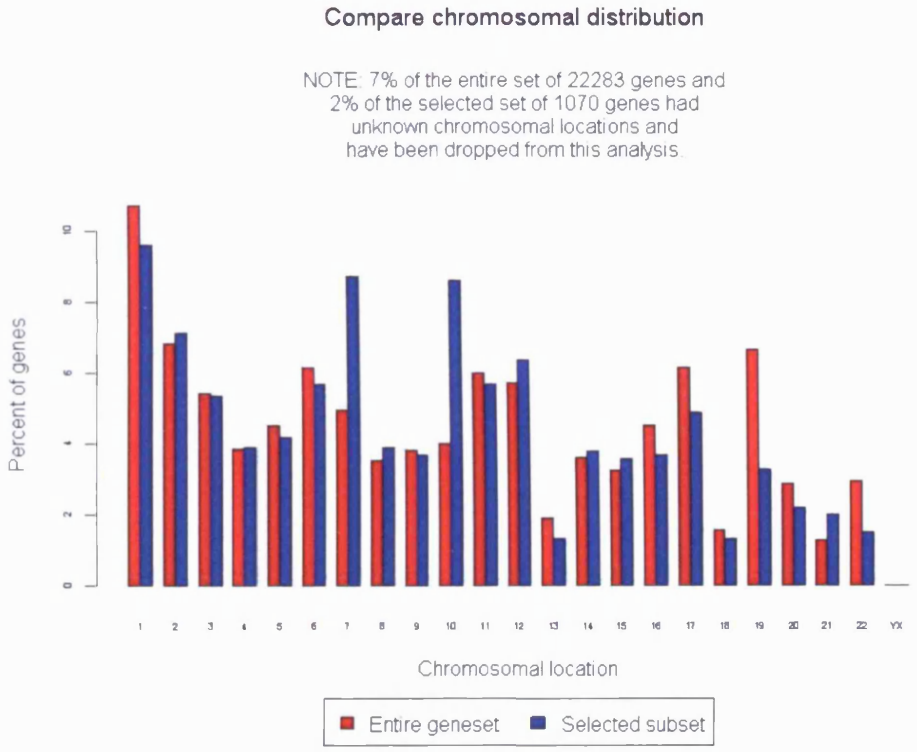


Figure 3.6 – Overall chromosomal distribution of the 1070 significant genes revealed from SAM analysis.

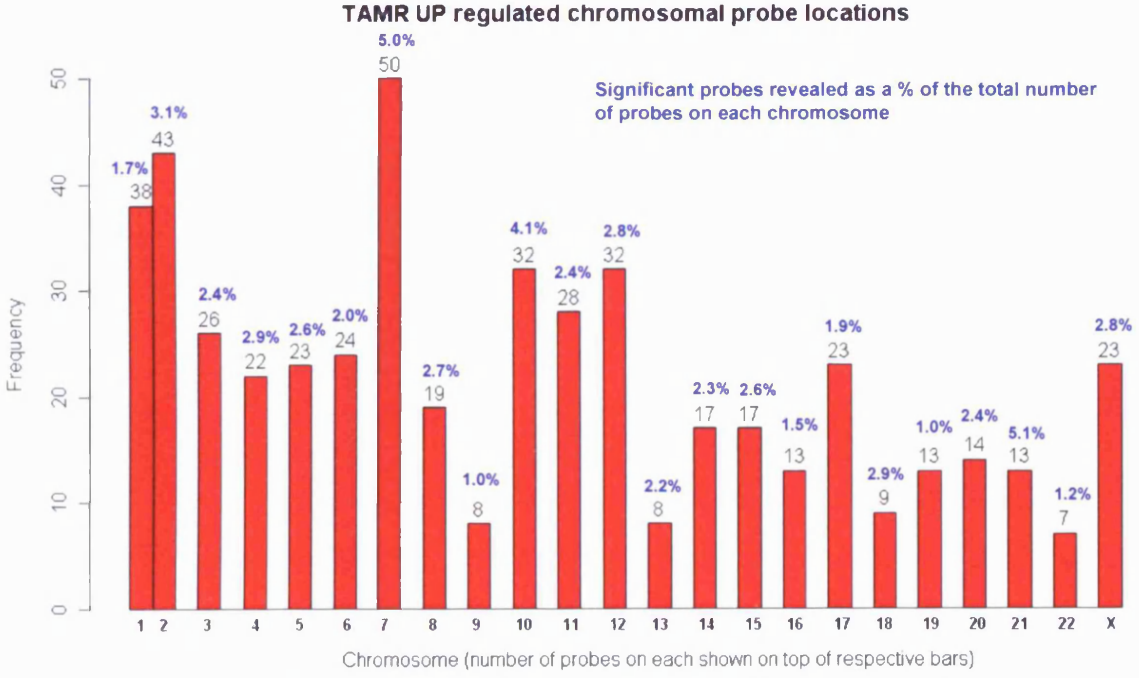


Figure 3.7 – TAMR model up regulated chromosomal distribution of significant Affymetrix probes for each individual chromosome as determined in relation to the control MCF7 model.

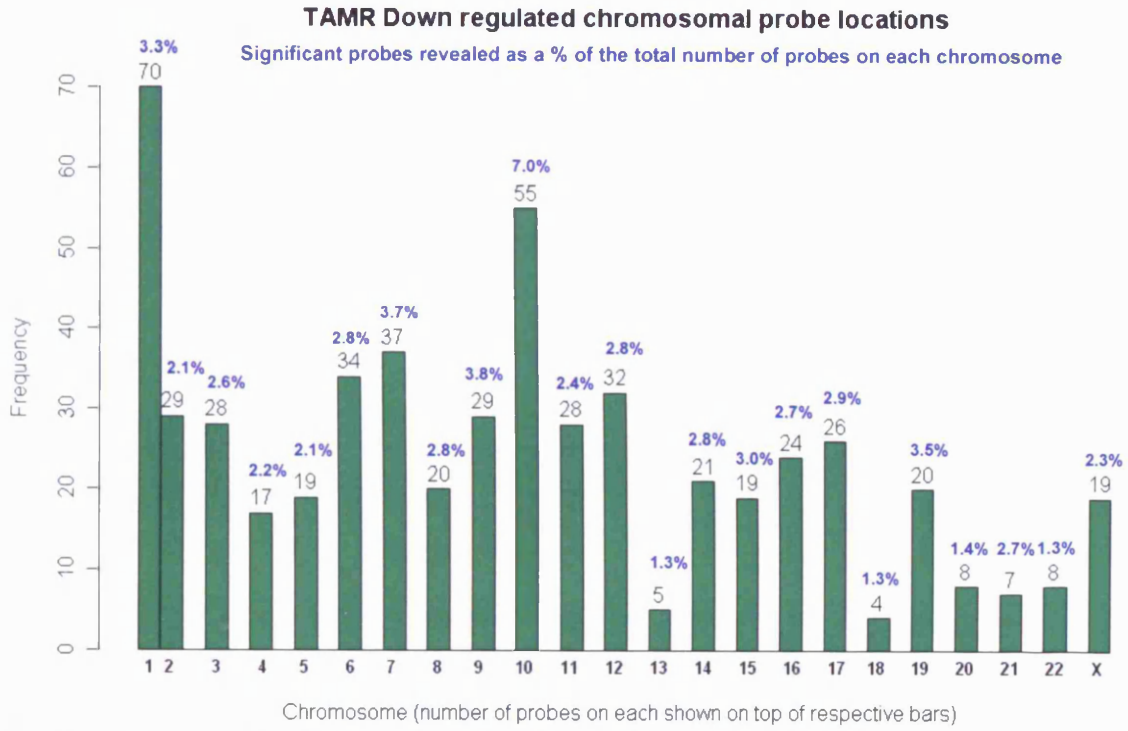


Figure 3.8 – TAMR model down regulated chromosomal distribution of significant Affymetrix probes for each individual chromosome as determined in relation to the control MCF7 model.

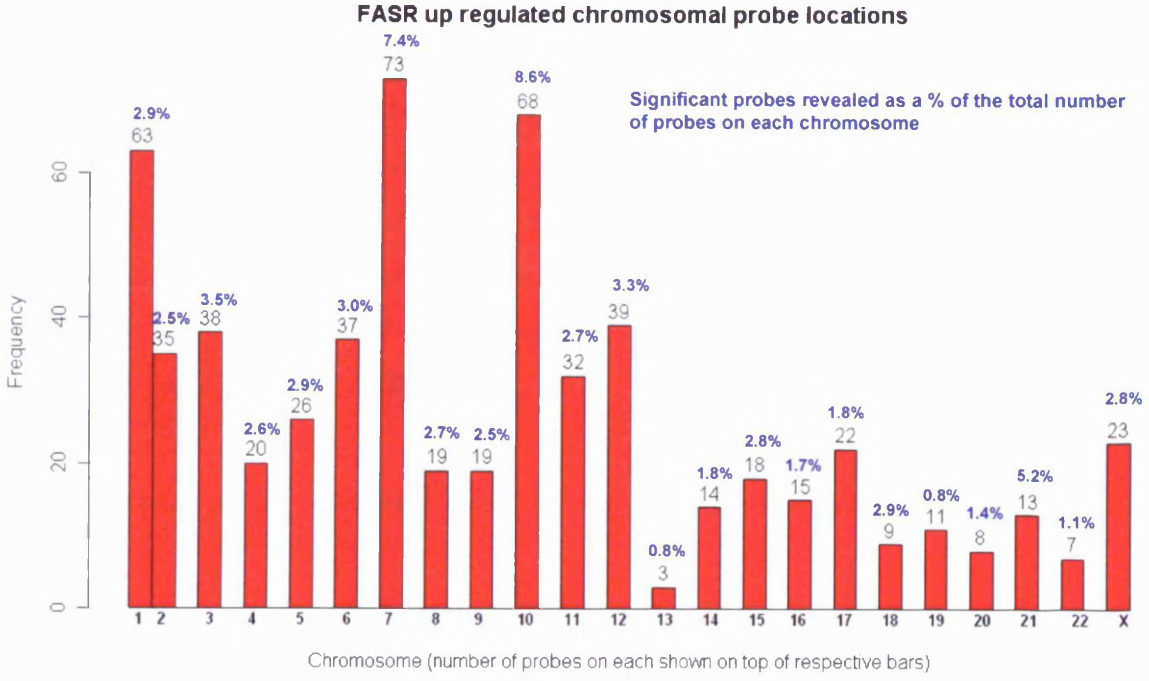


Figure 3.9 – FASR model up regulated chromosomal distribution of significant Affymetrix probes for each individual chromosome as determined in relation to the control MCF7 model.

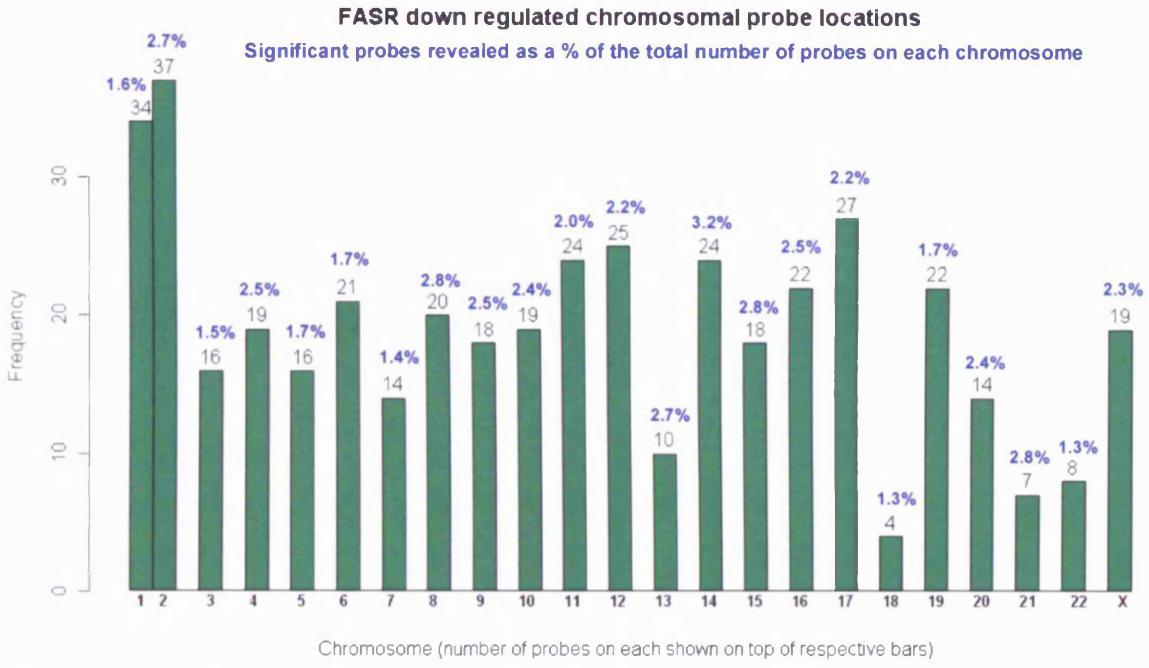


Figure 3.10 – FASR model down regulated chromosomal distribution of significant Affymetrix probes for each individual chromosome as determined in relation to the control MCF7 model.

The gene deregulations (n=87 in both instances) associated with chromosome 7 or chromosome 10 in the resistant cells versus the MCF-7 cells were further deciphered using additional analysis and histogram capabilities achieved through "R". This revealed that a predominance of the expression changes associated with chromosome 7 were increases, most prominently in the FASR model (84% increases, vs 54% increases for TAMR cells). While for chromosome 10, 78% of the changes were again expression increases in FASR cells, 63% of the changes associated with this particular chromosome in TAMR cells proved to be expression decreases. This is shown in Figures 3.7 and 3.8 for induced and suppressed events in TAMR models and Figures 3.9 and 3.10 for FASR models.

3.6 – Exploration of Broad Clustering Trends in the data using I-10

Early analysis of the 1070 differentially expressed probes was explored using the three dimensional plotting analysis tool within I-10. Figure 3.11 shows the FASR samples to exhibit the most changed probes – skewing the representations towards the FASR axis of the 3D plot. An assessment of up and down regulation of individual probes can be found later in the chapter.

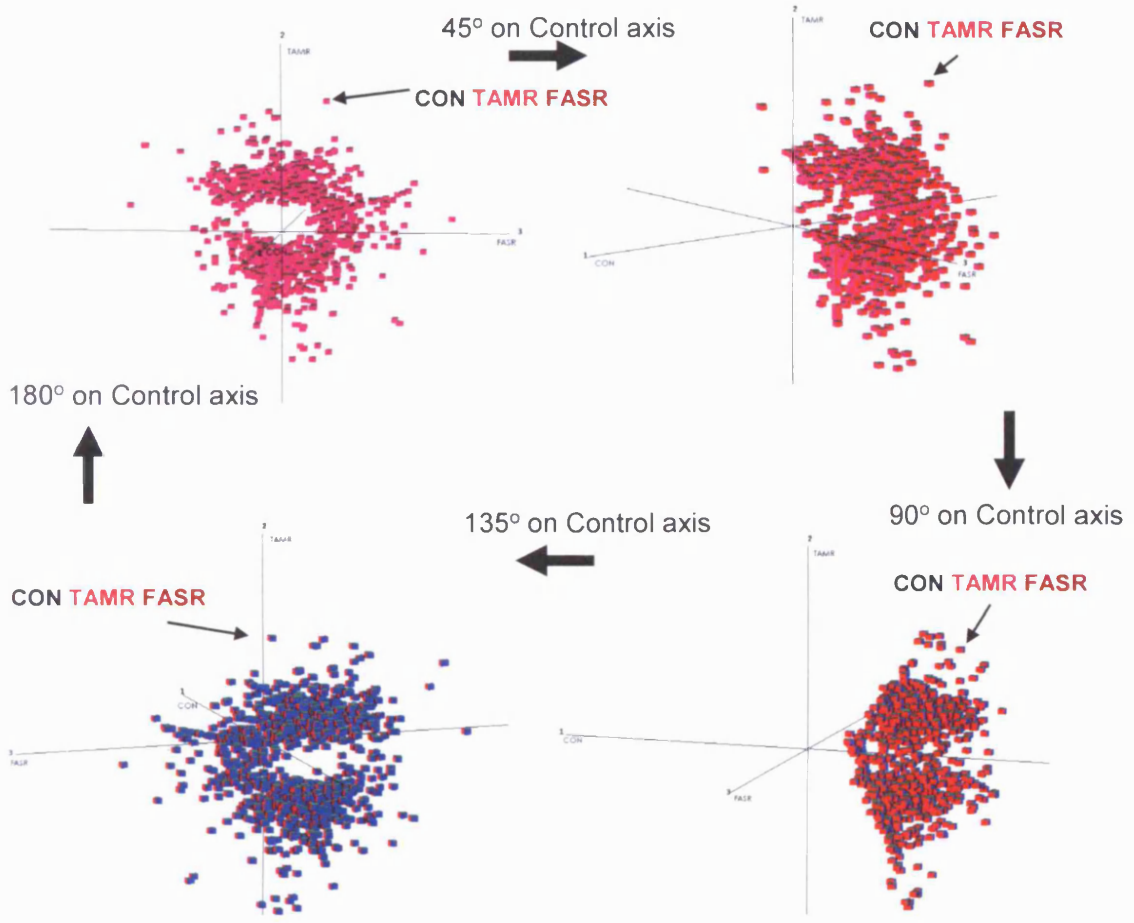


Figure 3.11 A three dimensional assessment showing a shift towards FASR model axis. Clockwise from the top left of the figure, the plot is turned 45 degrees along the control axis. The control axis is set to zero with the FASR and TAMR arms showing degree of fold change on each axis, one probe up regulated in TAMR(shown in red) and less up regulated in FASR (dark red).

Each of the two main clusters observed in the 3D scatter plot shown in figure 3.11 was further explored according to sub-clustering for pattern. For example, the analysis here focuses upon which genes are up regulated or down regulated in both FASR and TAMR models to try and identify consistent de-regulated generic to anti-oestrogen resistance. It is clear from the 3D plot that there is a skew towards the FASR arms.

3.7 Optimal Clustering Method Assessment

Selection of optimal clustering method to reveal patterns in data revealing expression profiles of interest can be difficult. Additionally, determination of the number of clusters that is most appropriate for the data under test to be used for optimal clustering and therefore pattern separation can be difficult to assess. A particular technique and cluster number should result in clusters which not only exhibit good statistical properties (compact, related and stable), however also give results that are biologically relevant (for example, GO class enrichment, pathway enrichment). Validation of the cluster method can be based solely on the internal properties of the data or on the expression data alone or in conjunction with relevant biological information. The clValid functionality in I-10 uses these properties to assess which clustering algorithm (for example HCA, PAM, K-Means) and clustering number best explores the given dataset.

Three types of cluster validation in ClValid are offered - "internal", "stability" and "biological". Internal validation uses intrinsic information in the probe data to assess the quality of the clustering taking only the dataset and the partition achieved in significant clusters as input. The stability measures are an advanced version of internal measures: they differ in that they evaluate the consistency of a clustering result by comparing it with the clusters obtained after each sample (MCF7 or TAMR or FASR) is removed sequentially. To achieve biologically meaningful clusters, biological validation evaluates the ability of a clustering algorithm to purify gene sets in comparison to GO ontology against the dataset. Consequently clValid can investigate both the biological homogeneity and stability of the clustering results.

3.7.1 Internal validation of clustering methods

For internal validation, the clValid system measures for each clustering technique the compactness, connectedness, and separation of the cluster partitions. Connectedness or connectivity relates to the extent observations are placed in the same cluster as their nearest neighbours in the data space (Handl et al, 2005) [106]. Compactness assesses cluster homogeneity, usually by looking at the intra-cluster variance, while separation quantifies the degree of separation between clusters (usually by measuring the distance between cluster

centroids). Since compactness and separation demonstrate opposing trends (compactness increases with the number of clusters but separation decreases), popular methods combine the two measures into a single score. The dunn index and silhouette width are both examples of non-linear combinations of the compactness and separation, and with the connectivity comprise the three internal measures available in clValid, all displayed graphically vs increased number of clusters for each clustering technique comparison applied to the data under test (Brock et al, 2008) [107].

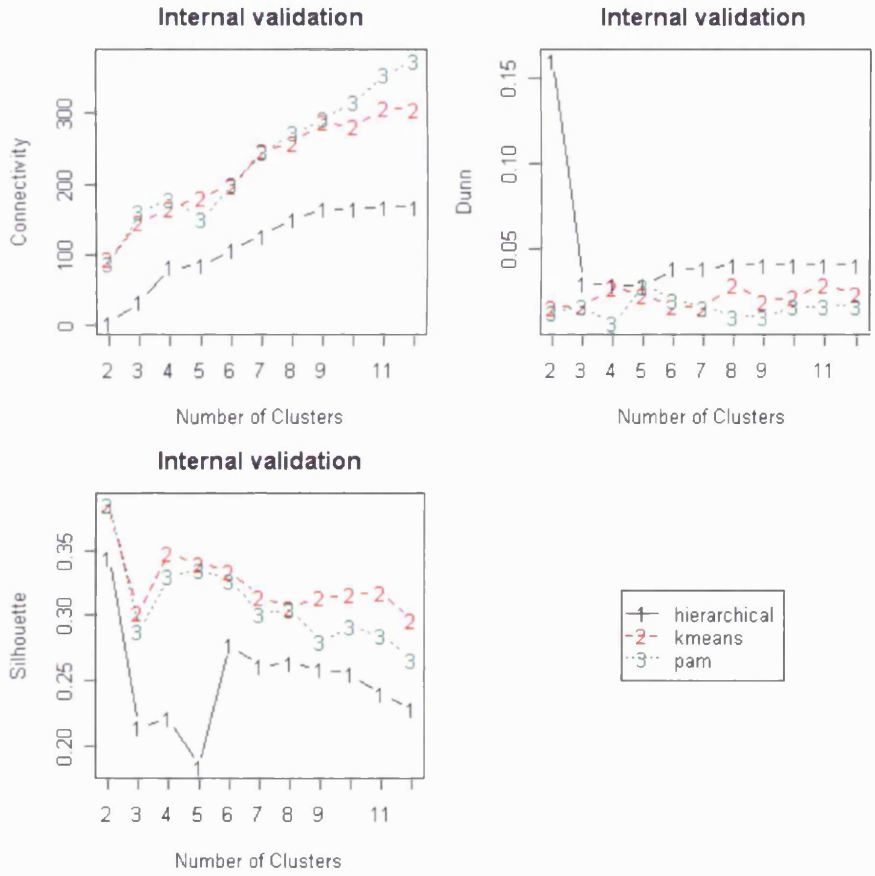


Figure 3.12 Internal validation of hierarchical, K-Means and PAM clustering applied to the 1070 significant genes using clValid library through I-10.

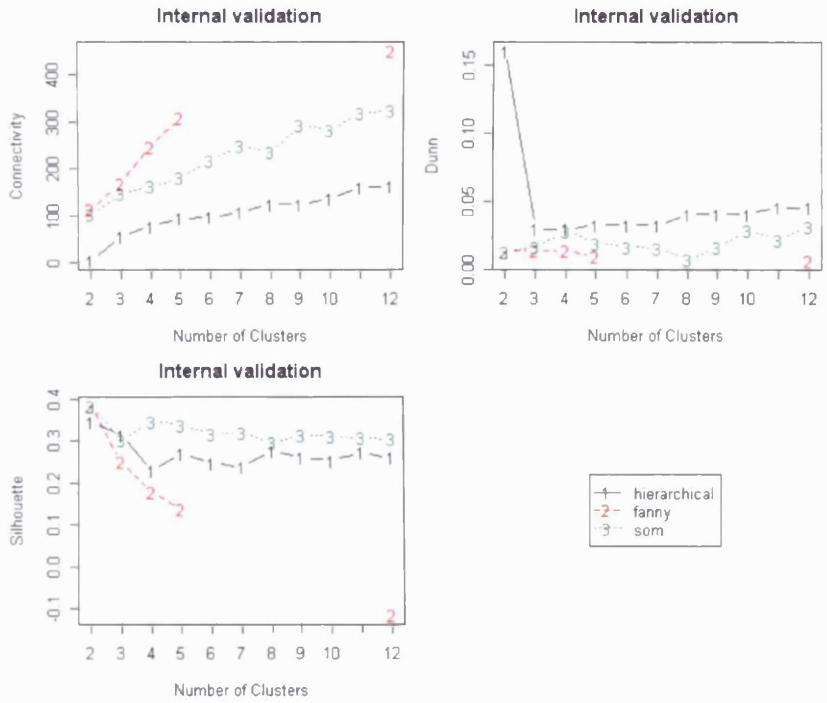


Figure 3.13 Internal validation of hierarchical, fanny and SOM clustering applied to the 1070 significant genes using clValid library through I-10.

	Score	Method	Clusters
Connectivity	2.9290	Hierarchical	2
Dunn	0.1604	Hierarchical	2
Silhouette	0.3849	SOM	2

Table 3.1 Comparison of internal measures and most applicable method summarised by clValid.

As seen in table 3.1, connectivity has a value between 0 and infinity and should try to be minimised, high Dunn index and high silhouette value is desirable. When comparing the five clustering algorithms, hierarchical clustering performed best in terms of connectivity and Dunn index and self organising maps (SOM) in terms of silhouette width. However as seen in figures 3.12 and 3.13, although the results are not exceptional, the optimal cluster number of 2 in the analysis suggests the system is only picking the broad two main clusters – induced and suppressed events and not the more subtle events occurring. Further analysis of the clustering is required. The next step was to analysis the stability of the clustering methods.

3.7.2 – Stability of clustering methods

The stability measures are a special version of the internal measures where the result from clustering based on the full data to clustering based on removing data for each treatment arm, one at a time. These methods work especially well if the data is highly correlated, as is the case in high-throughput microarray data. The stability includes the average proportion of non-overlap (APN), the average distance (AD), the average distance between means (ADM), and the figure of merit (FOM) (Brock et al, 2008) [107]. In all cases the average is taken over all the deleted treatment arms, and a stability measure should be small if clusters are stable.

The APN measures the average proportion of observations not placed in the same cluster by clustering based on the full data and clustering based on the data with a single arm removed. The APN is in the interval 0 to infinity with values close to zero corresponding with highly consistent clustering results (Brock et al, 2008) [107].

The AD measure computes the average distance between observations placed in the same cluster by clustering based on the full data and clustering based on the data with a single arm removed. The AD has a value between 0 and infinity, and smaller values are preferred although not essential (Brock et al, 2008) [107].

The ADM measure computes the average distance between cluster centres for observations placed in the same cluster by clustering based on the full data and clustering based on the data with a single column removed. ADM only uses the Euclidean distance in the current implementation of clValid. It also has a value between 0 and infinity - again smaller values are preferred (Brock et al, 2008) [107].

Finally the FOM measures the average intra-cluster variance of the observations in the deleted column, where the clustering is based on the remaining (undeleted) samples. This estimates the mean error using predictions based on the cluster averages.

As shown in table 3.2, the given dataset split into 12 clusters with Pam and 2 for the older implementation of a similar algorithm - K-Means. Up to 20 were tested however no great advantage was shown beyond 12.

	Score	Method	Clusters
APN	0.1762269	K-Means	2
AD	2.7860434	PAM	12
ADM	0.9656024	K-Means	2
FOM	1.7561138	PAM	12

Table 3.2 – Stability scores with K-Means and PAM performing best out of the three clustering techniques including hierarchical clustering.

This is the first result to suggest the existence of 12 clusters in the dataset as well as the 2 broad clusters as previously described.

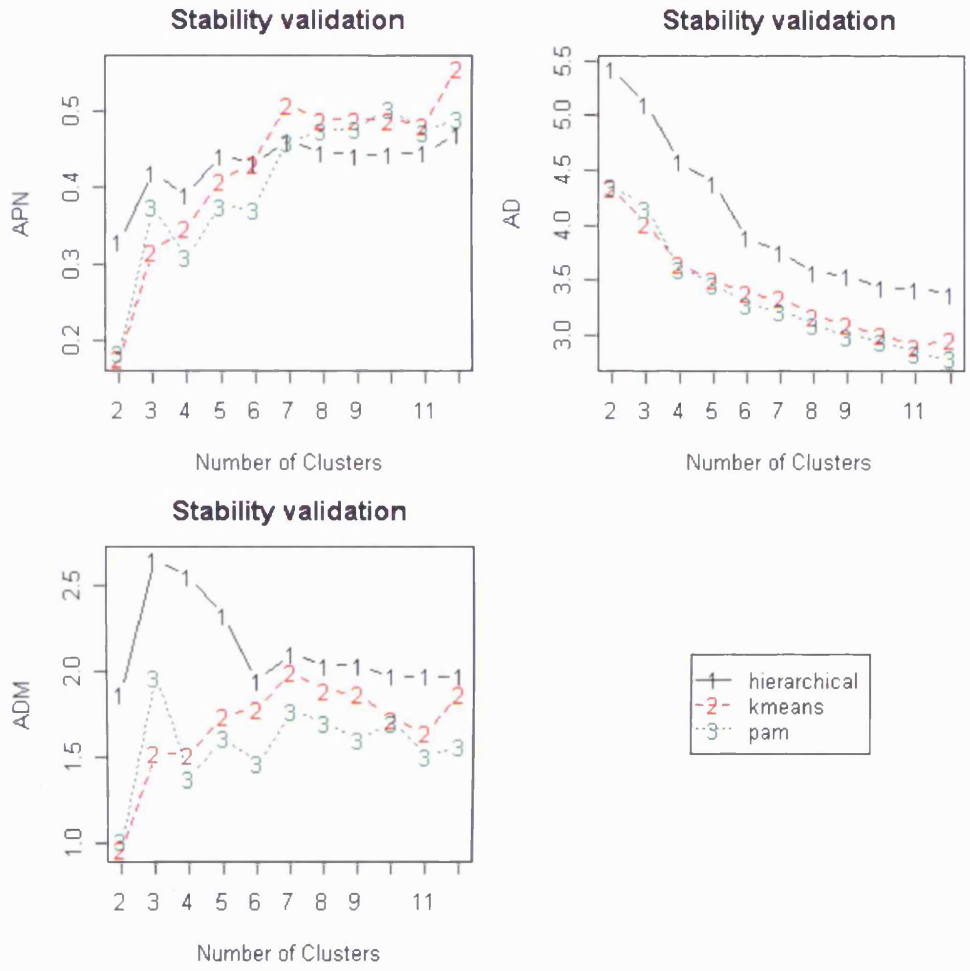


Figure 3.14 – Stability measures of hierarchical, K-Means and PAM clustering using the 1070 significant genes revealed.

Figure 3.14 highlights the similarity of K-Means and PAM clustering techniques with a very similar profile in terms of APN, AD and ADM measure with a greater increase in performance shown by all the measures. Hierarchical clustering starts to perform better at the higher cluster number – particularly approaching 12 clusters. However overall it is clear why the system chose PAM as the best performer.

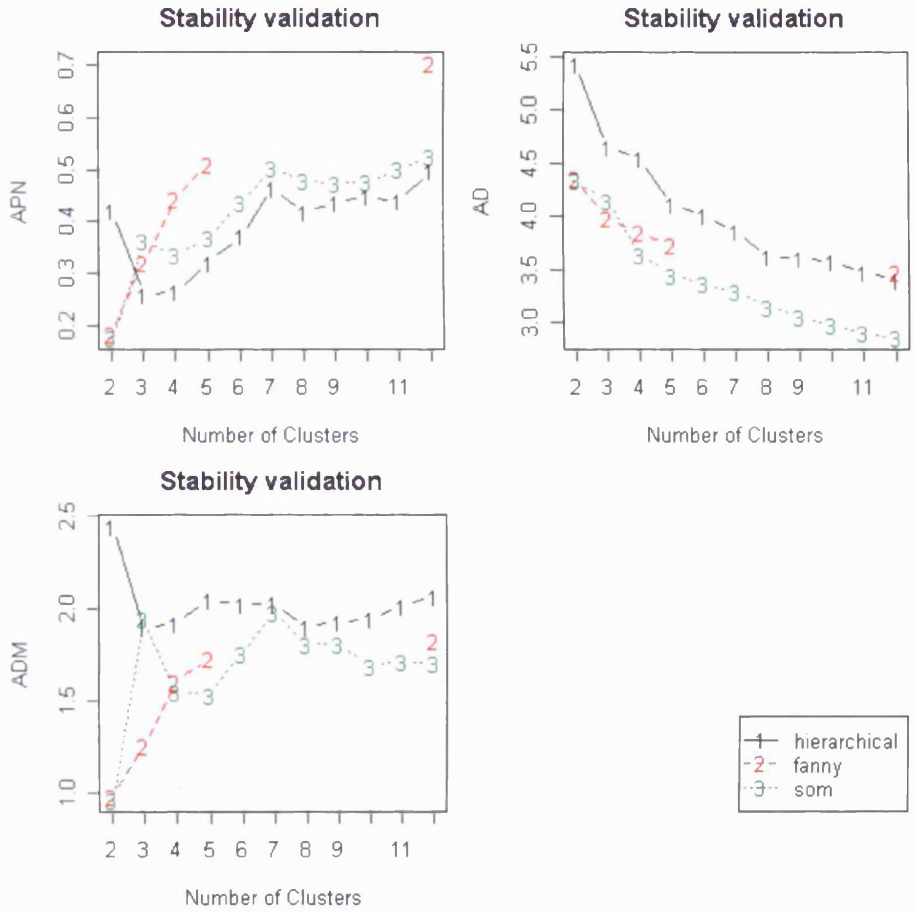


Figure 3.15 – Stability measures of hierarchical, Fanny and SOM clustering of the 1070 significant genes.

	Score	Method	Clusters
APN	0.1755	SOM	2
AD	2.8574	SOM	12
ADM	0.9572	SOM	2
FOM	1.7765	SOM	12

Table 3.3 – Stability scores with SOM comparing best against hierarchical clustering

As summarised in table 3.3, self organising maps perform better than HCA and the FANNY (fuzzy clustering) algorithm, with the latter giving no result at certain cluster numbers due to computational limits during the run. It was observed again that hierarchical clustering performed better at higher cluster numbers.

3.7.3 – Biological validation

Biological validation available through I-10 as part of the cl-Valid library, evaluates the ability of a clustering algorithm to produce biologically related clusters. This technique has been optimised for microarray data, where observations correspond to genes (where "genes" could be open reading frames (ORFs) and represented by probes, express sequence tags (ESTs), serial analysis of gene expression (SAGE) tags, etc.). There are two measures available, the biological homogeneity index (BHI) and biological stability index (BSI) (Brock et al, 2008) [107].

The BHI measures how homogeneous the clusters are in relation to biological classes based on GO classification. The algorithm has been designed such that it explores if genes placed in the same statistical cluster also belong to the same functional classes. The BHI has a range of 0 to 1 with larger values corresponding to more biologically homogeneous clusters (Brock et al, 2008) [107].

The BSI is similar to the previous stability measures, and inspects the consistency of clustering for genes with similar biological functionality. Each treatment arm is removed one at a time, and the cluster membership for genes with similar functional annotation is compared with the cluster membership using all available samples. The BSI has a range of 0 to 1 with larger values corresponding to more stable clusters of the functionally annotated genes (Brock et al, 2008) [107].

	Score	**Method**	**Cluster**
BHI	0.1801512	Hierarchical	10
BSI	0.5778908	Hierarchical	2

Table 3.4 – Hierarchical clustering performs the best with optimal number of biological clusters being 10 and 2 depending on the measure observed when compared to K-Means and PAM.

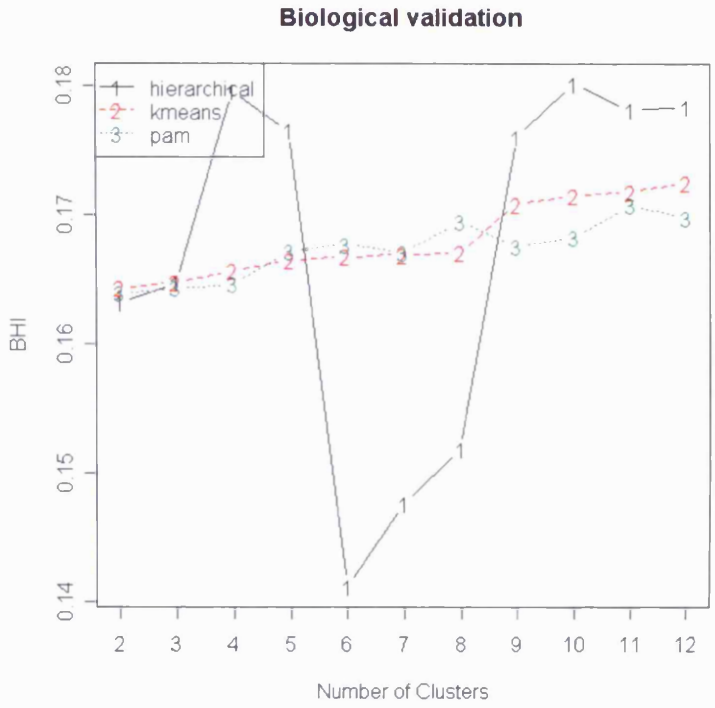


Figure 3.16 – Biological validation with BHI number of hierarchical clustering, K-means and PAM using the 1070 significant gene list.

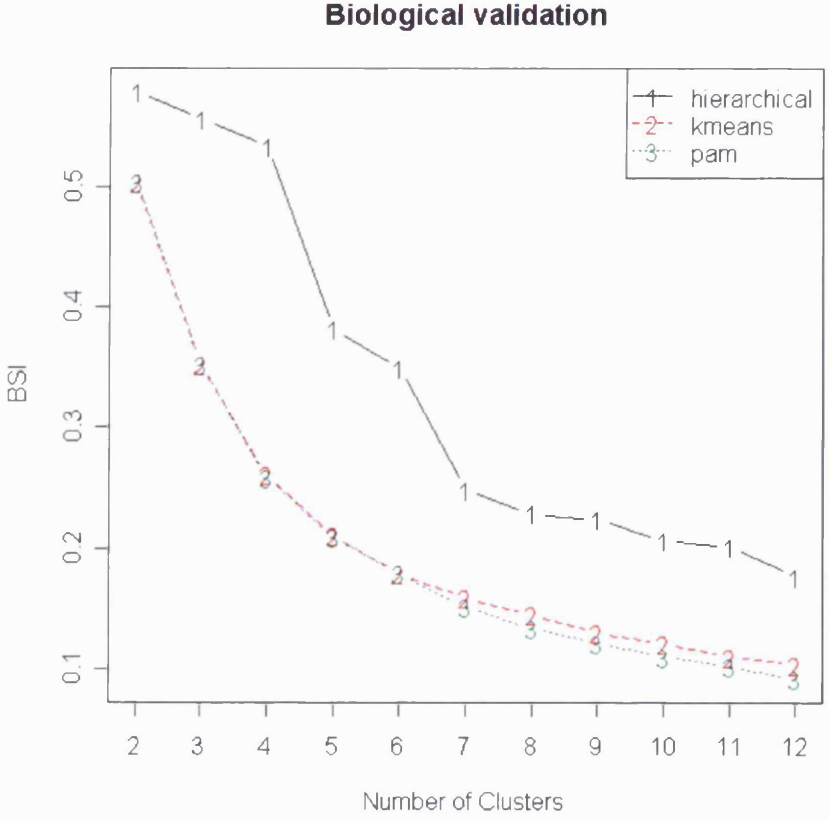


Figure 3.17 – Biological validation with BSI value of hierarchical clustering, K-means and PAM using the 1070 significant gene list.

	Score	Method	Clusters
BHI	0.1868	fanny	12
BSI	0.5779	hierarchical	12

Table 3.5 – Summary of the best performing algorithms when comparing scores of hierarchical, fanny and SOM of the 1070 significantly revealed genes.

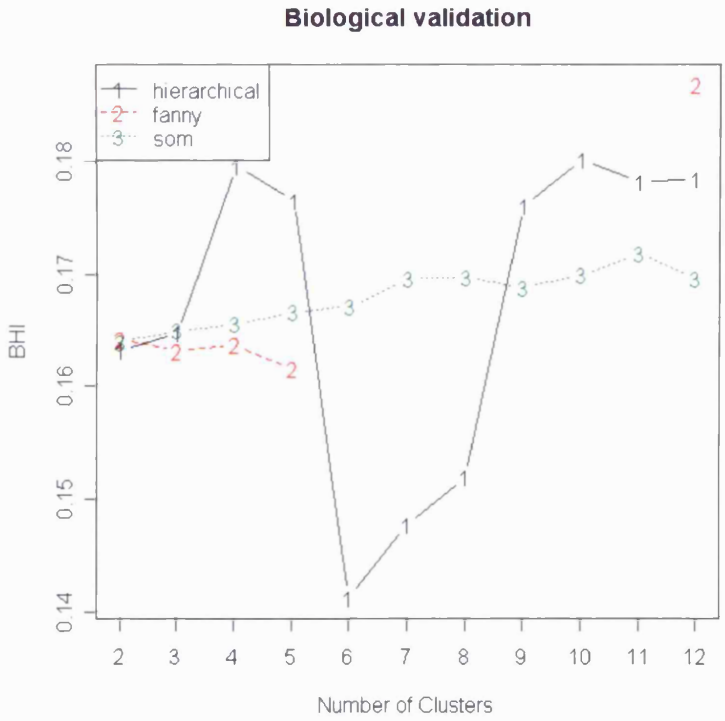


Figure 3.18 – Biological validation - BHI value comparing hierarchical, Fanny and SOM of hierarchical clustering, fanny and SOM using the 1070 significant gene list.

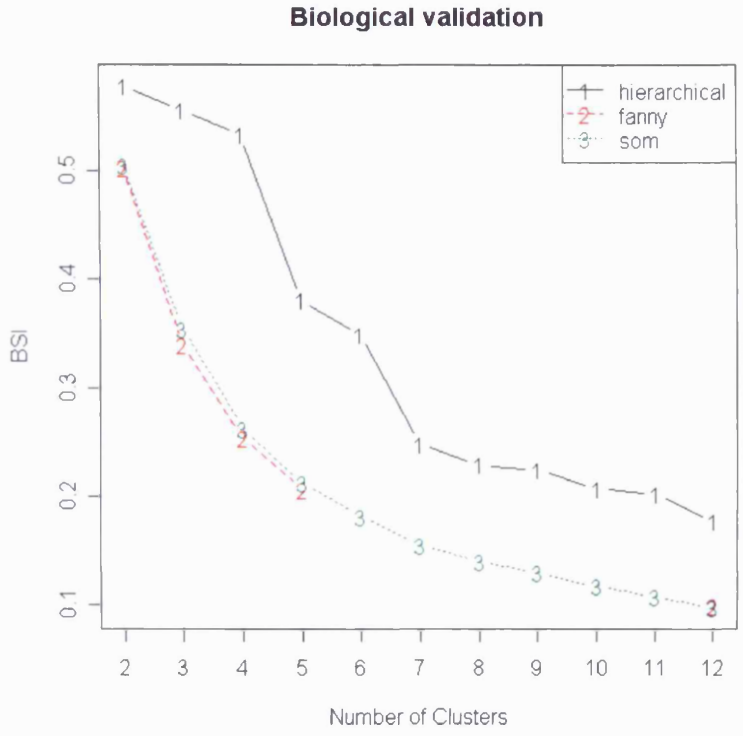


Figure 3.19 – Biological validation with BSI value when comparing hierarchical, Fanny and SOM using the 1070 significant gene list.

When examining the clustering techniques in terms of Biological functional analysis and clustering GO ontological terms by function, hierarchical clustering performs well. It is also interesting to note the sharp drop in BHI value at the six cluster point which could suggest there are six strong profiles present in the data. However 12 is deemed the optimal number of clusters ultimately based on the results in terms of consistency taking all factors across all validation methods and the values returned into account.

3.8 – Exploring Hierarchical Clustering Membership

3.8.1 – Broad clustering of dataset and subsequent sub clustering to reveal patterns and robust gene changes

All samples were then clustered for overall degree of similarity. The TAMR arm was deemed by HCA to be most similar to control while FASR was the most different. This was also inferred in the visual representation of PCA in preliminary analysis. Twelve clusters comprise an optimal

number in the dataset based on analysis with pvClust, and these clusters have been studied more closely.

Cutting the tree at the same level as that revealed by pvClust produces a heat map coloured to clearly show the 12 different clusters shown in figure 3.20. The HCA heat map has been coloured using a recommended Bioconductor heatmap colour gradient red for up regulation, black for no change and green for down regulation based on the given dataset. The exact membership of each can be extracted in I-10 to a file listing each Affymetrix probe ID per cluster.

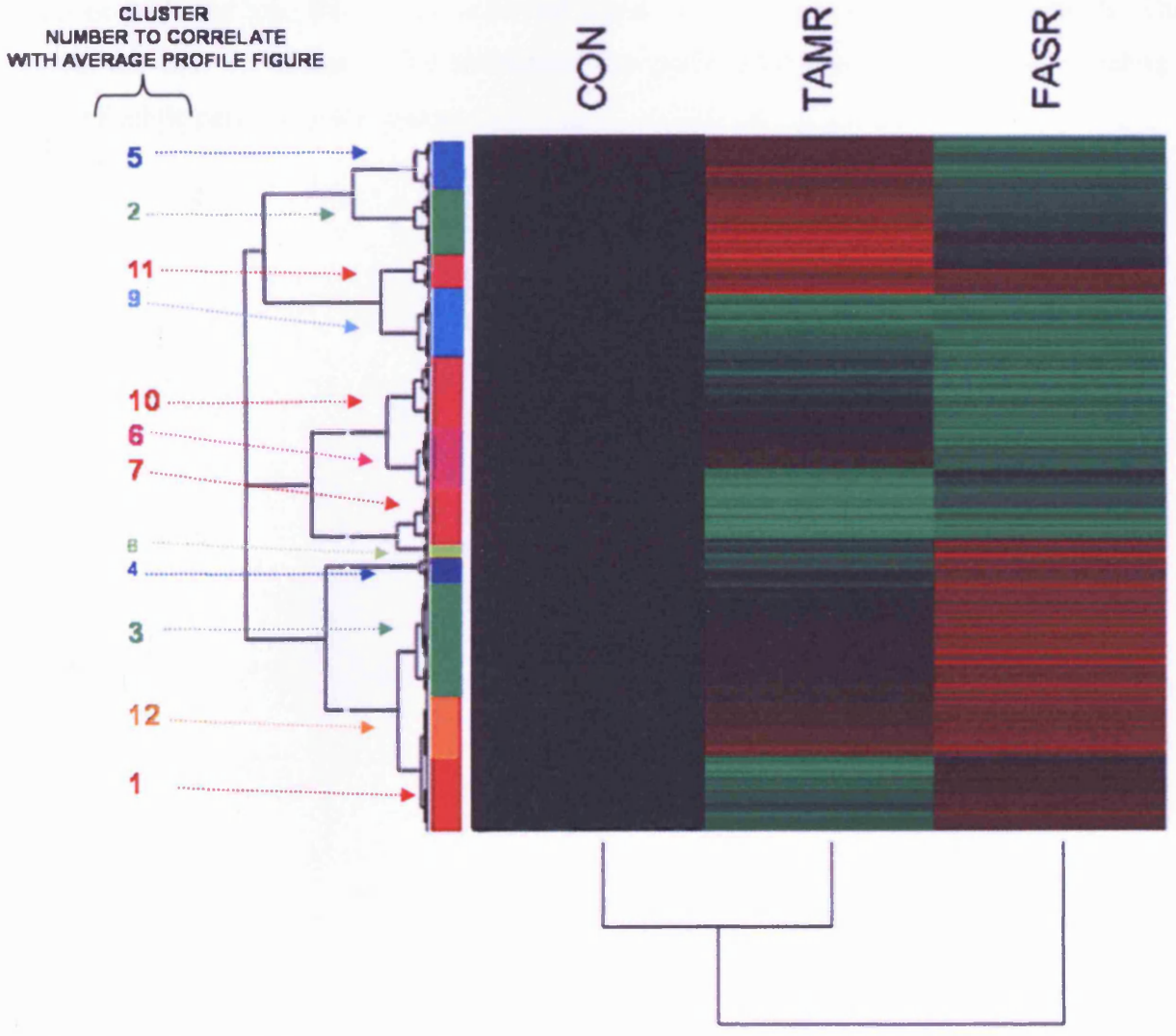


Figure 3.20 Hierarchical clustering heat map recreated showing 12 clusters with the addition of a rainbow colour scheme for cluster differentiation on the y axis cut a the same level as HCA performed by ClValid. The three sample replicates were averaged for the purposes of the figure which indicates that Control and TAMR being most similar followed by FASR samples being the most different from either group when clustered on the x axis. No fold change cut-off was chosen.

Although there are 12 defined patterns within each of the 12 clusters, each profile varied slightly within each cluster, however overall they adhered to a general trend as shown by the overall heat map in figure 3.20. Therefore by assessing each of the 12 clusters using the profile viewer, a class representing each profile - either greatly induced or suppressed versus the control model can be developed. Figure 3.21 highlights the profiles present and the decided classes of up and down regulation across the three models. Seven combinations were present. An unchanged event is classified as one without a change of at least '1' in either direction. Interestingly, a seventh class correlates with the low BHI value achieved when biologically validating the methods. This suggests the BHI validation of the techniques was particularly strong in terms of revealing a degree of subtle patterns in the dataset.

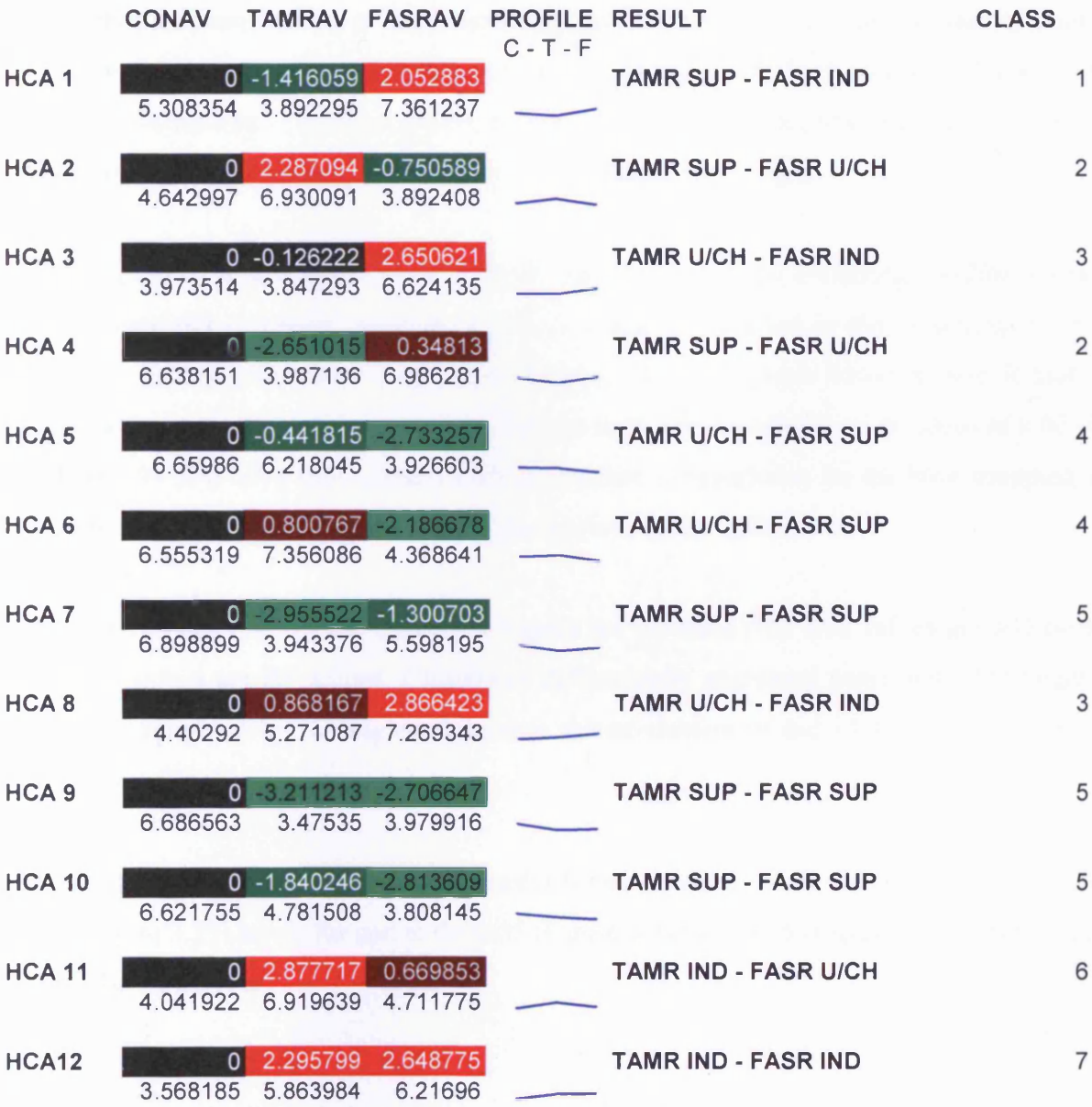


Figure 3.21: A summary of the different profiles comprising the 12 cluster across all treatment arms. Events which were shared in both FASR and TAMR models are clusters HCA 9 and HCA 12 and to a lesser extent HCA 7 and 10. The class column to the right of the figure broadly tries to group the profiles by broad similarity. A value between 0 and +/- 1.0 is classified as unchanged (U/CH).

Using the pvClust module in I-10, uncertainty can be removed from hierarchical clustering. For each cluster in hierarchical clustering, quantities called p-values can be derived via multiscale bootstrap re-sampling.

This method implemented in I-10 provides two types of p-values: AU (approximately unbiased) p-value and BP (bootstrap probability) value. AU p-value, which is computed by multi-scale bootstrap re-sampling, is a better approximation to unbiased p-value than BP value computed by normal bootstrap re-sampling to determine most robust gene changes.

PvClust performs hierarchical cluster analysis via the hierarchical clustering function within 'R' and automatically computes p-values for all 12 clusters contained in the clustering of original data. I-10 displays the result in the same way as other functions however here it highlights clusters with relatively high/low p-values. For the three way comparison – p values of 0.05 (95%) and 0.005 (99.5%) have been chosen with the number of repetitions for the boot strapping set to 1000 which enhances p value calculations particularly at the 0.005 level.

Values on the branches of the clustering figures are p-values (%). Red values are AU p-values, and green values are BP values. Clusters of differentially expressed genes with AU larger than 95% are highlighted by rectangles; dominant shared clusters (9 and 12 from figure 3.26) have been explored in this way.

The significance of key probes within cluster 9 can be observed at the 0.05 (Figure 3.22) and 0.005 (Figure 3.23) levels for and at the 0.05 (Figure 3.24) and 0.05 (Figure 3.25) level for cluster 12 can also be derived.

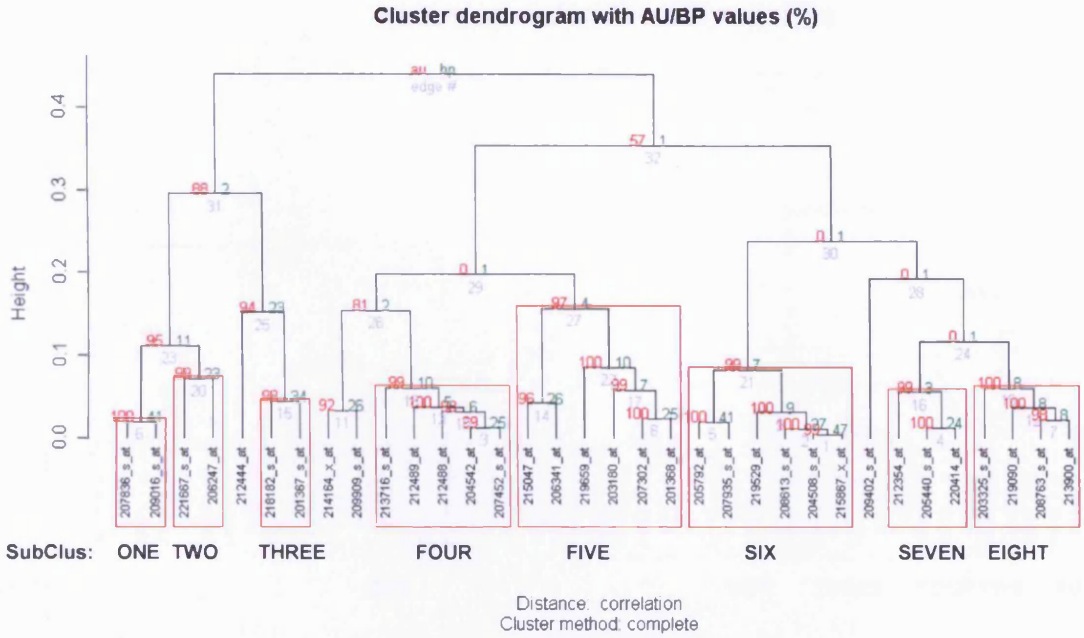


Figure 3.22 – Significance level of p=0.05 for cluster 9 – red boxes designate significant probes. The tree structure at higher levels indicates higher level functional relationship.

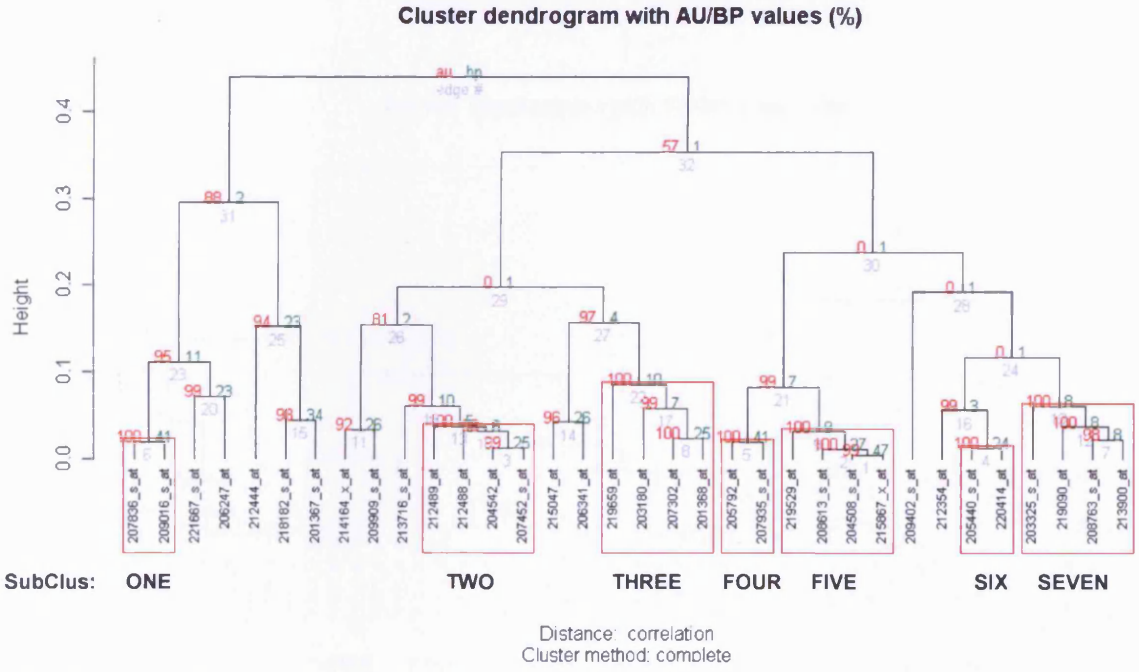


Figure 3.23 – Very high significance level p value=0.005 for cluster 9 – red boxes designate most significant probes. The tree structure at higher levels indicates higher level functional relationship.

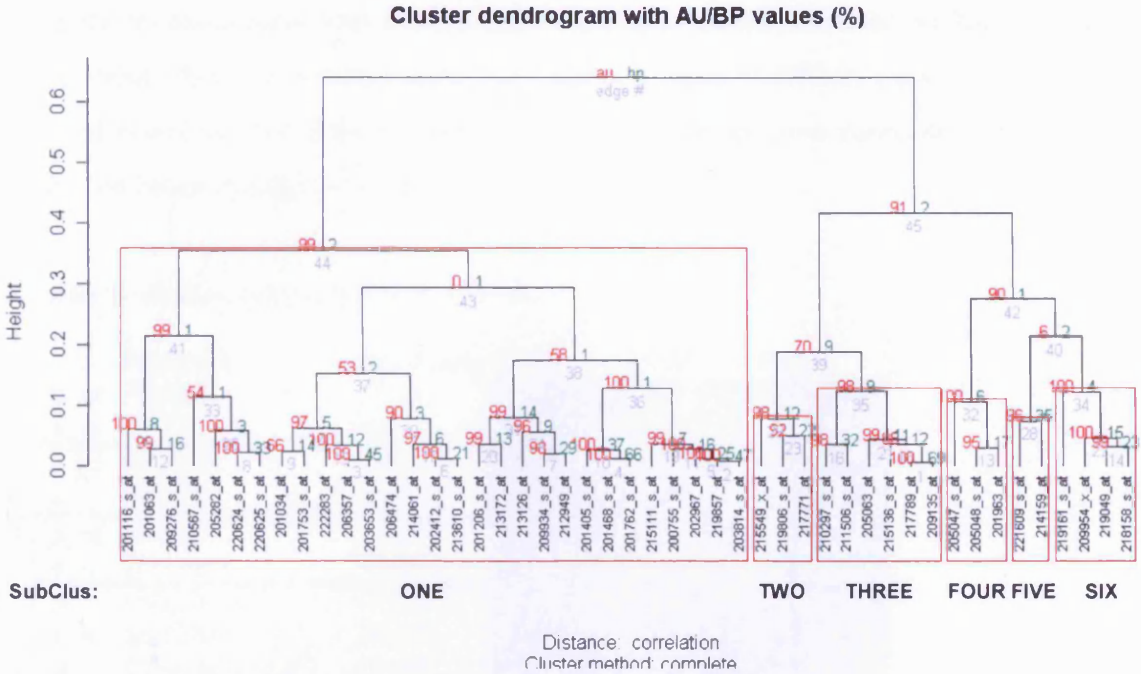


Figure 3.24 – Significance level=0.05 for cluster 12 – red boxes designate significant probes. The tree structure at higher levels indicates higher level functional relationship.

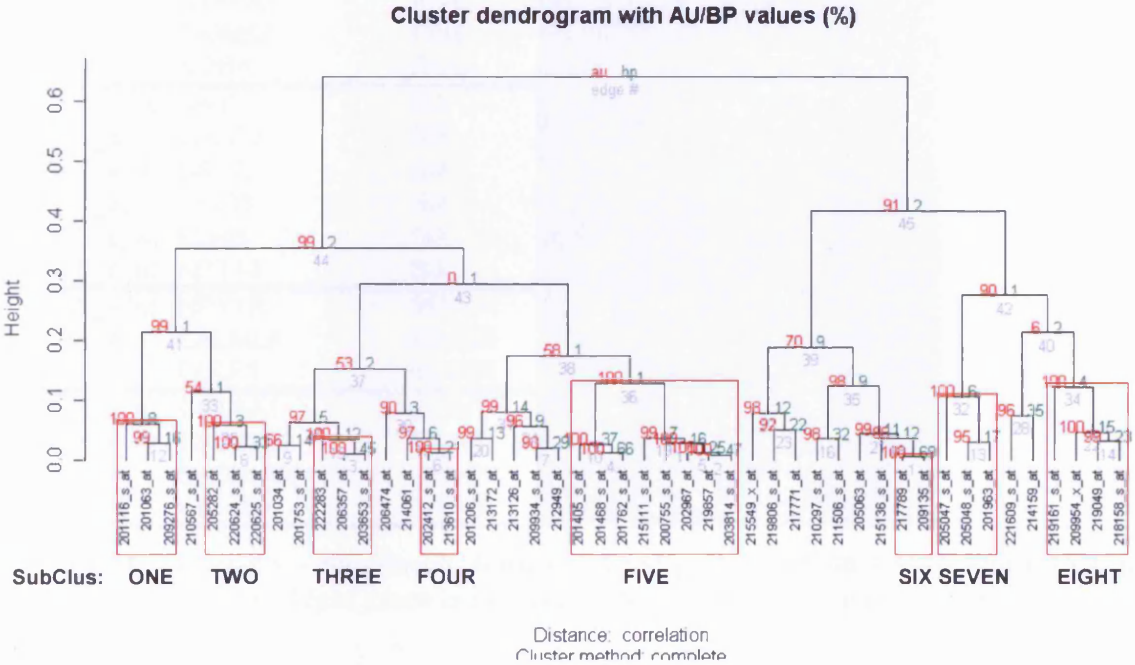


Figure 3.25 – Very high significance level p value =0.005 p of cluster 12 – red boxes donate most significant probes. The tree structure at higher levels indicates higher level functional relationship.

When significant assessment was determined for the individual probes, the profile of each gene at both the p value =0.05 shown in Figure 3.26 and a p value =0.005 in Figure 3.27 representing only the red boxed probes from cluster 9. The sub cluster column correlates with the clusters divided by red boxes in previous figures 3.22 through 3.25.

HCA Cluster 9 - 0.05 p value significance level

AffyID	Acronym	SubCluste	CON	TAMR	FASR
207836_s_at	RBPMS	ONE	0	-3.579025	-2.459565
209016_s_at	KRT7	ONE	0	-2.261502	-1.588827
221667_s_at	HSPB8	TWO	0	-5.316286	-3.599954
206247_at	MICB	TWO	0	-3.968358	-2.731069
218182_s_at	CLDN1	THREE	0	-3.335398	-2.478821
201367_s_at	ZFP36L2	THREE	0	-2.88672	-2.504559
212489_at	COL5A1	FOUR	0	-3.550189	-3.21827
213716_s_at	SECTM1	FOUR	0	-4.022166	-2.782263
204542_at	ST6GALNAC2	FOUR	0	-2.198318	-1.768169
207452_s_at	CNTN5	FOUR	0	-3.11053	-2.360759
212488_at	COL5A1	FOUR	0	-1.80763	-1.602539
207302_at	SGCG	FIVE	0	-5.835074	-4.361935
201368_at	ZFP36L2	FIVE	0	-2.879692	-2.144059
203180_at	ALDH1A3	FIVE	0	-3.753755	-2.699822
219659_at	ATP8A2	FIVE	0	-3.002358	-2.295868
215047_at	TRIM58	FIVE	0	-2.61527	-1.873355
206341_at	IL2RA	FIVE	0	-2.44934	-2.09349
204508_s_at	CA12	SIX	0	-4.203604	-4.211468
205792_at	WISP2	SIX	0	-5.091934	-4.947414
215867_x_at	CA12	SIX	0	-2.281661	-2.237671
219529_at	CLIC3	SIX	0	-2.993186	-2.76278
208613_s_at	FLNB	SIX	0	-2.026182	-2.041554
207935_s_at	KRT13	SIX	0	-2.459798	-2.416434
205440_s_at	NPY1R	SEVEN	0	-3.858283	-3.749856
220414_at	CALML5	SEVEN	0	-4.782678	-4.590468
212354_at	SULF1	SEVEN	0	-3.45099	-3.584589
203325_s_at	COL5A1	EIGHT	0	-3.80402	-3.046433
208763_s_at	TSC22D3	EIGHT	0	-2.607978	-2.138546
219090_at	SLC24A3	EIGHT	0	-2.125476	1.933362
213900_at	C9orf61	EIGHT	0	-1.554039	-1.440704

Figure 3.26 – cluster 9 – Significant down regulated probes in both TAMR and FASR at the 0.05 level vs MCF7 control. Light green indicates a higher degree of down regulation.

Cluster 9, as shown in figure 3.26, reveals genes which are currently understudy in other projects as suppressed genes. These include *CA12*, *WISP2* and *FLNB* both within significant Sub cluster 'SIX' as shown in figure 3.26. These targets are being explored as potential tumour suppressors

showing epigenetic silencing in TAMR cells. The present studies suggest the silencing also extends to FASR, along with the future genes in the list. *CA12*, *WISP2*, *FLNB* along with future candidates to be suppressed in resistant states.

HCA Cluster 9 - 0.005 p value significance level

AffyID	Acronym	SubCluster	CON	TAMR	FASR
207836_s_at	RBPMS	ONE	0	-3.579025	-2.459565
209016_s_at	KRT7	ONE	0	-2.261502	-1.588827
212489_at	COL5A1	TWO	0	-3.550189	-3.21827
204542_at	ST6GALNAC2	TWO	0	-2.198318	-1.768169
207452_s_at	CNTN5	TWO	0	-3.11053	-2.360759
212488_at	COL5A1	TWO	0	-1.80763	-1.602539
207302_at	SGCG	THREE	0	-5.835074	-4.361935
201368_at	ZFP36L2	THREE	0	-2.879692	-2.144059
203180_at	ALDH1A3	THREE	0	-3.753755	-2.699822
219659_at	ATP8A2	THREE	0	-3.002358	-2.295868
205792_at	WISP2	FOUR	0	-5.091934	-4.947414
207935_s_at	KRT13	FOUR	0	-2.459798	-2.416434
204508_s_at	CA12	FIVE	0	-4.203604	-4.211468
215867_x_at	CA12	FIVE	0	-2.281661	-2.237671
219529_at	CLIC3	FIVE	0	-2.993186	-2.76278
208613_s_at	FLNB	FIVE	0	-2.026182	-2.041554
205440_s_at	NPY1R	SIX	0	-3.858283	-3.749856
220414_at	CALML5	SIX	0	-4.782678	-4.590468
203325_s_at	COL5A1	SEVEN	0	-3.80402	-3.046433
208763_s_at	TSC22D3	SEVEN	0	-2.607978	-2.138546
219090_at	SLC24A3	SEVEN	0	-2.125476	-1.9[illegible]62
213900_at	C9orf61	SEVEN	0	-1.554039	-1.440704

Figure 3.27 – cluster 9 – Significant down regulated probes in both TAMR and FASR at the 0.005 level vs MCF7 control. Light green indicates a high degree of down regulation.

Cluster 9, as shown in figure 3.27, also contains future potential targets of interest for deregulation in resistance including *NPY1* receptor, as revealed later in the chapter following ontological enrichment. The close association of each probe can be observed in figure 3.28 when the corresponding probes are plotted in 3D.

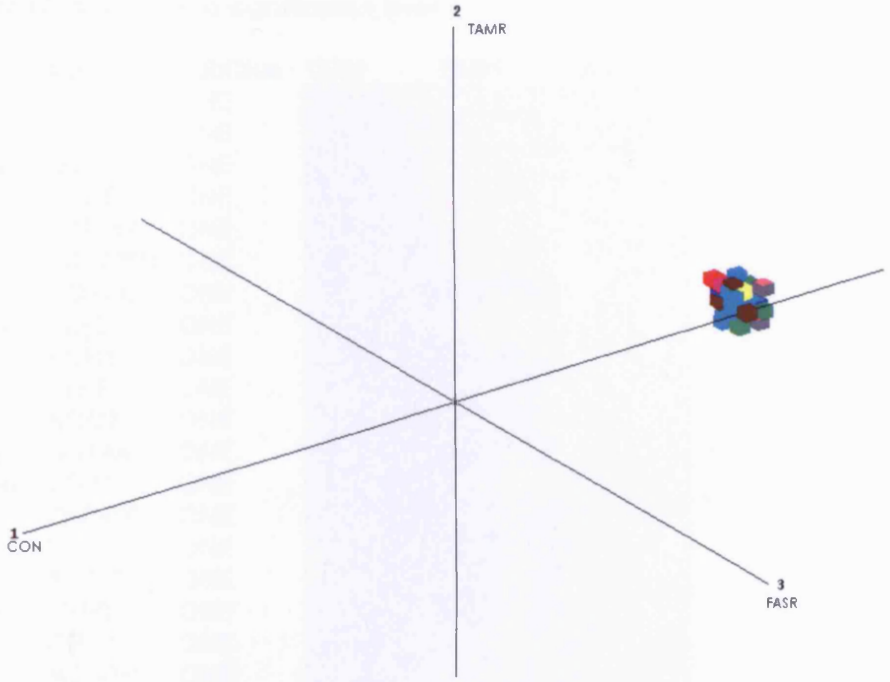


Figure 3.28: Cluster 9 as revealed by HCA. Note the cluster members are tightly packed confirming similar expression and potentially similar functional membership. Based on the profile view in figure 3.28, labeled left to right– cluster 1 = Red, 2 = Green, 3 = Blue, 4 = Purple, 5 = Cyan, 6 = Yellow, 7 = Brown on the PCA diagram. Note: The reader should not confuse this labelling with the colour system shown in the hierarchical cluster tree in figure 3.20.

HCA Cluster 12 - 0.05 p value significance level

AffyID	Acronym	SubClus	CON	TAMR	FASR
219857_at	C10orf81	ONE	0	2.80904	3.461525
201034_at	ADD3	ONE	0	2.934892	4.631029
201753_s_at	ADD3	ONE	0	2.159884	3.423021
213126_at	MED8	ONE	0	3.076077	3.56478
214061_at	WDR67	ONE	0	2.794862	3.800288
215111_s_at	TSC22D1	ONE	0	2.103588	2.398638
201405_s_at	COPS6	ONE	0	1.222057	2.018484
220624_s_at	ELF5	ONE	0	3.431427	4.695682
201063_at	RCN1	ONE	0	1.382022	2.421021
220625_s_at	ELF5	ONE	0	1.875605	3.193231
203814_s_at	NQO2	ONE	0	1.664837	2.060643
202967_at	GSTA4	ONE	0	1.634157	2.118917
202412_s_at	USP1	ONE	0	2.805206	3.881203
222283_at	ZNF480	ONE	0	1.827862	3.059545
201468_s_at	NQO1	ONE	0	1.287435	2.100707
209934_s_at	ATP2C1	ONE	0	1.777429	2.001345
205282_at	LRP8	ONE	0	1.262789	2.032969
206357_at	OPA3	ONE	0	2.06407	3.170337
212949_at	NCAPH	ONE	0	2.562166	2.85814
209276_s_at	GLRX	ONE	0	1.265837	2.15298
203653_s_at	COIL	ONE	0	2.076709	3.114957
206474_at	PCTK2	ONE	0	2.191751	3.096692
213610_s_at	KLHL23	ONE	0	3.154574	3.859274
201762_s_at	PSME2	ONE	0	1.127528	1.811071
200755_s_at	CALU	ONE	0	2.723388	2.978275
201206_s_at	RRBP1	ONE	0	2.245146	2.731236
201116_s_at	CPE	ONE	0	1.289185	2.603686
213172_at	TTC9	ONE	0	1.498063	2.090015
210567_s_at	SKP2	ONE	0	1.884069	3.01866
219806_s_at	C11orf75	TWO	0	2.282696	1.434731
215549_x_at	LOC64385	TWO	0	2.694807	1.488428
217771_at	GOLM1	TWO	0	2.475228	1.383317
210297_s_at	MSMB	THREE	0	4.411368	3.119193
217789_at	SNX6	THREE	0	3.156488	2.53183
209135_at	ASPH	THREE	0	3.1738	2.608906
215136_s_at	EXOSC8	THREE	0	1.563496	1.474199
211506_s_at	IL8	THREE	0	3.103046	2.219327
205063_at	SIP1	THREE	0	2.66043	2.143703
205047_s_at	ASNS	FOUR	0	2.633072	2.843374
205048_s_at	PSPH	FOUR	0	2.29034	2.092785
201963_at	ACSL1	FOUR	0	2.061295	1.734714
221609_s_at	WNT6	FIVE	0	3.55638	3.117833
214159_at	PLCE1	FIVE	0	2.396061	2.335652
219049_at	ChGn	SIX	0	3.519977	4.091576
218158_s_at	APPL1	SIX	0	2.45638	2.486399
219161_s_at	CKLF	SIX	0	1.718686	1.400774
209954_x_at	SS18	SIX	0	1.61735	1.637341

Figure 3.29 – Cluster 12 – Significant up regulated probes in both TAMR and FASR at the 0.05 level vs MCF7 control. Light red indicates a high degree of up regulation.

HCA Cluster 12 - 0.005 p value significance level

AffyID	Acroynm	SubClus	CON	TAMR	FASR
201063_at	RCN1	ONE	0	1.382022	2.421021
209276_s_at	GLRX	ONE	0	1.265837	2.15298
201116_s_at	CPE	ONE	0	1.289185	2.603686
220624_s_at	ELF5	TWO	0	3.431427	4.695682
220625_s_at	ELF5	TWO	0	1.875605	3.193231
205282_at	LRP8	TWO	0	1.262789	2.032969
222283_at	ZNF480	THREE	0	1.827862	3.059545
206357_at	OPA3	THREE	0	2.06407	3.170337
203653_s_at	COIL	THREE	0	2.076709	3.114957
202412_s_at	USP1	FOUR	0	2.805206	3.881203
213610_s_at	KLHL23	FOUR	0	3.154574	3.859274
219857_at	C10orf81	FIVE	0	2.80904	3.461525
215111_s_at	TSC22D1	FIVE	0	2.103588	2.398638
201405_s_at	COPS6	FIVE	0	1.222057	2.018484
203814_s_at	NQO2	FIVE	0	1.664837	2.060643
202967_at	GSTA4	FIVE	0	1.634157	2.118917
201468_s_at	NQO1	FIVE	0	1.287435	2.100707
201762_s_at	PSME2	FIVE	0	1.127528	1.811071
200755_s_at	CALU	FIVE	0	2.723388	2.978275
217789_at	SNX6	SIX	0	3.156488	2.53183
209135_at	ASPH	SIX	0	3.1738	2.608906
205047_s_at	ASNS	SEVEN	0	2.633072	2.843374
205048_s_at	PSPH	SEVEN	0	2.29034	2.092785
201963_at	ACSL1	SEVEN	0	2.061295	1.734714
219049_at	ChGn	EIGHT	0	3.519977	4.091576
218158_s_at	APPL1	EIGHT	0	2.45638	2.486399
219161_s_at	CKLF	EIGHT	0	1.718686	1.400774
209954_x_at	SS18	EIGHT	0	1.61735	1.637341

Figure 3.30 – Cluster 12 – Significant up regulated probes in both TAMR and FASR at the 0.005 level vs MCF7 control. Light red indicates a high level of up regulation.

Figure 3.29 and figure 3.30 corresponds to only the most significant probes as donated in red boxes at the 0.05 significance level and 0.005 probe set respectively. Both results were generated independently of each other. Cluster 12 contains potential targets of interest unregulated in both forms of resistance as revealed later using ontological searching.

Again, to aid understanding of the relationship between each probe within each cluster, the result can be plotted in 3D through I-10 to reveal genes of interest within each membership group as seen in figure 3.31.

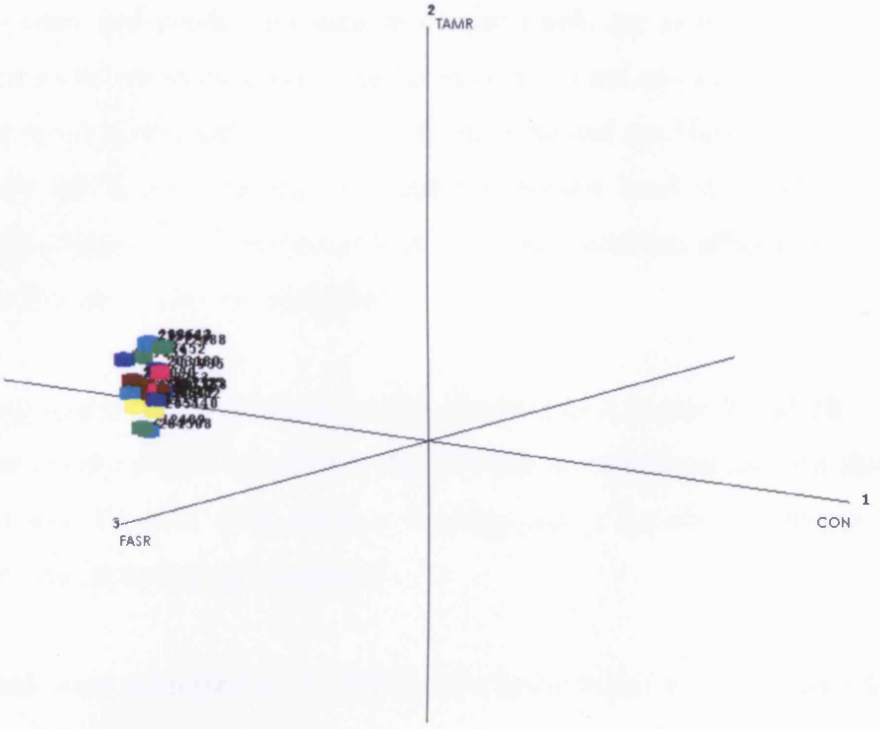


Figure 3.31: Cluster 12 - The cluster members are tightly packed together confirming similar functional membership. Based on the profile view from figure 3.30, as labeled left to right– Cluster 1 = Red (behind main cluster group), 2 = Green, 3 = Blue, 4 = Purple, 5 = Cyan, 6 = Yellow, 7 = Brown and 8 = Grey (behind main cluster group) on the PCA diagram. Note: Again, the colour scheme is not to be confused with the colour system shown in the Hierarchical cluster tree in Figure 5.18.

3.9 Exploration to Reveal Potential Signalling Targets in Resistance

FATIGO ("Babelomics") software, accessible in I-10, uses a multi tiered level approach to annotation using embedded GO ontology. Gene Ontology (GO) is, probably, the most successful among the current initiatives for the standardisation of the nomenclature of biological processes. It is divided into molecular, functional and sub cellular location categories. Also information regarding upstream Transcription Factor regulators is available through FATIGO+. GO represents the biological knowledge as a tree. Upper levels in the tree represent more general concepts and as the tree is traversed towards deeper levels, the definitions are more and more precise (e.g. cell cycle > regulation of cell cycle > positive regulation of cell cycle). Since genes are annotated at different levels it is common to use the inclusive analysis instead of using

directly the annotation of the genes at the deepest level possible. In the inclusive analysis a level of abstraction is chosen and genes annotated at deeper levels are assigned to this level. The efficiency of the test increases because there are fewer terms to test and more genes per term, but the selection of the level is arbitrary. FATIGO has implemented the Nested Inclusive Analysis (NIA), in which the test is done recursively until the deepest level in which significance is obtained and only this last level is reported. In this way both variables: efficiency of the test and highest precision in the term found are optimised.

This system was applied to the significant probes revealed of Clusters 9 and 10 – clusters in which all probes are either induced in both TAMR or FASR or suppressed and therefore potential regulators of blockable TAMR/FASR biology and therefore a possible to reason as to why resistance develops toward both drug treatments.

A total of 30 probes were submitted to FATIGO for Cluster 9 and 47 for Cluster 12. At each level, probes shown from the literature to have shown functional significance or play a role in key pathways are highlighted in bold and shaded in red within the orange tables over the following pages. Depending on the level chosen, the Affymetrix probes might fall into molecular functions, particularly at the higher levels (e.g level 3).

3.9.1 – Cluster 9 – Tamoxifen resistance and Faslodex resistance suppressed genes

Available ontological tables range from level 3 through to 9 detailing the ontology for each level. They report information regarding the gene probe ID, the number of significant genes in each cluster and the percentage comprising the whole analysed gene population. For cluster 9, there was no information available at levels 1 and 2 with no significant classes generated after level 9. Some genes also have overlapping function between the levels.

Gene Ontology : molecular function. Level:3	Genes	Nº genes	Percentage
protein binding	207935_s_at 208613_s_at 207836_s_at 205440_s_at 207302_at 209016_s_at 205792_at 207452_s_at 219529_at	9	60
ion binding	201368_at 204508_s_at 220414_at 219529_at	4	26.67
nucleic acid binding	201368_at 207836_s_at 208763_s_at	3	20
neurotransmitter binding	205440_s_at	1	6.67
lyase activity	204508_s_at	1	6.67
peptide binding	205440_s_at	1	6.67
ion transmembrane transporter activity	219529_at	1	6.67
transferase activity	204542_at	1	6.67
nucleotide binding	207836_s_at	1	6.67
structural constituent of cytoskeleton	207935_s_at	1	6.67
channel activity	219529_at	1	6.67
lipid binding	207452_s_at	1	6.67
oxidoreductase activity	203180_at	1	6.67
receptor activity	**205440_s_at**	**1**	**6.67**

Table 3.6: Summary of genes revealed in cluster 9 and their associated molecular function. The majority of the genes are responsible for protein, ion and nucleic acid binding.

At level three, it can be observed that the genes encoded by the probes are involved largely in protein, ion and nucleic acid binding processes as shown in table 3.6 and Figure 3.33.

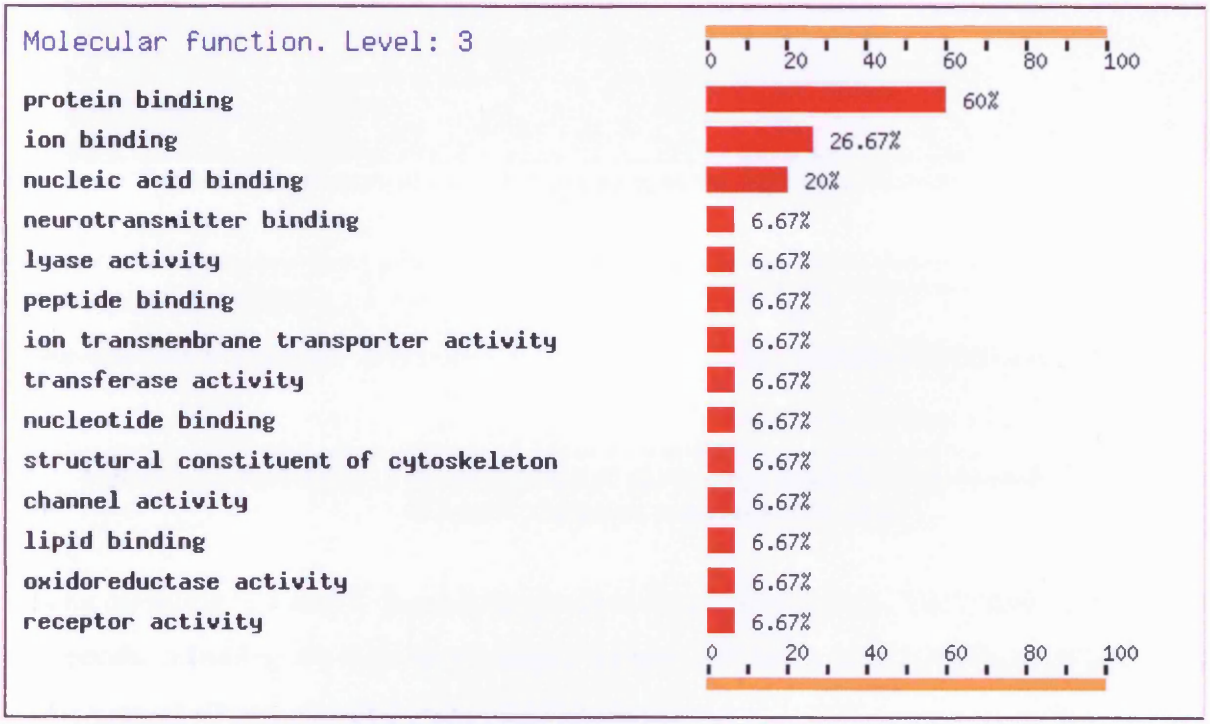


Figure 3.33: Summary of molecular function of cluster 9 at level 3. Most of the probes in the cluster have a role in protein, ion and nucleic acid binding.

Gene Ontology : molecular function. Level:7	Genes	Nº genes	Percentage
peptide receptor activity, G-protein coupled	205440_s_at	1	50
chloride channel activity	219529_at	1	50

Table 3.7: Summary of molecular function of the genes from cluster 9.

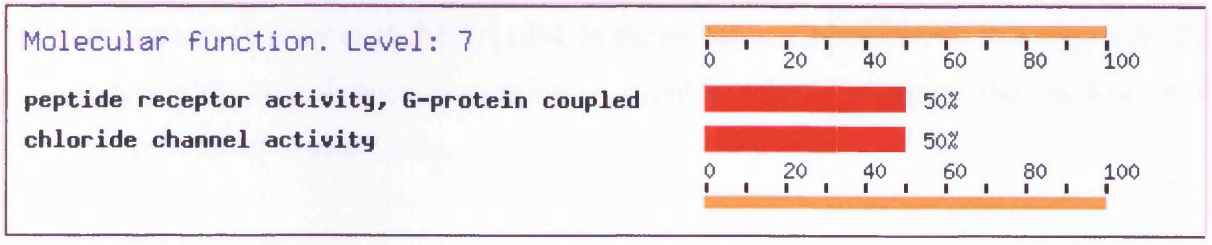


Figure 3.34: Summary of the percentage of genes responsible for a given molecular function. Only two remain at this level.

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
neuropeptide Y receptor activity	205440_s_at	1	100

Table 3.8: Summary of probes remaining at the highest molecular ontological level.

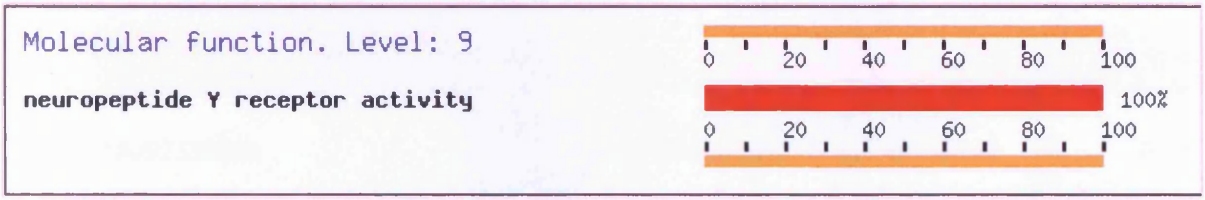


Figure 3.35: Summary of the percentage of genes responsible for a given molecular function at level 9. Only one remains at this level.

Focusing on levels 3, 7 and 9 showed the greatest change of ontology. The higher levels became more specific revealing the G-protein coupled receptor 205440_s_at to have the most ontological classification of all the uploaded probes for this cluster.

As probe 205440_s_at persisted in analysis through many levels of GO ontology focusing on receptors, was taken forward into DAVID to determine more information. The probe 205440_s_at, also known as the *NPY Y(1)* receptor, induces the expression of *CRE* containing target genes through the *CaM kinase-CREB* pathway, and inhibits *CRE* containing genes when cellular cAMP levels are elevated (Sheriff et al, 2002) [108]. Recently, a role of *neuropeptide Y (NPY)* in tumor biology was suggested based on the high density of *NPY* receptors in breast and ovarian cancers. These *NPY* receptors are a potential new molecular target for the therapy of malignant tumours (Körner et al, 2004) [109]. In the models used however in this project *NPY (1)* receptor decreases in resistance suggesting a possible role in response that is lost in the aggressive, proliferate resistant state.

Oncomine analysis, available as a link within I-10, revealed that *NPY* receptor in resistance is at higher expression level in luminal phenotypes, higher in *ER+* and *PgR+* and *HER2* –ve breast cancer. This would equate with a relationship with indolent, endocrine response phenotypes as seen in the present model system study and the receptor would be worthy of further exploration both as a biomarker (loss in resistance) and as a potential target in the responsive state.

3.9.2 – Cluster 12 – Ontology of Tamoxifen resistance and Faslodex resistance induced probes

This cluster revealed GO ontological information only between levels 4 through 8. No information was returned for levels 1 through 3 or past level 8. Only levels 4 and 7 are displayed.

Gene Ontology : molecular function. Level:4	Genes	Nº genes	Percentage
metal ion binding	205282_at 203814_s_at 219049_at 201063_at 201116_s_at 201963_at 205048_s_at 209135_at 222283_at	9	52.94
cation binding	205282_at 203814_s_at 201063_at 201116_s_at 209135_at 222283_at	6	35.29
electron carrier activity	203814_s_at 209276_s_at 201468_s_at 209135_at	4	23.53
DNA binding	215111_s_at 220624_s_at	2	11.76
hydrolase activity, acting on ester bonds	205048_s_at 202412_s_at	2	11.76
peptidase activity	201116_s_at 202412_s_at	2	11.76
oxidoreductase activity, acting on NADH or NADPH	203814_s_at 201468_s_at	2	11.76
phospholipid binding	217789_at	1	5.88
transmembrane receptor activity	**205282_at**	**1**	**5.88**
ligase activity, forming carbon-nitrogen bonds	205047_s_at	1	5.88
apolipoprotein receptor activity	**205282_at**	**1**	**5.88**
ligase activity, forming carbon-sulfur bonds	201963_at	1	5.88
lipoprotein binding	205282_at	1	5.88
disulfide oxidoreductase activity	209276_s_at	1	5.88
protein dimerization activity	217789_at	1	5.88
identical protein binding	217789_at	1	5.88
dioxygenase activity	209135_at	1	5.88
oxidoreductase activity, acting on single donors with incorporation of molecular oxygen	209135_at	1	5.88
receptor binding	**219161_s_at**	**1**	**5.88**
oxidoreductase activity, acting on paired donors, with incorporation or reduction of molecular oxygen	209135_at	1	5.88
transferase activity, transferring glycosyl groups	219049_at	1	5.88

Table 3.9: Summary of probes level 4 molecular ontological level for cluster 12

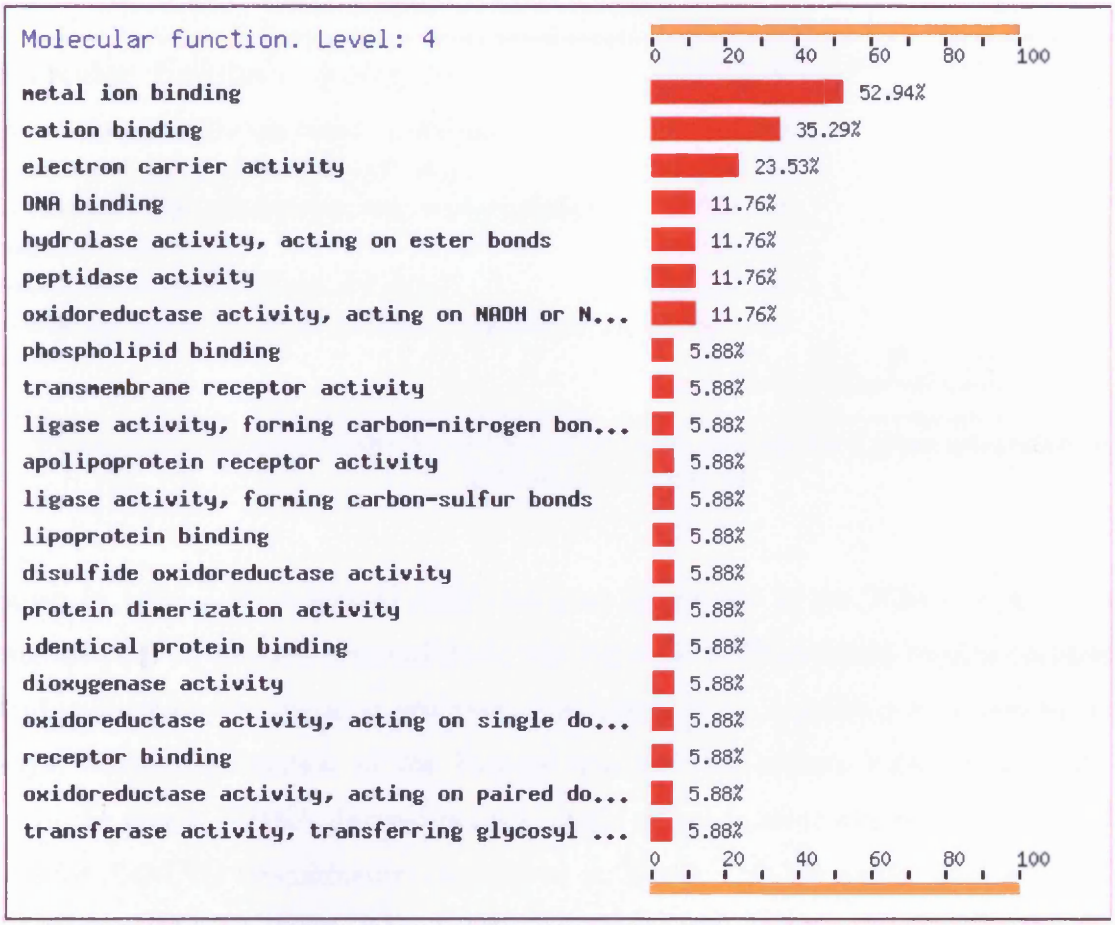


Figure 3.36: Summary of the percentage of genes responsible for a given molecular function at level 4 for cluster 12.

Gene Ontology : molecular function. Level:7	Genes	Nº genes	Percentage
ubiquitin-specific protease activity	202412_s_at	1	20
phosphoserine phosphatase activity	205048_s_at	1	20
glucuronosyl-N-acetylgalactosaminyl-proteoglycan 4-beta-N-acetylgalactosaminyltransferase activity	219049_at	1	20
chemokine activity	**219161_s_at**	**1**	**20**
metallocarboxypeptidase activity	201116_s_at	1	20
glucuronylgalactosylproteoglycan 4-beta-N-acetylgalactosaminyltransferase activity	219049_at	1	20

Table 3.10: Summary of probes at molecular ontological level 7 for cluster 12.

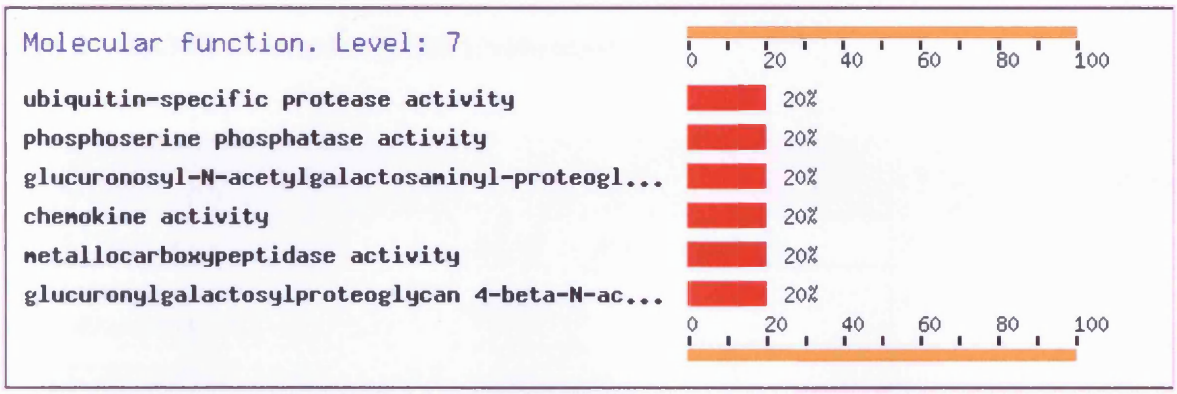


Figure 3.37: Summary of the percentage of genes responsible for a given molecular function at level 4 for cluster 12.

As shown in table 3.10, a highly significant gene of interest is the 202412_s_at – ubiquitin thiolesterase (*USP1*) which is responsible for playing a role in DNA repair. Protein ubiquitination and deubiquitination are dynamic processes implicated in the regulation of numerous cellular pathways. Monoubiquitination of the Fanconi anemia (FA) protein *FANCD2* appears to be critical in the repair of DNA damage because many of the proteins that are mutated in FA are required for *FANCD2* ubiquitination (Nijman et al, 2005) [110].The study proposes that *USP1* deubiquitinates *FANCD2* when cells exit S phase or recommence cycling after a DNA damage insult and may play a critical role in the FA pathway by recycling *FANCD2*. An oncomine search shows its presence in high grade *ER*- breast cancer and increases of enzymes that regulate DNA repair than may be of benefit in endocrine resistance.

3.9.3 Summary of remaining Cluster ontology taken to most significant level

Cluster 1 – TAMR Suppressed – FASR Induced

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
ATPase activity, coupled	209380_s_at 218986_s_at 207521_s_at	3	50
fibroblast growth factor receptor activity	**203638_s_at**	**1**	**16.67**
protein phosphatase type 1 activity	204284_at	1	16.67
lysine N-acetyltransferase activity	200898_s_at	1	16.67

Table 3.11: Summary of probes at molecular ontological level 9 for cluster 1.

Cluster 2 – TAMR Induced – FASR Unchanged

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
ATPase activity, coupled	209641_s_at 201242_s_at	2	28.57
nicotinic acetylcholine-activated cation-selective channel activity	221107_at	1	14.29
transmembrane receptor protein tyrosine phosphatase activity	205846_at	1	14.29
delayed rectifier potassium channel activity	205262_at	1	14.29
MAP kinase tyrosine/serine/threonine phosphatase activity	208891_at	1	14.29
pancreatic ribonuclease activity	205141_at	1	14.29

Table 3.12: Summary of probes at molecular ontological level 9 for cluster 2.

Cluster 3 – TAMR unchanged – FASR Induced

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
ATPase activity, coupled	209735_at 212136_at 204567_s_at 213036_x_at 204873_at	5	33.33
prostaglandin receptor activity	210636_at 204897_at	2	13.33
phosphoinositide phospholipase C activity	203895_at	1	6.67
glycine C-acetyltransferase activity	205164_at	1	6.67
myo-inositol:sodium symporter activity	212944_at	1	6.67
inositol-polyphosphate 5-phosphatase activity	212990_at	1	6.67
lysine N-acetyltransferase activity	209106_at	1	6.67
hepatocyte growth factor receptor activity	**203510_at**	**1**	**6.67**
ribonuclease P activity	209482_at	1	6.67
3',5'-cyclic-AMP phosphodiesterase activity	203708_at	1	6.67

Table 3.13: Summary of probes at molecular ontological level 9 for cluster 3.

Cluster 4 – TAMR Suppressed – FASR Unchanged

Gene Ontology : molecular function. Level:9	Genes	N° genes	Percentage
inositol 1,4,5-triphosphate-sensitive calcium-release channel activity	203710_at	1	100

Table 3.14: Summary of probes at molecular ontological level 9 for cluster 4.

Cluster 5 – TAMR Unchanged – FASR Suppressed

Gene Ontology : molecular function. Level:9	Genes	N° genes	Percentage
potassium:chloride symporter activity	218066_at	1	33.33
platelet-derived growth factor receptor activity	205226_at	1	33.33
MAP kinase tyrosine/serine/threonine phosphatase activity	204015_s_at	1	33.33

Table 3.15: Summary of probes at molecular ontological level 9 for cluster 5.

Cluster 6 – TAMR Unchanged – FASR Suppressed

Gene Ontology : molecular function. Level:9	Genes	N° genes	Percentage
diacylglycerol O-acyltransferase activity	203669_s_at	1	20
2-acylglycerol O-acyltransferase activity	203669_s_at	1	20
ATPase activity, coupled	208161_s_at	1	20
solute:hydrogen antiporter activity	204981_at	1	20
MAP kinase tyrosine/serine/threonine phosphatase activity	208892_s_at	1	20
low voltage-gated calcium channel activity	205845_at	1	20

Table 3.16: Summary of probes at molecular ontological level 9 for cluster 6.

Cluster 7 – TAMR Suppressed – FASR Suppressed

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
MAP kinase activity	210059_s_at	1	50
ephrin receptor activity	206114_at	1	50

Table 3.17: Summary of probes at molecular ontological level 9 for cluster 7.

Cluster 8 – TAMR Unchanged – FASR Induced

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
ATPase activity, coupled	208795_s_at 201241_at 205023_at	3	42.86
sodium:hydrogen antiporter activity	203909_at	1	14.29
fibroblast growth factor receptor activity	**211237_s_at**	**1**	**14.29**
3',5'-cyclic-GMP phosphodiesterase activity	205593_s_at	1	14.29
non-membrane spanning protein tyrosine phosphatase activity	201629_s_at	1	14.29
solute:hydrogen antiporter activity	203909_at	1	14.29

Table 3.18: Summary of probes at molecular ontological level 9 for cluster 8.

Cluster 10 – TAMR Suppressed – FASR Suppressed

Gene Ontology : molecular function. Level:9	Genes	Nº genes	Percentage
ATP-gated cation channel activity	221372_s_at 215464_s_at	2	40
ephrin receptor activity	203499_at	1	20
insulin-like growth factor receptor activity	**203627_at**	**1**	**20**
transmembrane receptor protein tyrosine phosphatase activity	203038_at	1	20
epidermal growth factor receptor activity	203627_at	1	20

Table 3.19: Summary of probes at molecular ontological level 9 for cluster 10.

Cluster 11 – TAMR Induced – FASR unchanged

Gene Ontology : molecular function. Level:6	Genes	Nº genes	Percentage
long-chain-fatty-acid-CoA ligase activity	205768_s_at	1	25
sodium ion binding	201243_s_at	1	25
pyrophosphatase activity	201243_s_at	1	25
Rho GDP-dissociation inhibitor activity	201288_at	1	25
potassium ion binding	201243_s_at	1	25
zinc ion binding	208510_s_at	1	25

Table 3.20: Summary of probes at molecular ontological level 9 for cluster 11.

Note: Although similar functionality of certain probes appears in more than one cluster, for example ATPase activity coupled appears in more than one cluster, there was multiple coverage of probes representing genes with a particular clustering association. This was sufficient to place the gene into its own cluster. The red shaded areas previously highlighted across the different levels verifies, particularly in the FASR induced clusters (Cluster 1, 3 and 8) that genes such as *FGR4* or *Met* are suppressed in both forms of resistance (*IGFR* – cluster 10), which reassures us of the changes in the clusters may be equally robust as biomarkers/potential targets of individual/shared resistant states.

3.10 Comparison of Cluster 9 Suppressed vs Cluster 12 Induced results of TAMR and FASR

FATIGO+ has the ability of comparing two gene lists. This is particularly interesting for comparing two different lists showing differentiation in opposite directions. The system compares each list against the other against GO ontology as shown previously for individual clusters.

No ontological information was present at levels 1 and 2.

At level 3, Cluster 9 suppressed events have more genes associated with cell developmental processes as opposed to cellular metabolism sharing a greater percentage in Cluster 12 which corresponds to induced events.

3.11 – Conclusion

The MCF7 models have not lost or regained a particular phenotype – most notably they have not shifted to a basal phenotype in resistance.

Hierarchical clustering, PAM and Self organising maps perform well for the MCF7 Control vs TAMR vs FASR comparison. The priority of each probe within a HCA cluster can be assessed using p-values of each dendogram branch. Using the pvClust package of I-10, the system allocates priority using boot strapping of each cluster revealed using clValid for Hierarchical clustering. Assessment at the 0.05 and 0.005 significance p-value level was examined which reveals individual candidates within each system. The individual cluster membership of each clustering technique was compared and cluster results aligned. The values for K-Means and PAM were very similar however PAM performed marginally better. The K-Means results are therefore omitted. This assumption was confirmed in the analysis by the clValid method. Although grouping the data into six clusters performs well statistically – confirmed by a high affinity value from the PAM silhouette plot values, 12 clusters reveals all changes within the data encompassing certain profiles hidden within each of the smaller six clusters revealed in preliminary analysis. Across the different clustering methods, significant probes of interest are revealed within clusters of each method.

Genes which were induced in both TAMR and FASR included ubiquitin thiolesterase which is responsible for playing a role in DNA repair and carboxypeptidase A and E activity has been revealed as a biomarker in pulmonary neuroendocrine tumors.

Genes which were revealed as very significantly suppressed in both TAMR and FASR included the G-protein coupled *NPY Y(1) receptor*. This gene induces the expression of CRE containing target genes through the *CaM kinase-CREB* pathway with a role of *neuropeptide Y* (*NPY*) in

tumor biology suggested based on the high density of *NPY* receptors in breast and ovarian cancers (Körner et al, 2004) [109]. Ontology for all clusters revealed is also presented.

New libraries reinforce the ever developing nature of 'R'. Due to the modular nature of I-10, the application can be continually updated. The introduction of the pvClust methods were implemented during the later stages of research. However they are very powerful in the way they categorically assess how many cluster are present and the best methods to choose.

Annotation tools are also evolving. The introduction of FATIGO, alongside more traditional tools such as DAVID help the biologist make more decisive decisions in terms of which genes are the most interesting to further study which could potentially form future therapeutic targets.

Chapter 4

'Superstes'

Development of a clinical cancer survival query tool

Chapter 4 – Superstes – Development of a clinical cancer survival query tool

4.1 Background

4.1.1 Prognostic indices in cancer

Current prognostic indices outlined in Chapter 1, Section 1.1, which includes the St. Gallen criteria, the TNM system and the Nottingham Prognostic Index (NPI) can integrate information from several validated prognostic variables and assign patients to different prognostic categories. However, these prognostic models do not provide estimates for survival probability (Blamey et al, 2007) [111].

Studies have shown that the breast cancer mortality reduction caused by mammography screening is estimated to amount to 28–30% in 2007 based on studies conducted in the Netherlands (Blamey et al, 2007) [111]. However it has been suggested that patients examined in the early 1990's would have benefited more from adjuvant therapy given than mammography screening. This is thought to have played a role in women who died at the age range of 45–54 who would not have participated in the screening programme, with the resulting mortality reduction thought largely to be attributed to adjuvant treatment in this age group more than the introduction of mammography screening (Blamey et al, 2007) [111]. In a recent study by Blamey *et al*, it was also noted that improvement in patient survival times could possibly be explained by more accurate lymph node staging in combination with more breast cancer patients being detected due through mammography screening since the NPI was developed (Blamey et al, 2007) [111].

Ioannidis *et al* in a recent review stated that genomic risk assessment could potentially outperform clinical-pathological risk assessment (Ioannidis et al, 2002) [112]. However, more evidence needs to be generated than that which currently exists if genomic information is to be accepted as being more accurate in the prediction of survival. The situation where clinical-pathological variables are replaced entirely with gene-level data still remains far into the future (Ioannidis et al, 2002) [112]. Currently, accurate prognostic models depend on the combination

of covariates explaining molecular information such as Er and PgR status, the extent of disease and tumour morphology.

Consequently, the influence different clinical and pathological attributes impact on survival of cancer remains of paramount interest. It is of great benefit to both patients and oncologists to determine whether survival is influenced by one or more variables collectively. These variables can vary from being categorical, such as the type of treatment a patient received, or continuous variables, such as the patient's age (Blamey et al, 2007) [111]. Ultimately it is hoped that potentially new combinations of variables will be discovered to evaluate whether established prognostic models, such as the NPI, can be further improved which itself is based on the variables of tumour size, grade and node status. This goal of determining new prognostic models may be achievable through development and application of bioinformatic analysis tools to large clinical breast cancer databases with associated follow-up data and diverse clinico pathological variables.

4.1.2 The SEER dataset

To discover new prognostic models based on combinations of different patient variables requires a dataset of cancer patients. The National Cancer Institute's cancer database Surveillance Epidemiology and End Results Program (SEER) in the United States is one such patient dataset and contains over 400,000 cases of cancer derived from population-based cancer registries between 1972 and 2002 (Ries et al, 2005) [100]. It contains detailed patient clinical, pathological attributes which range from their age through to any treatment received such as surgery. Although not directly available from the institute's web site, the data source is available on an academic license basis for research and development purposes.

4.1.3 Prognostic tools developed using SEER data

There are many data analysis tools using SEER data patient information. A well documented example is the survival query tool called Adjuvant! Online launched in 2001 (Hess et al, 2008) [113]. The tool assesses the benefits and risks of adjuvant therapy for patients with early onset breast cancer (Hess et al, 2008) [113]. This online web-based application has been targeted at

oncologists working together with patients to estimate risk of a negative outcome (i.e. relapse or death related to a cancer) of a particular patient. A decision whether to proceed with a particular management strategy can be taken after assessing the risk to patient quality of life versus outcome of the disease using the tool. A treatment decision can then be decided between the oncologist and patient. However, Adjuvant! Online is not without its critics. A study by the Harvard Medical School highlighted a key shortcoming. The study stated that empirical models are only as precise as the dataset they are based upon (Chen et al, 2008) [114]. If a patient is defined by lots of different variables describing information about the patient – Er status, tumour size, age among other variables, narrowing the search parameters will reduce the dataset upon which a prediction of survival will be made. The group suggested that a possible undesired consequence of this approach is considerable statistical uncertainty. The Adjuvant!Online tool attempts to avoid this issue by grouping patients into large 'bins'. While large bins decrease the uncertainties associated with estimation and thus increase their statistical validity, they are not able to finely stratify patients. For example, patients perceived to have the highest risk as estimated in Adjuvant!Online (derived from a large cohort of individuals where the selected covariate criteria resemble a particular patient in every respect) may actually have divergent prognoses (Chen et al, 2008) [114] so results predicted by Adjuvant!Online can be often difficult to recreate. Consequently, the Harvard Medical School released a query tool to address these issues - CancerMath.net – developed with the Institute of Quantitative Medicine at Massachusetts General Hospital in Boston. CancerMath.net offers small calculators for breast, melanoma and renal cell carcinoma to produce 15 year survival curves. However, CancerMath.net is still in a trial phase and can only base its predictions on a limited subset of covariates which are entered by the patient.

Clearly, although some initiatives are being applied to the SEER dataset, there remains a need to develop a new tool which allows the user to explore survival patterns using many combinations of patient variables. Furthermore, rationalising particular patient variables with statistical methods which assess the significance such combinations have on patient survival could have wide ranging clinical benefits for future treatment strategies.

4.2 Analysis of Survival Data

One of the first approaches for modeling survival was the life table, developed by *Berkson and Gage* in 1950. A life table contains a range of survival times for all patients divided into sub intervals. For intervals representing 'alive', 'died' or 'censored', a value is calculated based on the number and proportion of cases. A patient would be classed as censored if they left the study or remain alive when the study completes. Manipulation of each quantity allows parameters of interest such as prognostic variables, which therefore link to survival, to be estimated. The main problems that occur with this approach are that they do not assess the impact certain categorical variables have on survival times (Chanrion et al, 2007) [82]. This ultimately led to regression methods being applied to survival.

4.2.1 Parametric and non parametric statistics

To assist in the discovery process, different statistical methods are employed to analyse the contribution a particular covariate is having in terms of affecting survival outcome. There are different types of statistics which facilitate this discovery.

Parametric statistics are those where the population is assumed to fit any parameterized distributions (most typically a normal distribution). Parametric statistical methods are mathematical procedures for statistical hypothesis testing which assume that the distributions of the variables being assessed belong to known parameterized families of probability distributions (Hartigan et al, 1979) [44]. A typical example is a t-test.

In contrast, non-parametric statistics are widely used for studying populations that take on a ranked order, for example measurements of the effectiveness of a drug on a scale of 1 to 10. The use of non-parametric methods is often necessary when data has a ranking but no clear numerical interpretation (Hartigan et al, 1979) [44]. As non-parametric methods make fewer assumptions, their applicability is much wider than the corresponding parametric methods. In particular, they may be applied in situations where less is known about the distribution in question. Also, due to the reliance on fewer prior assumptions, non-parametric methods are more robust.

In certain instances, possibly even when the use of parametric methods could be justified, non-parametric methods are often easier to use. Due both to this simplicity and due to their greater robustness, non-parametric methods are seen by some statisticians as leaving less room for improper use and misunderstanding (Hartigan et al, 1979) [44].

Non-parametric models differ from parametric models in that the model structure is not specified a priori but is instead determined from the data (Hartigan et al, 1979) [44]. A histogram is a simple nonparametric estimate of a probability distribution such as that of age or marital status of patients. Typical examples of non-parametric models include Kaplan-Meier survival curves, the log rank test and the Cox proportional hazards model. Each of these tests could be applied to different cohorts of patients based on patient variable information extracted from the SEER dataset to assess the impact particular patient variable combinations have on patient survival.

4.2.2 Kaplan Meier survival curves

A standard linear regression model has shortfalls in that survival times are not usually normally distributed and "censored" data occur frequently (Chen et al, 2008) [115]. This was shown by Kaplan-Meier and was considered to be a breakthrough in survival analysis. Kaplan-Meier first demonstrated the construction of a survival curve using a non parametric method.

In clinical trials, it can take years to find appropriate patients for a trial to test a particular drug. The term 'follow up' is often used to indicate examination periods of patients to record different amounts of information relevant to the clinical trial. Studies which 'follow up' past patients, have the common problem that patients will be 'followed up' at different points in their treatment, often over a period of many years. This leads to the issue of patient survival tracking where patients would have started the clinical trial at different points in time. However results of a clinical trial are analysed at a single point in time and so at that time, some patients would have received varying lengths of 'follow up' of their treatment during the study.

This is illustrated clearly with a theoretical example such as a clinical trial. A clinical trial for drug X, recruit patients who embark on their treatment in the first year of the commencement of

the trial. Analysis of the data collected during the trail takes place after five years. A patient who survives till the end of the trial, yet joined the trial at the start would have five years of progress information. A patient who survives until the end of the trial yet joined the trial much later – such as after three years of the trial starting – will only have two years of information recorded. The problem of a patient having varying amounts of 'follow up' during the trial only becomes an issue if the patient remains alive at the end of the trial period. However it is also not desirable to remove patients who did not start the trial at the beginning as the information gathered during the time the patient participated in the trial could still be valuable to the overall study. Consequently there needs to be a way in which this patient can be statistically removed from the resulting survival curve at their respective time point. The patient, if only in the trial for two years cannot be classified as having died yet also cannot be classified as having survived (Chen et al, 2008) [115]. However, if the patient dies, it is important this is recorded and is indicated by a 'step down' on the survival curve. A survival curve illustrating this information can be observed in figure 4.1

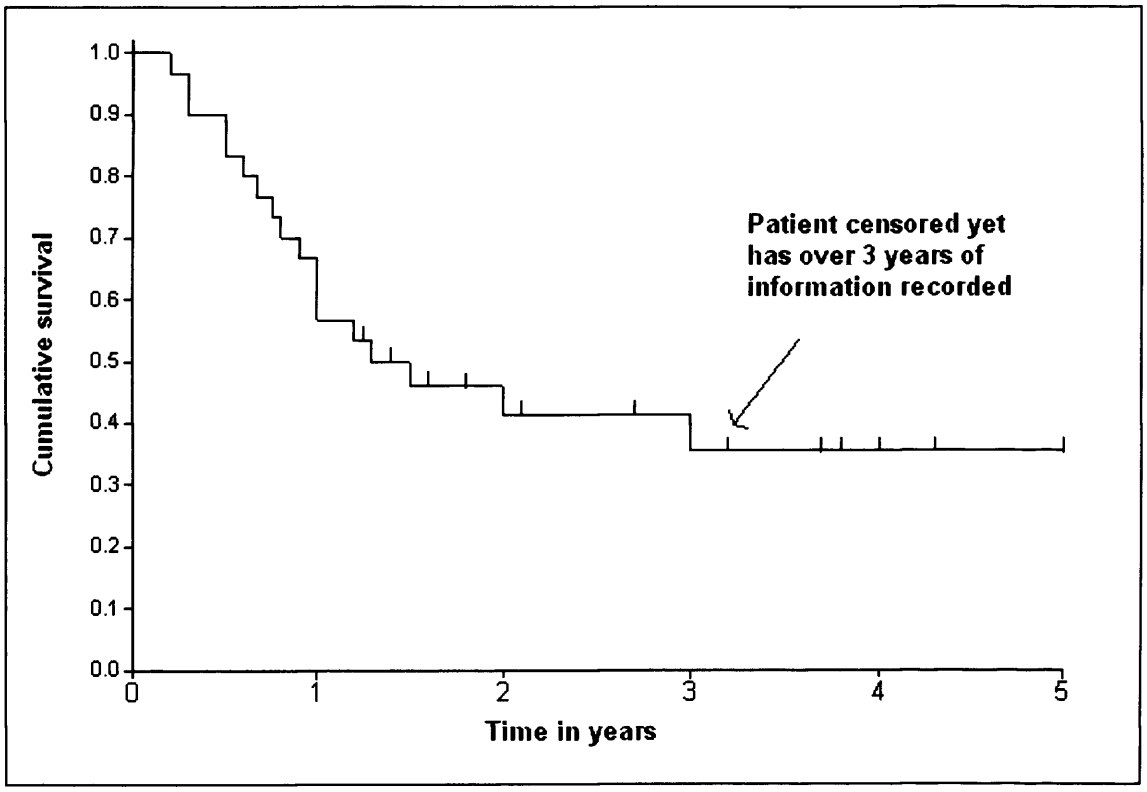


Figure 4.1 Theoretical Kaplan-Meier plot for a clinical trial showing the effect of when a patient dies (producing a step down in the curve) and how censoring a patient is indicated (vertical lines or 'tick marks').

Removing a patient at a particular point in time when their period of study ends is termed as censoring. This has the ultimate goal of producing the most accurate survival curve as possible given a particular dataset. As shown in figure 4.1, vertical tick marks indicate on the survival curve where a patient has been censored. It must be remembered that as the censoring process has an impact on the total number of patients contributing to the survival curve, the remaining patients who die after that point will contribute as a higher proportion of the remaining population.

The resulting effect on the survival curve is a series of steps which will be slightly larger than if the censored patient remained. Censoring a patient reduces the sample size of patients at risk after the time of censorship. Reducing sample size always reduces reliability, so the more patients are censored and the earlier they are censored the more unreliable the curve will become. As a result of each censored patient reducing the reliability of the curve from that point forward, the end of the curve is most affected. This could be viewed as being unfortunate, since the end of the curve represents long term survival which is the ultimate goal (Chen et al, 2008) [115].

4.2.3 The log rank test

The log rank test, first proposed by Nathan Mantel, (also known as the Mantel-Cox test) is a hypothesis test to compare the survival distributions of two samples (Bland et al, 2004) [164]. For example, two groups of patients who have different cancer attributes. It is a nonparametric test and appropriate to use when the data has been censored, such as in the case of a clinical trial as previously outlined.

The log rank statistic can be derived as the score test for the Cox proportional hazards model comparing two groups. It is therefore asymptotically equivalent to the likelihood ratio test statistic based from that model (Bland et al, 2004) [164].

In an analysis of comparison, it might be thought to be feasible to plot survival curves for each group of patients and compare the proportions surviving at any specific time. However, the downfall of this approach is that it does not provide a comparison of the total survival experience

of the two groups of patients, but rather gives a comparison at some arbitrary time point (Bland et al, 2004) [164]. The log rank test is the most popular method for comparing survival between groups and takes the whole review period of a patient into account (commonly known as 'follow up'). It has the considerable advantage that it does not require the user to know anything regarding the shape of the survival curve or the distribution of survival times (Bland et al, 2004) [164]. This is therefore a very useful test to measure the significance of patient variables between two cohorts of patients, such as those generated from a dataset such as SEER.

If a survival time is censored, the individual is considered to be at risk of dying in the week of the censoring but not in subsequent weeks (Bland et al, 2004) [164]. This way of handling censored observations is the same as for the Kaplan-Meier survival curve as previously outlined.

The log rank test is based on the same assumptions as the Kaplan Meier survival curve — censoring is unrelated to prognosis, the survival probabilities are the same for patients recruited early and late in the study, and the events happened at the times specified (Bland et al, 2004) [164]. Deviations from these assumptions matter most if they are satisfied differently in the groups being compared, for example if censoring is more likely in one group than another.

The log rank test is most likely to detect a difference between groups when the risk of an event is consistently greater for one group than another. It is unlikely to detect a difference when survival curves cross.

As a result of the log rank test being purely a test of significance, it cannot provide an estimate of the size of the difference between the groups. In this scenario a typical methods to assess the significance of the difference would be to use the Cox proportional hazards model.

4.2.4 Cox proportional hazards model

The proportional hazards method computes a coefficient for each predictor variable that indicates the direction and degree of flexing that the predictor has on the survival curve (Kumar et al, 1994) [165]. A value of zero indicates that a variable has no effect on the curve – it is not a

predictor of survival at all. The model functions on the basis that the underlying hazard rate (as opposed to the survival time) is a function of the clinical variables (such as tumour size, grade) with no assumptions made as to the characteristics of the hazard function. The hazard function can be thought of as an individual's death in the immediate future with the assumption that the individual has survived up to a present point in time (Kumar et al, 1994) [165]. Alternatively, the hazard function can be thought of as an equivalent function for survival or death. Once enough covariates are determined, it could be possible to create a customised survival curve for any particular combination of predictor values. More importantly, the method provides a measure of the sampling error associated with each predictor's coefficient. The method is widely used as it is not dependent on any assumptions regarding the underlying survival distribution.

In a cancer setting, the goal with the Cox model is to identify indicator variables whereby survival characteristics are compared between two or more groups. To describe the effect covariates, such as race and stage, have on survival, the hazard function has been shown in literature to perform well (Kumar et al, 1994) [165]. As shown in figure 4.2, the chances of death in humans are very high immediately after birth however for many years, the chances of death plateaus until later in life where the chances of death increase sharply (Kumar et al, 1994) [165]. The onset of cancer can therefore affect these events greatly and determining what particular variables have contributed to acceleration in death could be significant in understanding cancer methods of action.

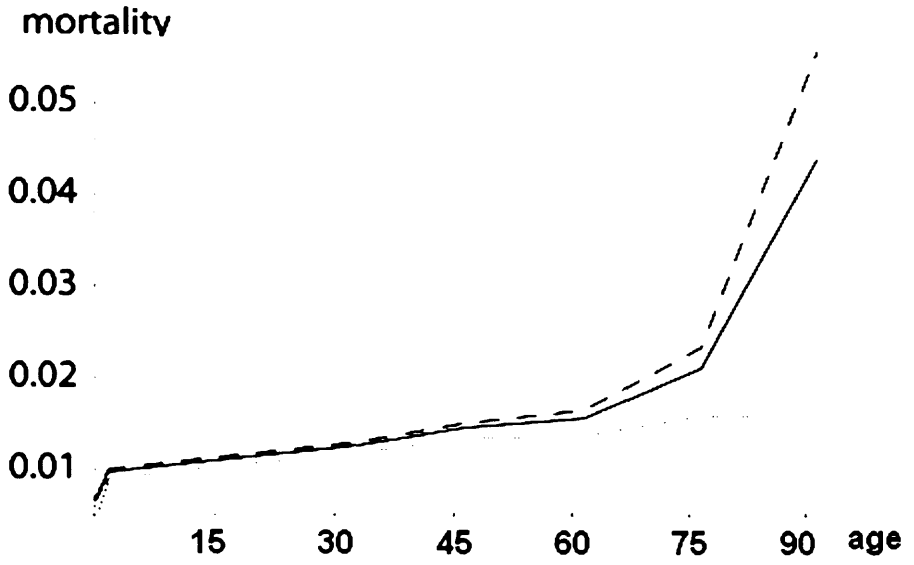


Figure 4.2: Mortality in humans showing the high risk at birth, a plateau period until old age where the chance of death is high.

As a result, the Cox proportional hazard model would be very valuable to assess the impact that a particular combination of patient variables has on survival time.

4.3 Aims and Objectives

The SEER dataset provides a source of patient information for multiple cancer types such as breast cancer and colorectal cancer. There are over 40 different patient variables for each patient. If multiple combinations of patient variables from the SEER dataset could be routinely searched , this would provide a very powerful data source which together with previously outlined statistical techniques highlight how different patient variables impact survival. Consequently, development of an online search tool to explore such patient variables has was therefore proposed.

The aims of this chapter include:

1 Preparation of the SEER cancer patient dataset to facilitate high throughput data searching and analysis.

2 Development of a cancer survival query tool utilising the SEER patient dataset encompassing routine statistical analysis and modelling techniques. The tool would facilitate the exploration of survival patterns in breast and colorectal cancer based on different patient attribute combinations.

3 Application of the developed cancer survival query tool to demonstrate the effect of different patient variable combinations on survival outcome. This would be assessed by significance determination using modelling techniques such as the Cox proportional hazard model among others.

4.4 Strategy for Development

To fulfil the requirements of the aims the following steps were taken in development:
1 – Transformation of the SEER dataset according to the SEER coding manual into a purely numerical form.
2 – Storage of the transformed dataset into a multi-user database – Microsoft SQL server.
3 – Creation of a web based interface together with statistical programming to provide statistical validation of results from different patient variable combinations.
4 – Development of a service to facilitate sharing of the capabilities of the SEER cancer patient data set in a secure manner so that it can be made available to the cancer research community.

4.5 Cancer Survival Query Tool Architecture and Implementation

The name 'Superstes' (latin for 'survival') was chosen for the cancer survival query tool and will be referred to in all future sections.

4.5.1 Providing a cancer patient data resource

During development and application of the Superstes tool, the SEER patient dataset has provided the source of clinical data due to the wealth of individual patient variables it contains. Data has been extracted from the database for 70,000 breast cancer patients diagnosed between 1992 and 2002 encompassing 17 prognostic variables including tumour size, grade, lymph node status, *ER*

and *PgR* status. Data was extracted between this time period in order to examine variables in relation to 10 year survival. The period of 10 years chosen mirrors a similar time period to that used in the original Nottingham prognostic index study of variables predicting good, moderate and poor prognosis in 1982 (Galea et al, 1992) [7].

By working with a similar dataset (as the original identical dataset used to generate the NPI is not available), it was hoped that the SEER-derived dataset can be considered as representing a similar patient cohort for exploration and development of new prognostic models for breast cancer. Although the primary focus were to develop and apply a cancer query survival tool to explore new prognostic variables for breast cancer, data for colorectal cancer patients was also explored in order to provide proof of principle that different patient factor combinations and the resulting survival impact can be studied for other cancer types.

4.5.2 SEER patient dataset transformation – preparation for database storage

The breast and colorectal datasets were extracted from a master SEER patient lists obtained through a license agreement. The dataset received from the SEER program was in the form of a spreadsheet. Microsoft Excel was then used to edit this file to structure it into a table format ready for insertion into the database for querying in the developed cancer query tool – Superstes. The encoding was performed by searching through the dataset to standardise coding used to describe different patient variables. For example, where roman numerals for Stage such as I, II, III and IV were used, these were replaced by the numbers 1, 2, 3 and 4. Key patient variables from the dataset for breast and colorectal cancer included tumour grade, tumour site, extent, number of primary tumours, histology, tumour size, ER status, PgR status where appropriate) were carefully converted into a purely numeric form for storage using a Microsoft SQL database. This was performed with assistance from the SEER coding manual for the various patient variables [100]. A full summary of all patient variables in both breast and colorectal cancer used to create the database from the SEER cancer patient dataset can be found in table 4.1 for breast cancer and table 4.2 for colorectal cancer. Appendix 3 summarises the complete list of patient variables were altered in the database to allow searching and statistical analysis of the dataset.

Breast cancer patient variables	Description (database code)
Race	Ethnic origin e.g: white (2)
Year	Year of diagnosis e.g: 1997 (1997)
Histology	Tumour type e.g: papillary carcinoma (8050)
Tumour site	Location of tumour e.g: lower inner quadrant (503)
Tumour grade	Grade of the tumour e.g: 2 (2)
Cause of death	Is the patient alive or dead e.g: 1 or 0
Nodes examined	Number examined e.g: 50
Positive nodes	Number of nodes e.g: 25
Tumour extent	Tumour and surrounding involvement information e.g: Invasion of subcutaneous tissue (2)
Tumour size	In millimetres e.g: 12 (12)
Age	Patient age e.g: 45 (45)
Surgery received	Type of surgery e.g: partial mastectomy (10)
Radiation received	Type of radiation procedure e.g: beam (3)
Radiation sequence surgery	Order of treatment received e.g: radiation prior to surgery (3)
Martial status	Marital status of patient e.g: single (3)
Number of primaries	Number of primaries e.g: 2 (2)
PgR status	Progesterone receptor status e.g: positive (1)
ER status	Oestrogen receptor status e.g: positive (1)
Survival time in months	Time in months e.g: 34 (34)
Patient ID	Anonymous patient ID e.g: 9584

Table 4.1: Summary of the different patient variables describing each patient in the SEER breast cancer dataset. The description column contains a brief overview of each variable with the database code in brackets.

Numerous database projects have previously shown that computing performance is greatly enhanced using numbers as opposed to repeated strings of text. This is particularly important when considering speed of retrieval of a large queried dataset and its subsequent processing, since in this instance the databases examined contained over 70,000 entries for breast cancer and 90,000 entries for colorectal cancer. Moreover, the mathematical engine 'R' requires the data to be in a numerical format to perform some functions, sometimes binary according to the function to be processed. Data standardisation is symbiotic with ease of use, ensuring interfaces where users can make selections from drop down boxes on website forms that are correctly integrated at all levels. Once the dataset was in a satisfactory form, the data was inserted into the Microsoft SQL database.

Colorectal cancer patient variables	Description (database code)
Race	Ethnic origin e.g: black (1)
Sex	Male or Female e.g: male (1)
Year	Year of diagnosis e.g: 1985 (1985)
Histology	Tumour type e.g: tubular adenocarcinoma (8211)
Tumour site	Location of tumour e.g: cecum (5)
Tumour grade	Grade of the tumour e.g: 1 (1)
Cause of death	Is the patient alive or dead e.g: 1 or 0
Nodes examined	Number examined e.g: 23
Positive nodes	Number of nodes e.g: 12
Tumour extent	Tumour and surrounding involvement information e.g: muscularis mucosae (12)
Tumour size	In millimetres e.g: 23 (23)
Age	Patient age e.g: 34 (34)
Surgery received	Type of surgery e.g: total colectomy (5)
Radiation received	Type of radiation procedure e.g: radioactive implants (3)
Martial status	Marital status of patient e.g: single (3)
Number of Primaries	Number of primaries e.g: 2 (2)
Radiation sequence surgery	Order received e.g: radiation after surgery (2)
Survival time in months	Time in months e.g: 50 (50)
Patient ID	Anonymous patient ID e.g: 9584

Table 4.2: Summary of the different patient variables describing each patient in the SEER colorectal cancer dataset. The description column contains a brief overview of each variable with the database code in brackets.

4.5.3 Visual Basic.net and Visual Studio

Visual Basic.net is an update of Microsoft Visual Basic which was used in the development of Informatics Tenovus (I-10) in Chapter 2. It was designed to meet the growing needs for web based applications which run on many different computer operating systems, commanded through a web browser (such as Microsoft Internet Explorer). It is an alternative to the programming language Java and a technology called the Java virtual machine which allows Java written applications to run locally through a user's web browser.

During development it was fortunate to have access to a Microsoft Windows server – a computer operating system optimised to serve web pages for extended periods of time. This was another

fundamental choice in selection of Visual Basic.net for the cancer survival query tool development.

Visual Basic.net applications are created using a Microsoft application called Visual Studio.net which is very similar to the application Visual Basic 6.0 as outlined in Chapter 2, Section 1.3. This application was used to design all technical aspects of the cancer survival query tool.

Due to prior experience with Visual Basic, it was a natural progression to be used for development of the cancer survival query tool. A key advantage determined early in development was the similarities to Visual Basic in commanding the statistical programming environment 'R' through the R-(D) COM interface.

4.5.4 R-(D)COM

As previously demonstrated in Chapter 2, Section 2.3.1, the query tool communicates with 'R' for statistical analysis of any given patient selection using the R-(D)COM interface. It is a Microsoft Windows operating system only based interface. During development it was fortunate that the interface was compatible with Visual Basic.net as documentation examples focus mainly upon the older Visual Basic programming language. Consequently, this allowed the usage of the 'R' statistical programming environment outline in Chapter 2, Section 2.2.1, to perform the statistical analysis for the developed cancer query tool.

4.5.5 Web service development

The query interface, for the cancer survival query tool, communicates with the SEER patient database indirectly for added security via web services. Web services act as an interface with data sources (such as a database) and user interfaces (such as the cancer survival query tool). The user interface and database cannot access each other directly. Only the web service knows how to communicate between the two parts of the system.

Due to the requirement of obtaining the SEER dataset via license agreement, one of the prerequisites for access is to not allow public distribution of the patient dataset. Due to the sensitive nature of patient data, even when anonymous, web service development is also beneficial for addition of future patient datasets. It enables such datasets to be added with full confidence of data protection.

Consequently, query results managed via the web service are processed using 'R' via a D-COM interface as demonstrated previously in the development of I-10 in chapter 2. This plays an important role in enabling the generation of Kaplan Meier survival curves, Cox proportional hazard models and log rank tests from a particular subset of patient variables which the user specifies via the user interface. Figure 4.3 outlines the interactions between the different components in the cancer survival query tool (Superstes).

Use of web service technology in Superstes has two main advantages. Firstly, as previously introduced, the underlying database structure, contents and queries cannot be seen or controlled in any way other than what has been hard coded into the web service. For example, web methods to update, delete or insert data cannot be performed unless it is encoded, protecting the stored data. Secondly, the underlying database technologies used to store and retrieve the data does not need to be understood by the user. Applications simply make a request and get a result in a predefined format. This makes Superstes cross platform compatible as a result of using a web service communication protocol called 'SOAP'. SOAP gives the potential for multiple applications to connect to the web service at any one time via the Internet. This facilitates multiple queries from Superstes and the potential for multiple simultaneous access of the SEER patient dataset. The only information developers using web service technology require is the hierarchy in which variables are passed and retrieved by queries of the dataset. The pivotal role the web service plays in the interactions between the components of Superstes is summarised in Figure 4.3.

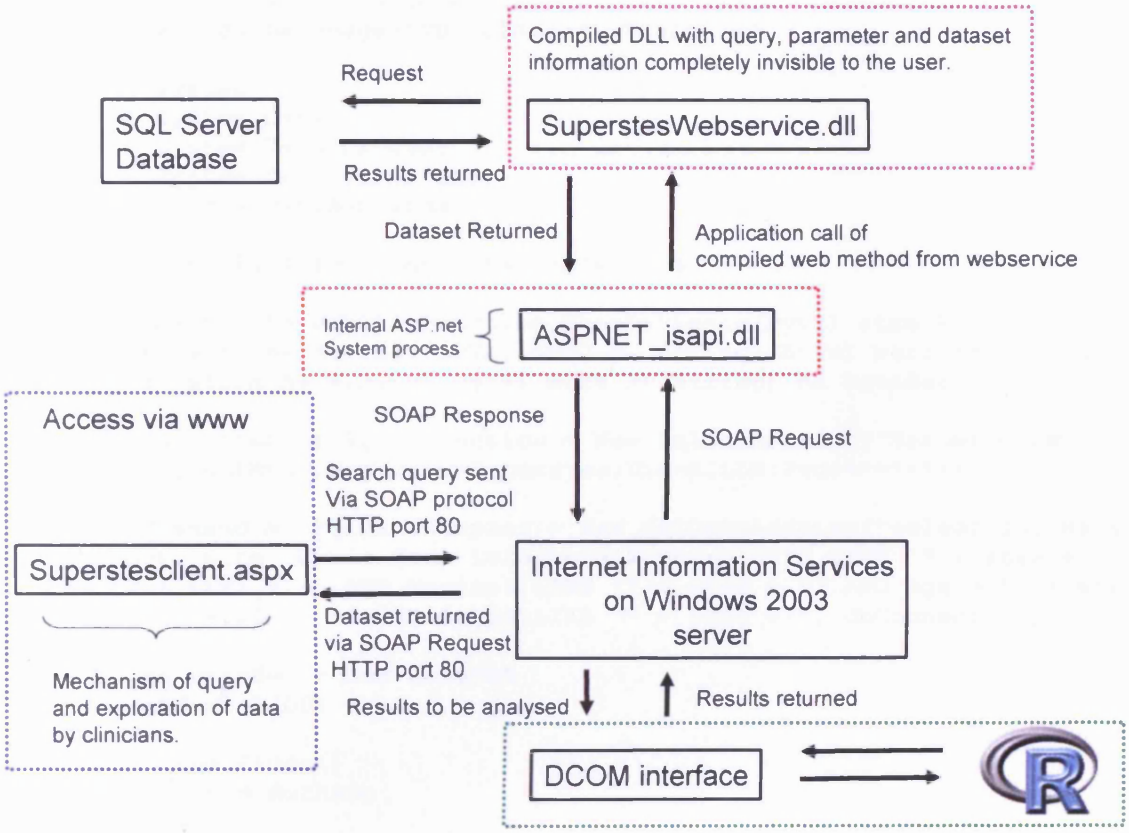


Figure 4.3: Interactions between the different components of Superstes and the pivotal role of the web service.

Figure 4.3 indicates key system features of Superstes using web services. The diagram shows the flow of information starting with a search of a particular set of patient variables in the user interface – Superstesclient.aspx. Data corresponding to relevant clinical variables is then requested from the SEER patient database. The results are processed in 'R' via the D-COM interface before finally being returned to the user interface page – Superstesclient.aspx. The code for the web service request is written as a file called an 'asmx' file in the programming language visual basic.net, as shown in code 4.1.

```
<%@ WebService Language="VB" Class="GetInfo" %>

Imports System
Imports System.Data
Imports System.Data.DataSet
Imports System.Data.SqlClient
Imports System.Web.Services

Public Class GetInfo : Inherits WebService

    <WebMethod()> Public Function ShowPatients(ByVal stra As String, _
    ByVal str1 As String, ByVal str2 As String, ByVal str3 As String, _
    ByVal str3a As String, ByVal str4 As String) As DataSet

Dim dbConnection As SqlConnection = New SqlConnection("Server=BIOSCI-
WINDOWS\SQLEXPRESS;Database=Superstes;Uid=HILLN;Pwd=********")

Dim objCommand As SqlDataAdapter = New SqlDataAdapter("select ID, Race, Sex,
Histology, Site, Grade from Colorectala where CTYP LIKE '" + stra + "' AND Race
LIKE '" + str1 + "' AND Marital LIKE '" + str2 + "' AND Age > '" + str3 + "' AND
Age < '" + str3a + "' AND Grade LIKE '" + str4 + ", dbConnection)

Dim DS As DataSet = New DataSet
objCommand.Fill(DS)
Return DS
dbConnection.Close()
dbConnection = Nothing

End Function

End Class
```

Code 4.1 – Creation of the web service for accessing, for example, the colorectal dataset.

The code is saved with the file extension .asmx which is then compiled at the command line using the 'wsdl' compiler which requires the specification of important system *.dll files which are required for compilation to be successful. This information is supplied by typing a particular set of commands at the command line to initiate the compiler. A most important step is the specification of a unique 'namespace' which is a requirement for web service functionality. The web site navigation (URL route structure e.g: http://137.44.25.44/Superstes/) is a requirement for functionality so that the Web service routes information to the correct locations if it was ever hosted on a different machine to that supplying the data source. A different web service was created for the two cohort search as more parameters are sent and received to enable the comparison to be made.

The compiler generates a *.vb file from the asmx file which creates the 'nameservice' calls which ultimately deals with the web service protocol information – in this case the protocol SOAP. An example of this code is shown in code 4.2 using the previously shown .asmx file for a single cohort query as an example.

```
Option Strict Off
Option Explicit On

Imports System
Imports System.ComponentModel
Imports System.Diagnostics
Imports System.Web.Services
Imports System.Web.Services.Protocols
Imports System.Xml.Serialization

'
'This source code was auto-generated by wsdl, Version=1.0.3705.0.
'
Namespace DataService1

    '<remarks/>
    <System.Diagnostics.DebuggerStepThroughAttribute(), _
     System.ComponentModel.DesignerCategoryAttribute("code"), _
     System.Web.Services.WebServiceBindingAttribute(Name:="GetInfoSoap",
[Namespace]:="http://tempuri.org/")> _
    Public Class GetInfo
        Inherits System.Web.Services.Protocols.SoapHttpClientProtocol

        '<remarks/>
        Public Sub New()
            MyBase.New
            Me.Url = "http://localhost/colrec/GetInfo.asmx"
        End Sub

        '<remarks/>

<System.Web.Services.Protocols.SoapDocumentMethodAttribute("http://tempuri.org/Sh
owPatients", RequestNamespace:="http://tempuri.org/",
ResponseNamespace:="http://tempuri.org/",
Use:=System.Web.Services.Description.SoapBindingUse.Literal,
ParameterStyle:=System.Web.Services.Protocols.SoapParameterStyle.Wrapped)> _
        Public Function ShowPatients( _
                    ByVal stra As String, _
                    ByVal str1 As String, _
                    ByVal str2 As String, _
                    ByVal str3 As String, _
                    ByVal str3a As String, _
                    ByVal str4 As String, _
                    ByVal str5 As String,
                    ) As System.Data.DataSet
            Dim results() As Object = Me.Invoke("ShowPatients", New Object()
{stra, str1, str2, str3, str3a, str4, str5, str6, str7, str8, str9, str9a, str10,
str11, str12, str13, str14, str15, str16, str17, str17a})
            Return CType(results(0),System.Data.DataSet)
        End Function
```

Code 4.2..continued overleaf

```
...Code 4.2 continued from previous page

'<remarks/>
        Public Function BeginShowPatients( _
                    ByVal stra As String, _
                    ByVal str1 As String, _
                    ByVal str2 As String, _
                    ByVal str3 As String, _
                    ByVal str3a As String, _
                    ByVal str4 As String, _
                    ByVal str5 As String,)
                    ByVal callback As System.AsyncCallback, _
                    ByVal asyncState As Object) As System.IAsyncResult
            Return Me.BeginInvoke("ShowPatients", New Object() {stra, str1, str2,
str3, str3a, str4, str5, str6, str7, str8, str9, str9a, str10, str11, str12,
str13, str14, str15, str16, str17, str17a}, callback, asyncState)
        End Function

        '<remarks/>
        Public Function EndShowPatients(ByVal asyncResult As System.IAsyncResult)
As System.Data.DataSet
            Dim results() As Object = Me.EndInvoke(asyncResult)
            Return CType(results(0),System.Data.DataSet)
        End Function
    End Class
End Namespace
```

Code 4.2: Compiled web service containing programmatic instructions of where the SEER patient data set resides and how to make a request of certain patient variables of the data set.

A key relationship which makes the system quite unique is the way in which the queries become result graphs. Results from the web service query are returned to the application as a dataset which any Visual Basic.net system can use. Once it has been returned as a dataset, the results can be read into an array format in the client application (as in the case of the user interface of Superstes). This array can then be passed into 'R' as a data matrix using the R-(D)COM interface. 'R' is used to generate histograms for the different patient variables such as sex, martial status or age range for example. It is also used to generate Kaplan Meier Survival curves, log rank tests and Cox proportional hazard models. As each histogram is written as an image file (called a JPEG), this not only enables the user to see the image in their browser window, it presents the option of saving to a file for use in presentations/documents.

4.5.6 User interface design and statistical capabilities of Superstes

A key aim with Superstes development was to produce an easy-to-use tool (ideally with the information represented in a uniform way for different cancer types i.e. breast, colorectal). The need for clear, concise interfaces was a lesson learned early during development of I-10. It was possible to create such a user interface for Superstes as a visual basic.net application in visual studio as previously introduced. It was created as an *.aspx page type with an accompanying *.vb script which contains functions which command 'R' via the (D)-COM interface to perform the different statistical techniques. Ultimately, when interpreted through Windows IIS (Internet Information services) the application is operated through a web browser. Although the bulk of the code for the Superstes application can be found on the accompanying CD-ROM, in this section important examples from the code are highlighted where there are novel coding features central to the function of Superstes. A master template, which contains the structure and layout of the application, was applied and also used to form the tree menu structure for cancer type selection, and the tab system which contains headings for analysis options. The menu and tab system was created in the programming language Javascript which is widely used for web site design [89]. At the top of every .aspx page is a link which loads the master layout file for Superstes and the code page containing the *.vb script file as shown in code 4.3.

```
<%@ Page Language="VB" MasterPageFile="~/superstes.master"
AutoEventWireup="false" CodeFile="sup.aspx.vb" Inherits="sup" title="Superstes -
Cancer Survival Query Tool" %>
```

Code 4.3: Initial code section which is loaded when Superstes is first launched in a web browser.

Although .aspx pages support most html formatting standards, the ways in which dynamic or server processed information is handled is different. One example is the selection of drop down (clickable) boxes on traditional html pages. These are called drop down boxes which contain list items, for example using the "race" drop down list as shown code 4.4.

```
<asp:DropDownList ID="Race" runat="server">

  <asp:ListItem Selected="True" Text="Any" Value="%"></asp:ListItem>
  <asp:ListItem Text="Black" Value="1"></asp:ListItem>
  <asp:ListItem Text="White" Value="2"></asp:ListItem>
  <asp:ListItem Text="Other" Value="3"></asp:ListItem>
  <asp:ListItem Value="4">Unknown</asp:ListItem>

</asp:DropDownList>
```

Code 4.4: Example of how a 'DropDownList' – which provides the user with a choice of different race shown here using the Visual Basic.net programming language.

As in traditional web page forms which the user completes in a web browser, the value of each item are then passed to the server as part of a query. For example, in code 4.4 if patients who are white need to be selected, this corresponds to a value of '2' in the database.

Connections to 'R' and to the SQL database are handled within the accompanying *.vb file of the parent *.aspx page. Certain system components are required upon page load. Connectivity to 'R' in Superstes is initiated in a similar way as to that in visual basic in I-10, but with slight syntax changes as shown code 4.5.

```
Imports System
Imports System.Data
Imports System.Data.DataSet
Imports System.Data.SqlClient

Partial Class sup

  Inherits System.Web.UI.Page
  Dim supInfo As DataService1.GetInfo
  Dim MyData As Data.DataSet
  Dim sconn As STATCONNECTORSRVLib.StatConnector  // Initialised upon page load

Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load

sconn = New STATCONNECTORSRVLib.StatConnector() // New connection made

sconn.Init("R")                                 // 'R' initialised
```

Code 4.5: Initialisation of the connection to 'R' using visual basic.net

Once these statements are executed, 'R' communication is established and data handling for SQL when creating and retrieving query results is initiated. Imports System.Data.SqlClient contains a set of methods for communication with the Microsoft SQL database.

As each query page is divided into a series of tabs for ease of navigation, all page elements are loaded together. For stylistic purposes, when the page is initially loaded and no histogram is present on the results tab, a simple blank image is displayed in its place until 'R' has produced the requested histogram. The code to produce this effect is shown in code 4.6 with the result shown in figure 4.4.

```
Image1.ImageUrl = "http://137.44.25.44/Superstes/blank.jpg"
```

Code 4.6: Method for insertion of a blank image for styling purposes in Superstes.

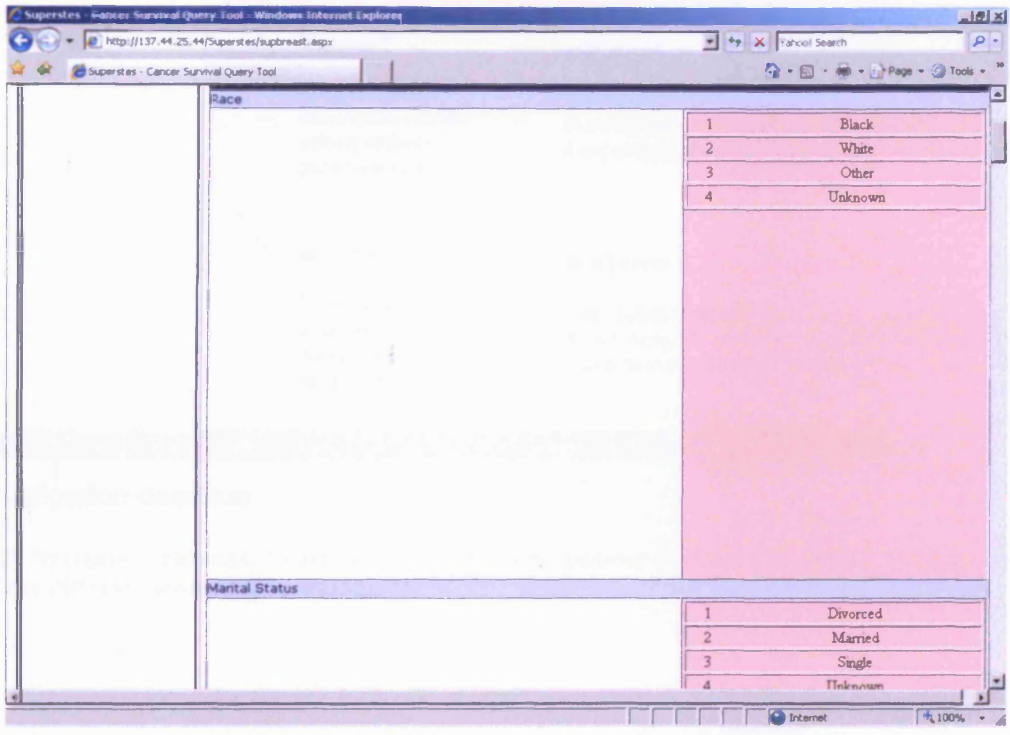


Figure 4.4: The effect of inserting a blank image file into Superstes before any results are loaded.

An overview of navigation through Superstes is shown in figure 4.5. The first choice the user makes in Superstes is cancer type - colorectal or breast cancer. The opening overview page of Superstes at http://137.44.25.44/Superstes/ can be seen in figure 4.6. Subsequent options

available on the search pages will vary slightly according to the type of cancer initially chosen (e.g. no steroid receptor covariate boxes for colorectal cancer). Cancer type is chosen from a tree menu structure to the left of the page with the main text body containing contact information of the developers, a brief application explanation, and the location for any main update information.

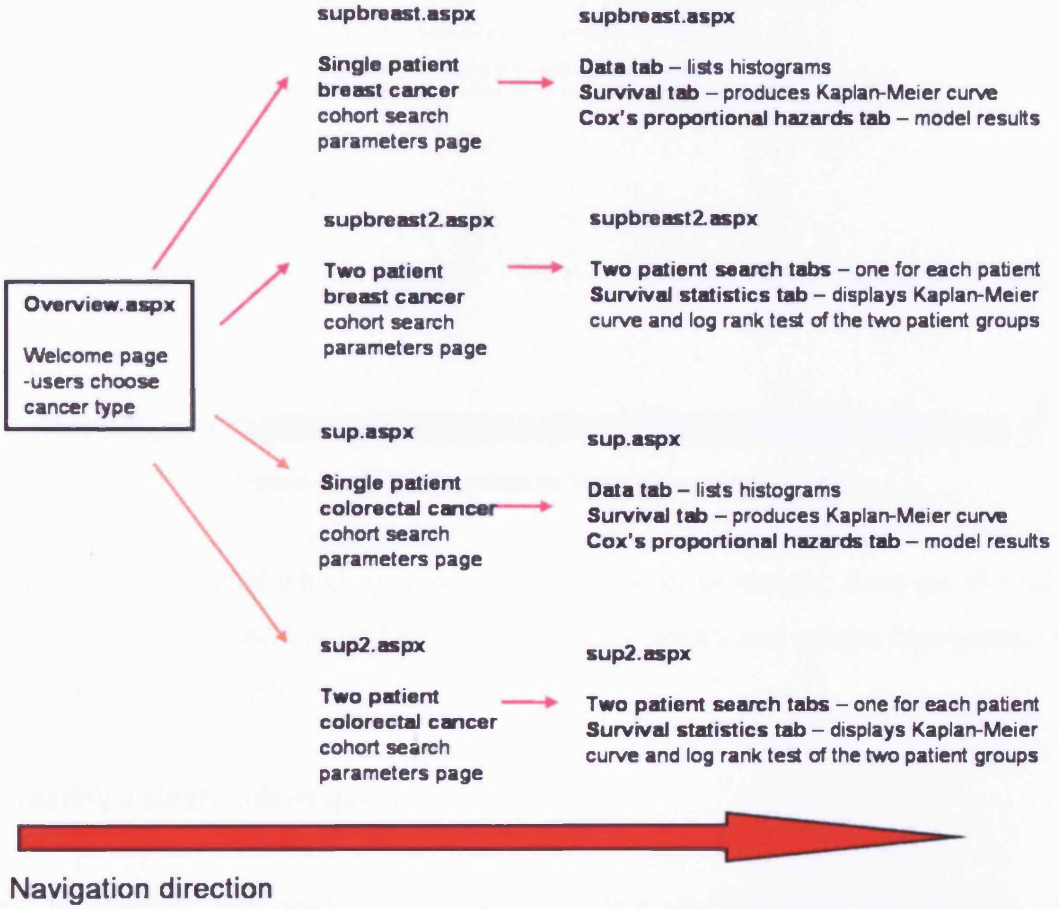


Figure 4.5 Navigation through Superstes from choosing patient variables to search through to obtaining results of the different modelling techniques.

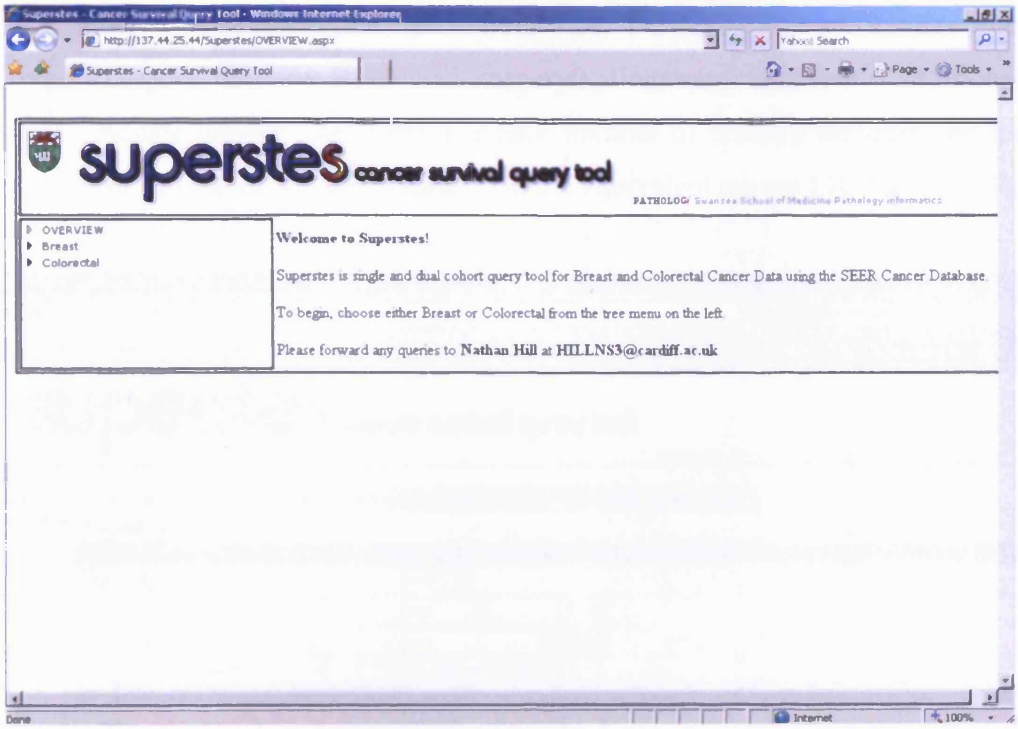


Figure 4.6 – Opening page to Superstes – overview.aspx

Once the user has selected which type of cancer they wish to analyse, there are five tabs which contain user interface selections and are used to display results and graphs. Navigation occurs as reading logically left to right.

4.5.7 Creating a single cohort query

The first tab contains drop down menu selections with a selection of patient variables which the user can choose to query for breast or colorectal cancer, with single or two cohort patient options. Figure 4.7, shows a typical selection of options available for a single cohort colorectal patient search. Histogram results for each covariate selected under the single cohort tab (martial status, age etc.) will appear under the tab 'Data' while the tab 'Survival statistics' contains Kaplan Meier plots describing survival and Cox-Proportional hazard modelling results appearing under the tab 'Cox-Proportional hazard models'. For example, under the 'Colorectal – single cohort' tab, search parameters that can be selected using tick boxes are divided into four sections: patient, tumour, treatment and additional tumour characteristics. Patient options available across both cancer types include race, age, marital status, and their age. Tumour options include grade, nodes

examined, number of positive nodes and tumour size. Treatment options include radiotherapy (and its type), surgery (and its type) and surgery/radiotherapy sequence. Additional tumour characteristics include tumour site, tumour extent, number of primary tumours and the tumour histology. Additional options for breast cancer on the equivalent tab are ER status and PgR status.

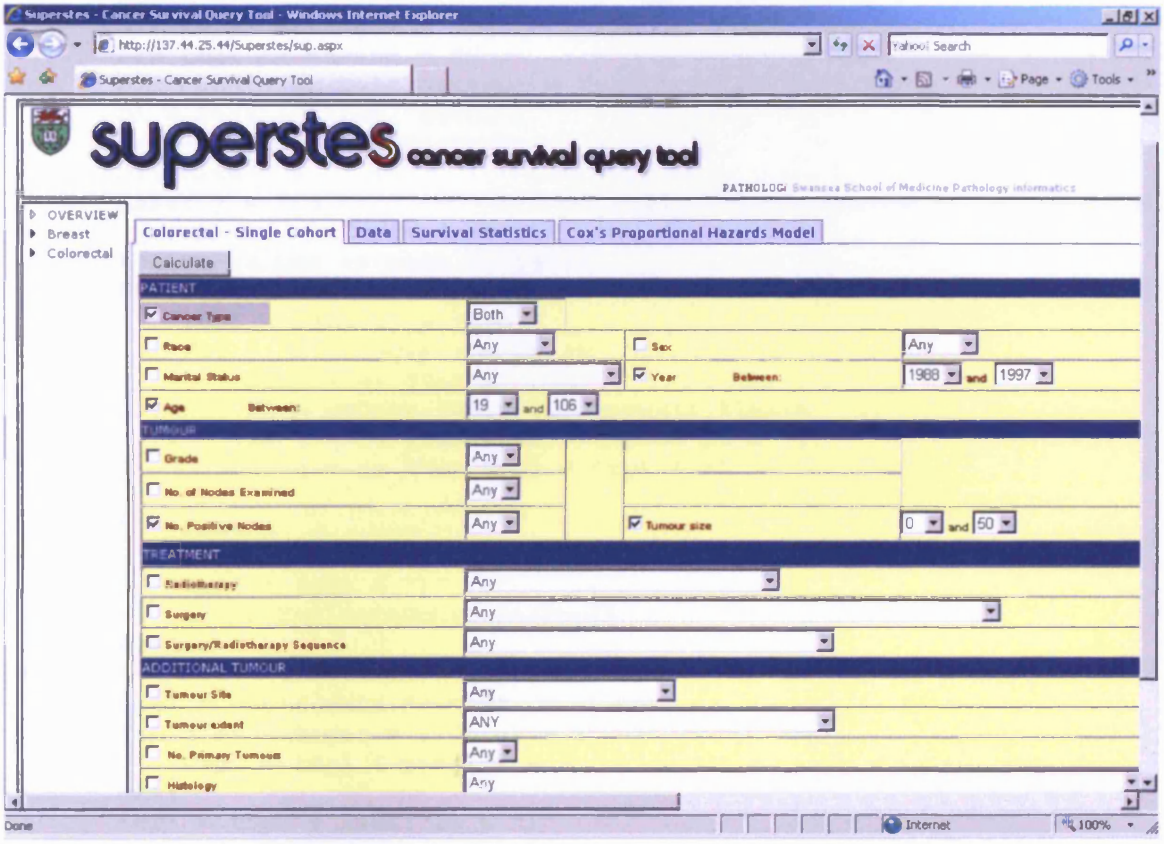


Figure 4.7: Example options available for the single cohort search for colorectal cancer.

Once parameters have been chosen, the system queries the underlying SQL server database and returns results via the website. If a single cohort comparison is being made, the user gets access to an extensive list of drop down boxes showing histograms of each of the search parameters which were available.

The 'calculate' button shown in figure 4.7 initialises a set of 'R' instructions upon submission of the covariate parameters that the user has chosen to query via the web service. The survival curve is initially created using the methodology outlined in code 4.7.

```
Protected Sub SubmitBtn_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Dim arr1(,) As String
        Dim rw, cl As Integer

        'On Error GoTo handle_error
        sconn.EvaluateNoReturn("require (stats)")
        sconn.EvaluateNoReturn("require (survival)")
        sconn.EvaluateNoReturn("require (Design)")
        Dim tm As String = ""
        Dim tmp1 As String = ""
        Dim MyData As Data.DataSet = Session.Item("mydata")
        cl = Session.Item("datcl")
        rw = Session.Item("datrw")
        Dim name(23)
        name(1) = "id"              // remaining 22 removed to condense code

       'Assign values to each variable
        For k As Integer = 1 To cl
            tmp1 = name(k) & "<-c("
            For j As Integer = 0 To rw - 1
                Dim r1 As String
                r1 = MyData.Tables(0).Rows(j).Item(k - 1)
                If j < rw Then tmp1 = tmp1 & r1 & ","
                If j = rw Then tmp1 = tmp1 & r1

            Next j

            tmp1 = tmp1 & ")"
            sconn.EvaluateNoReturn(tmp1)
        Next k

        tmp1 = "dat<-data.frame("
        For j As Integer = 1 To cl - 1
            tmp1 = tmp1 & name(j) & ","
        Next
        tmp1 = tmp1 & name(cl) & ")"
        sconn.EvaluateNoReturn(tmp1)

        Dim varnum As Integer = 0
        Dim varname(23) As String

'############## KAPLAN-MEIER ##############

        Dim survival(rw) As Double
        Dim cod(rw) As Double

        For j As Integer = 0 To rw - 1
            Dim r1 As String
            r1 = MyData.Tables(0).Rows(j).Item(16)
            survival(j) = CDbl(r1)
            'r1 = MyData.Tables(0).Rows(j).Item(18)
            If MyData.Tables(0).Rows(j).Item(18) = "0" Then r1 = "1"
            If MyData.Tables(0).Rows(j).Item(18) = "1" Then r1 = "0"
            cod(j) = CDbl(r1)
        Next
        'Exit Sub
```

..code 4.7 continued overleaf

```
…Code 4.7 continued from previous page
        sconn.SetSymbol("survival", survival)
        sconn.SetSymbol("cod", cod)

        ' Exit Sub
        sconn.EvaluateNoReturn("mod<-survfit(Surv(survival, cod))")
        sconn.EvaluateNoReturn("setwd('c:/Superstes/temp/')")

        Panel3.Controls.Clear()
        'Dim obj As New Object

        sconn.EvaluateNoReturn("library(MASS)")
        obj = Server.CreateObject("Scripting.FileSystemObject")
        filename2 = obj.GetTempName
        sconn.EvaluateNoReturn("jpeg('" & filename2 & ".jpg')")

        ' sconn.EvaluateNoReturn("detach('package:Design')")
        sconn.EvaluateNoReturn("plot(mod, col='red', xlab='Time in Months',
                                                    ylab='Cumulative survival')")

        sconn.EvaluateNoReturn("dev.off()")
        '       Panel3.Controls.Add(im(1))
        Image1.ImageUrl = "http://137.44.25.44/colrec/temp/" & filename2 & ".jpg"
```

Code 4.7: 'R' code in Visual basic.net for production of Kaplan-Meier survival curve.

The Kaplan-Meir survival curve is written to a file and then displayed under the survival statistics tab is shown in figure 4.8. The Cox's proportional hazard model information is also extracted and displayed under the model tab – the detailed coding for the manipulation of the data returned from 'R' can be found on the accompanying CD-ROM. An example of a Kaplan-Meier survival curve and associated Cox proportional hazards model is shown in figure 4.8 and 4.9 respectively.

Superstes empowers the user to assess which variables' coefficients are significantly different from zero; that is: which variables are significant predictors of survival. By exploring each covariate in turn it is hoped much information can be gleamed from the value of one covariate over another. In this respect, Superstes is unique when compared to other cancer survival query tools.

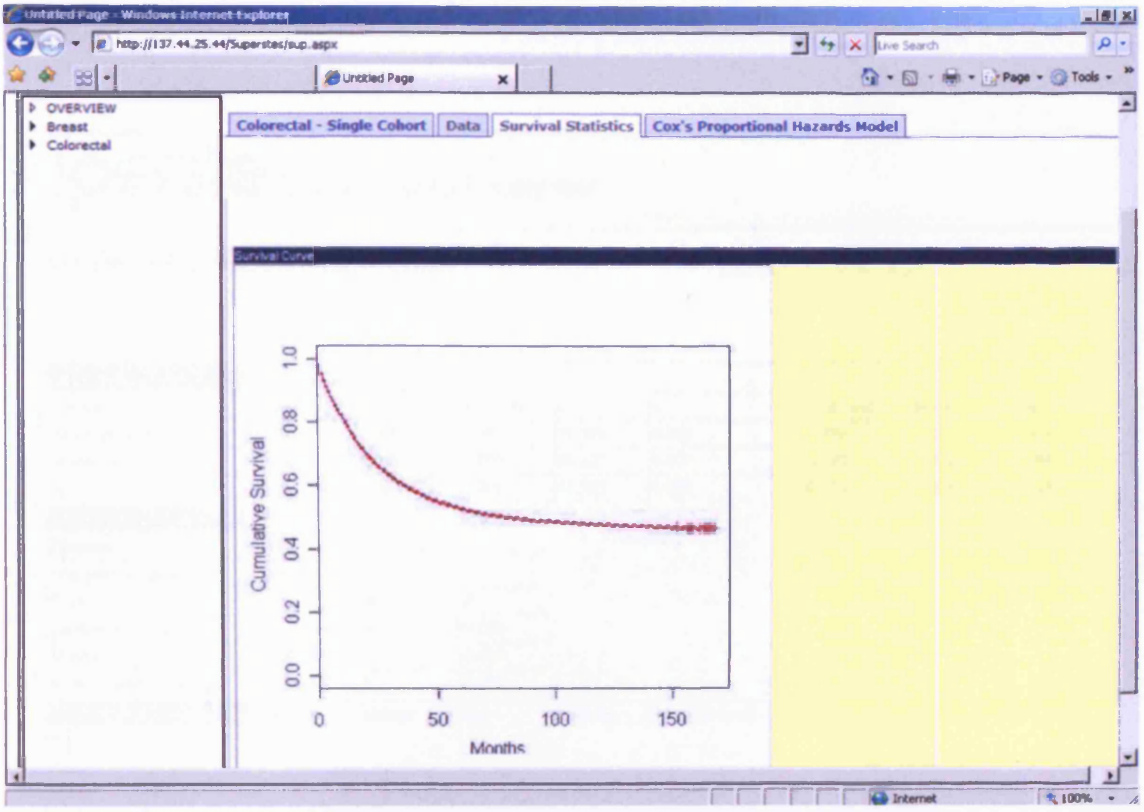


Figure 4.8 – Survival curve from a result of a typical single cohort colorectal query. Full demonstration provided in section 4.6 of this chapter.

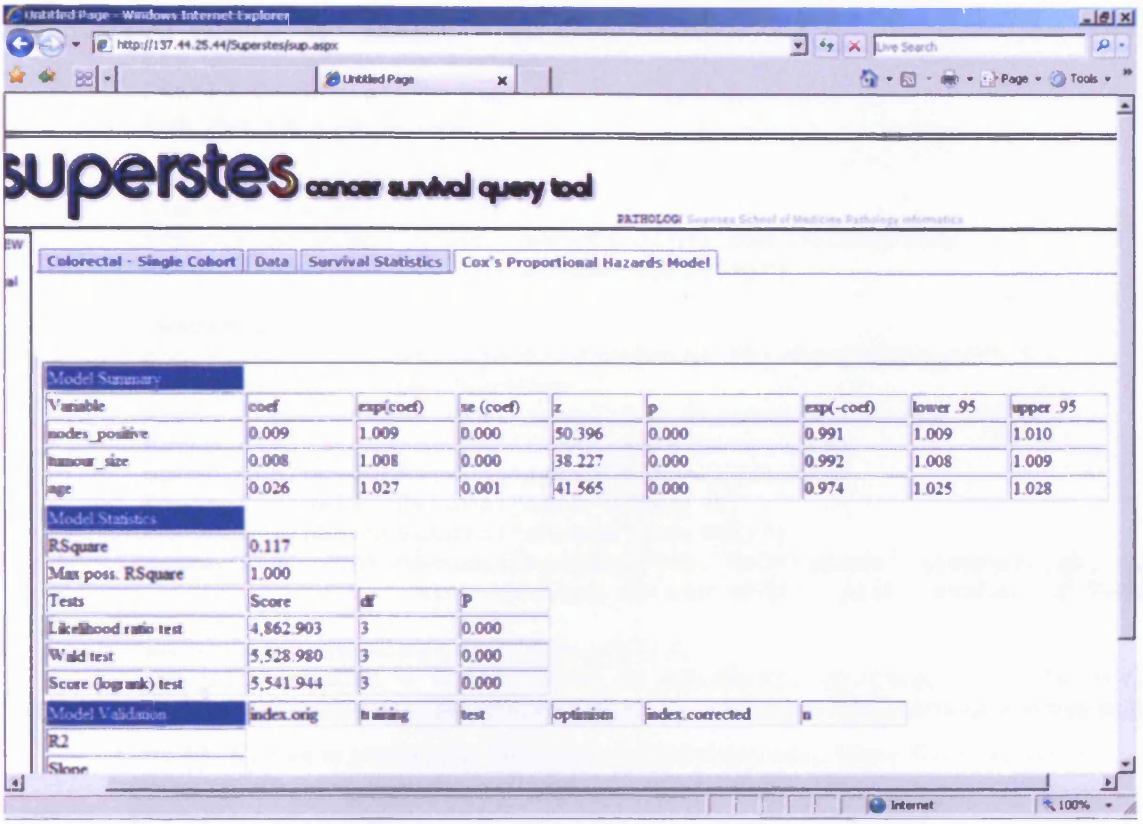


Figure 4.9: Cox's proportional hazard modelling for a single cohort colorectal search result.

Histograms generated under the "data" tab are created using Visual Basic.net and 'R' after the survival curve has been created and are displayed together with their respective attribute codes. All results are returned simultaneously to the user. Coding for histogram generation for the patient attribute marital status is shown in code 4.8 with an example screen capture in figure 4.10.

```
Dim varnum As Integer = 0
 Dim varname(23) As String
 Panel3.Controls.Clear()
 Dim obj As New Object

 Dim filename2 As String
 Dim str1 As String = ""
 sconn.EvaluateNoReturn("setwd('c:/Nathan/colrec/temp/')")
 sconn.EvaluateNoReturn("library(gplots)")

 'marital
 obj = Server.CreateObject("Scripting.FileSystemObject")
 filename2 = obj.GetTempName
 sconn.EvaluateNoReturn("jpeg('" & filename2 & ".jpg')")
 sconn.EvaluateNoReturn("tab<-table(marital)")
 sconn.EvaluateNoReturn("mat<-as.matrix(tab)")
 sconn.EvaluateNoReturn("mat<-t(mat)")
 sconn.EvaluateNoReturn("x<-max(marital)")
 sconn.EvaluateNoReturn("barplot(tab, col='green',xlim=c(1,x), xpd =FALSE,
                 xlab=Marital status code', ylab='Number of Patients')")

 sconn.EvaluateNoReturn("dev.off()")
 Image3.ImageUrl = "http://137.44.25.44/colrec/temp/" & filename2 & ".jpg"
```

Code 4.8: Coding to produce the histogram marital status using Visual Basic.net and 'R'.

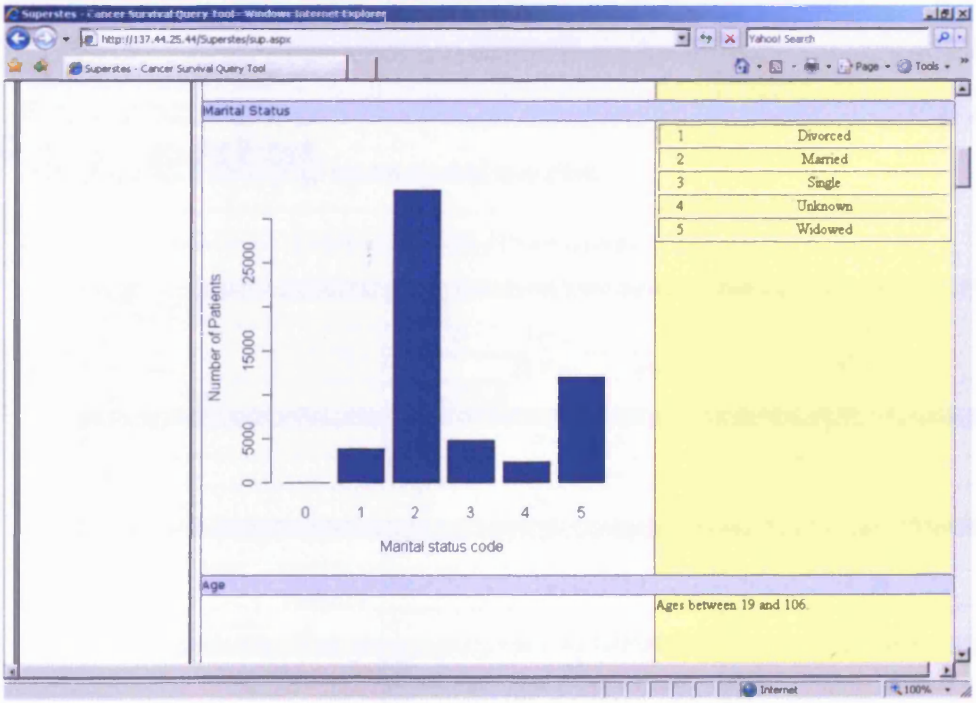


Figure 4.10: Marital status histogram generated under the 'Data' tab in Superstes.

The user should be able to build models and spot trends amongst the data according to the selected patient attribute profile by carefully filtering the database generating successive queries

using Superstes. The interface greatly enhances this process by removing all level of direct SQL database querying by the user.

4.5.8 Creating a two cohort query

A unique and powerful comparison feature within Superstes is the two cohort search option available for either breast or colorectal cancer. Here, the interface is identical to the single cohort query page; however, covariates are selected under two separate tabs for patient group 1 and patient group 2, with results of the query generated as a comparative survival curve by clicking a calculate button on the survival statistics tab as opposed to presenting individual histograms. Thus, the combination effect of one set of patient covariates against another can be explored using the two cohort query function with one Kaplan-Meier survival curve displaying the effect on survival of the selected variables of the two cohorts of patients. As shown in figure 4.11, the tabs have been simplified to show the two patient group interfaces and the results 'survival statistics' which reveals survival curve information.

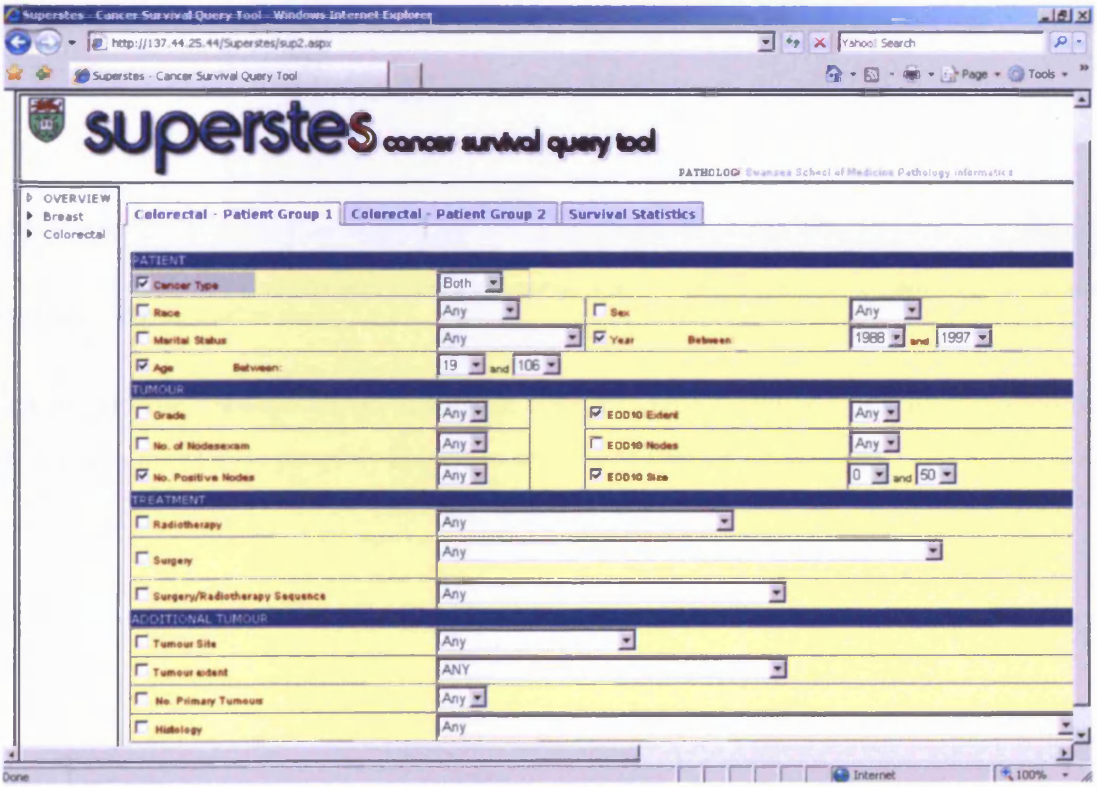


Figure 4.11: The two cohort query page for colorectal cancer in Superstes and the search options available.

A typical Kaplan-Meier survival curve subsequently plotted from an example two cohort colorectal search can be observed in figure 4.12. The outcome based on the search parameters can be compared from the two cohorts on the same graph where patient group 1 is plotted in red and patient group 2 is plotted in blue.

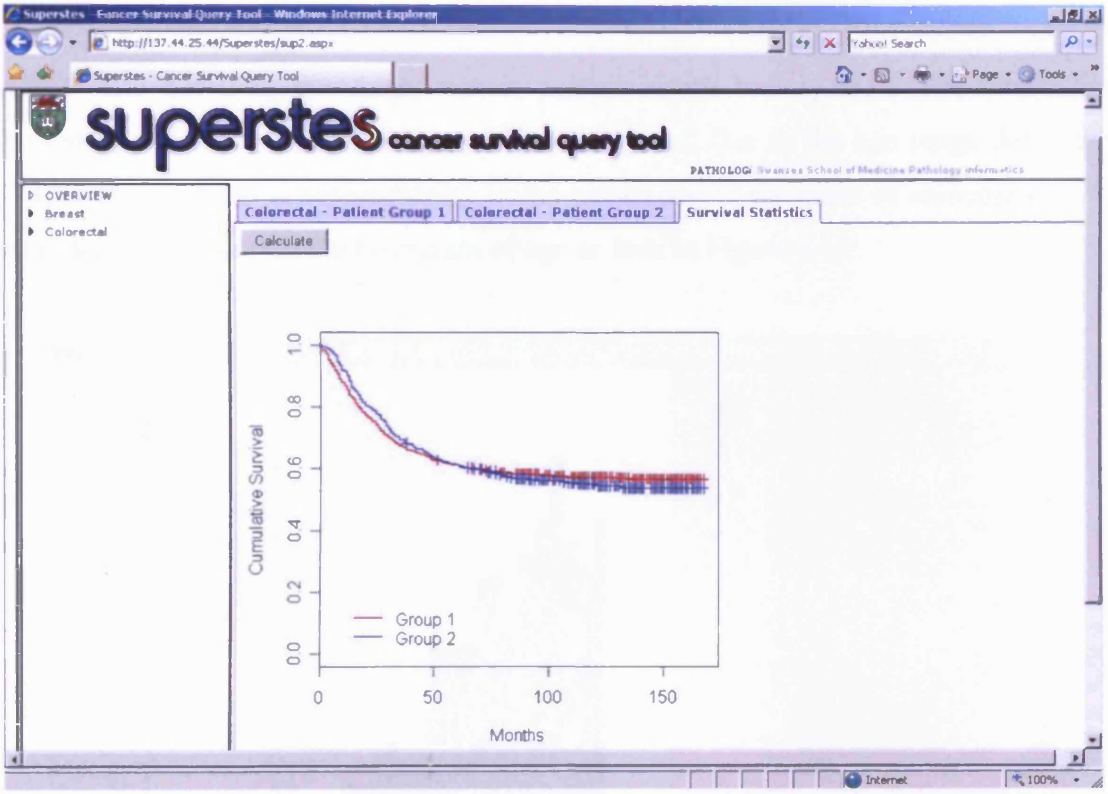


Figure 4.12 – Two cohort survival curve from a colorectal query – patient group 1 highlighted in red and patient group 2 highlighted in blue.

To demonstrate the capabilities of Superstes, four different selections of patient variables were chosen which a user could possibly choose to analyse the effect on survival.

4.6 Examples Usage of Superstes

4.6.1 Case study 1: A group of breast cancer patients who ranged in age between 20 and 50 years of age all developed breast cancer. All of the patients were not married yet developed grade 2 tumours. What is the effect on survival for such a cohort of patients?

To address this question, the single cohort patient search tool is used. The results show a distribution of women towards the age of 50 as expected due to the age range defined in the query. This phenomenon is not surprising in the results due to the onset of menopause, which is evident in the distribution of the histogram of age as seen in Figure 4.13.

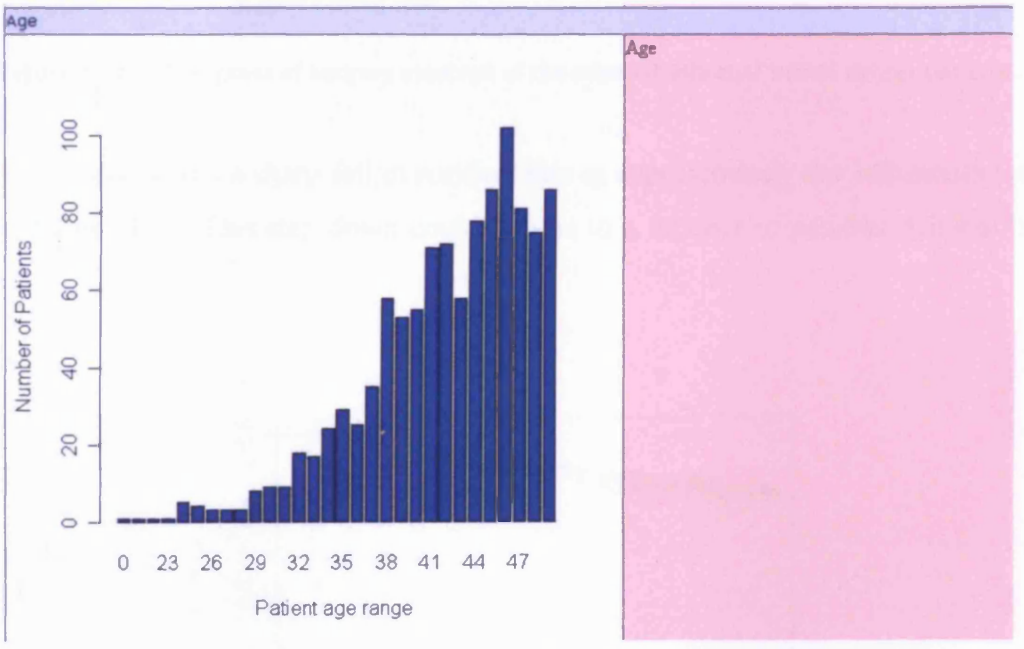


Figure 4.13 – Age of selected subset of patients showing high numbers towards 50, peaking at the age of 45 where over 100 patients had breast cancer.

Most of the patients selected, as seen in figure 4.14, received the very invasive procedure of partial mastectomy with dissection of axillary lymph nodes. In fact, over 500 received this surgical procedure. The remaining patients mostly had modified radical mastectomy, around 350, and around 100 received modified radical mastectomy with reconstructive surgery.

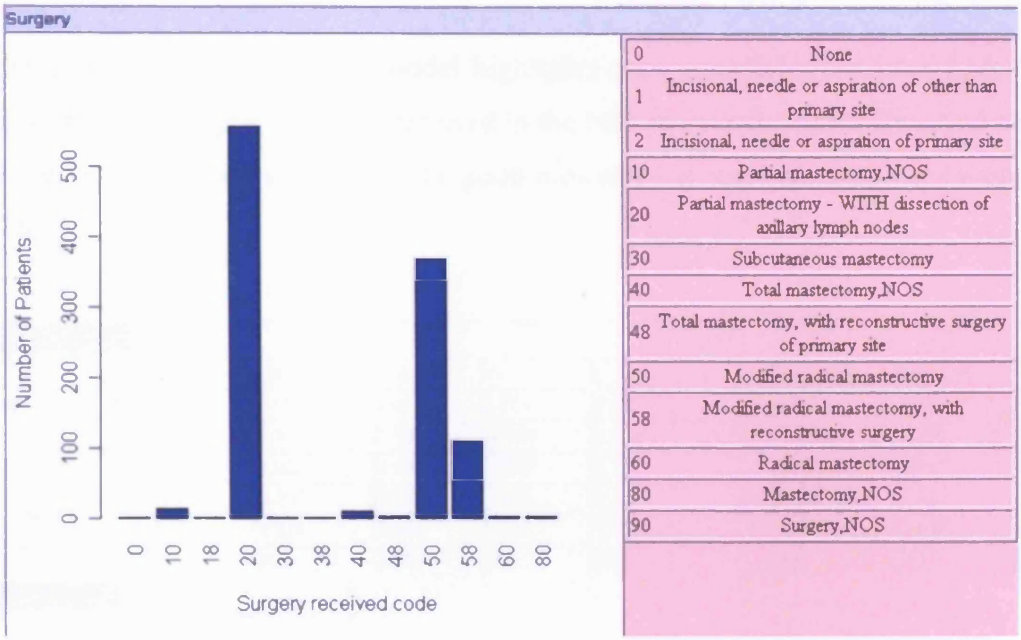


Figure 4.14 – Histogram of surgery received of the selected subset of breast cancer patients.

The survival curve shows a sharp fall in survival rate at approximately the 140 month time point as seen in Figure 4.15. This step down could be due to a number of patients dying at this time point.

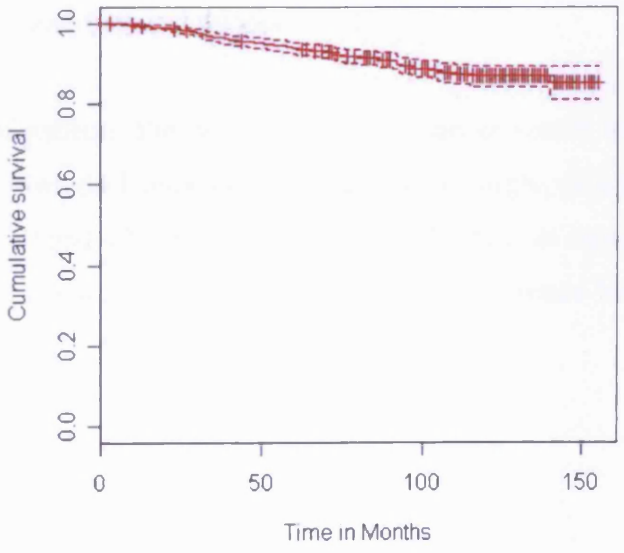


Figure 4.15 – Kaplan Meier survival curve for selected subset of patients from the single cohort search – Vertical bars shown on the plot illustrates where patients have been censored.

Finally the Cox proportional hazard model highlights poor correlation between nodes positive status, tumour size and age – all variables used in the NPI prognostic index. Consequently in this subset, these variables are shown not to be good indicators of survival. This can be observed in figure 4.16

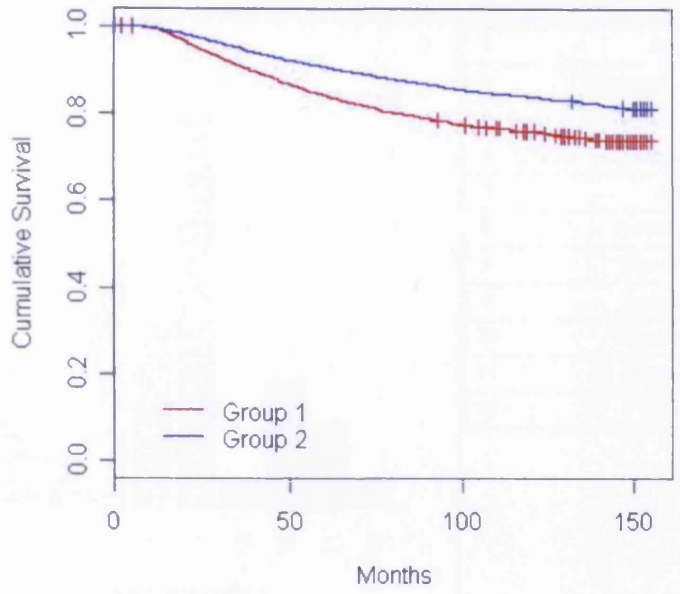
Model Summary								
Variable	coef	exp(coef)	se (coef)	z	p	exp(-coef)	lower .95	upper .95
nodes_positive	0.003	1.003	0.013	0.214	0.830	0.997	0.977	1.030
tumour_size	-0.002	0.998	0.004	-0.456	0.650	1.002	0.991	1.006
age	0.009	1.009	0.006	1.458	0.140	0.991	0.997	1.022
factor(er)1	0.409	1.506	0.296	1.382	0.170	0.664	0.843	2.689
factor(er)2	0.277	1.319	0.304	0.910	0.360	0.758	0.726	2.396
factor(er)3	NA	NA	0.000	NA	NA	NA	NA	NA

Model Statistics			
RSquare	0.007		
Max poss. RSquare	1.000		
Tests	Score	df	P
Likelihood ratio test	7.592	5	0.180
Wald test	7.130	5	0.211
Score (logrank) test	7.181	5	0.208

Figure 4.16 – Cox proportional hazard modelling of the five different covariates and their effect on survival.

4.6.2 Case study 2: What is the difference in survival between a selection of ER+ breast cancer patients who are aged between 20 and 40 years old compared with a similar group of patients between the ages of 40 and 60 years old?

To assess this group of patients, the two cohort breast cancer search tool was utilised. As shown in figure 4.17, a value towards 1 indicates survival. Interestingly, group 1 which represents those patients aged between 20 and 40 shows a lower survival rate than those older which is shown by group 2. The high *chi squared* value indicates a large difference between the two groups of patients.



Log Rank Test

Chi Square	df	P Value
177.876	1	0.00000

Figure 4.17 – Two cohort Kaplan-Meier plot between the two subsets of patients queried.

4.7 Using Superstes to determine survival for a given subset of colon cancer patients

4.7.1 Case study 1: A group of white, male patients who are married have all developed colon cancer which has been diagnosed as a type 2 tumour. They all range in age between 30 and 40 years old. What effect will this have on their survival?

To address this question, the single cohort colorectal search tool of Superstes was used. The results show that over 25 patients of the selected subset have a tumour located in the sigmoid colon and over 15 patients having a tumour located in the cecum. Finally the third largest group of 10 patients had a tumour located in the ascending colon as shown in figure 4.18.

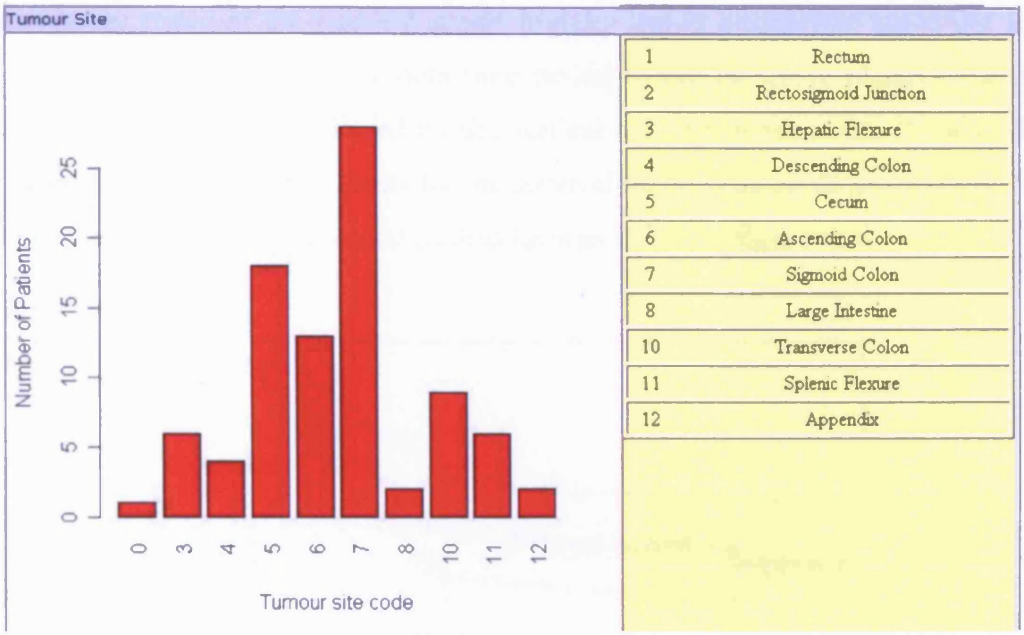


Figure 4.18 – The location of the tumour for those patients selected with colon cancer.

Over 15 patients of the selected subset had a tumour of 40mm in size as shown in figure 4.19. Tumours of approximately 30mm were found in 10 patients and also approximately 20 patients had a tumour size of 60mm.

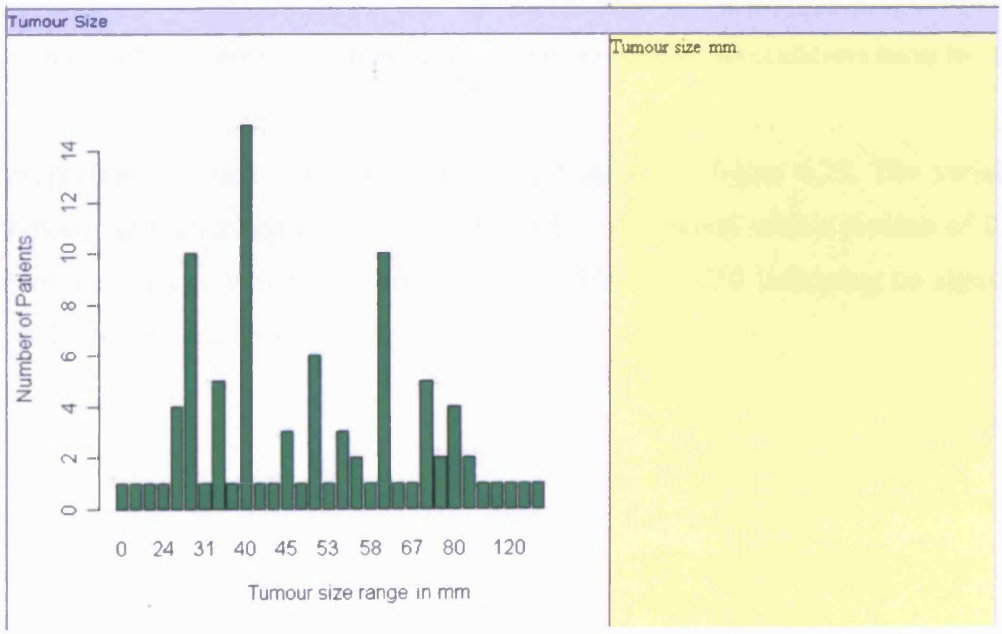


Figure 4.19 – Tumour size in mm of selected colon cancer subset.

The Kaplan-Meier curve of the selected subset initially shows a relatively sharp fall indicating patients who have died until the 50 month time period before the curve plateaus. At this point some patients are censored as indicated by the vertical red lines between the 70 and 150 month period. The point wise confidence limits for the survival function at the different points in time is also plotted as represented by horizontal dashed lines as shown in figure 4.20.

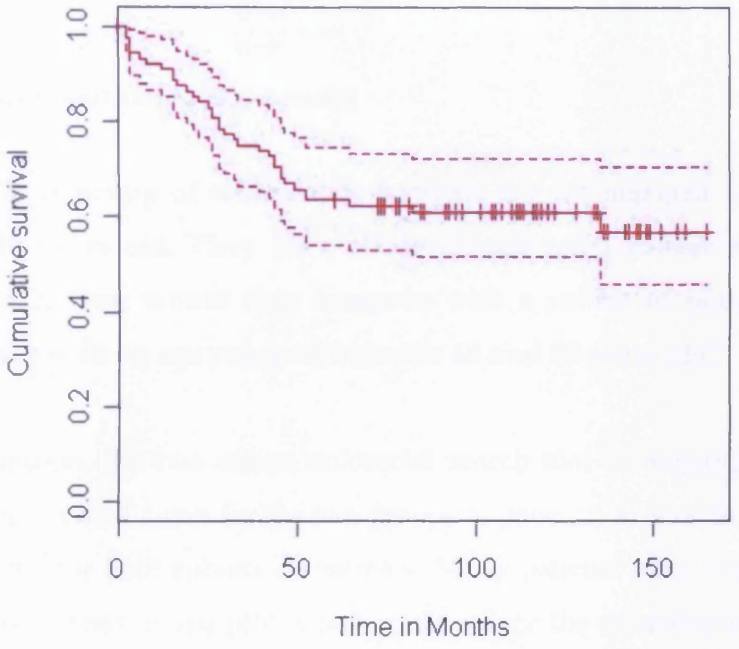


Figure 4.20 – Kaplan Meier plot of the selected colorectal subset shown with confidence limits for the survival function.

The cox proportional hazard model results are highlighted in figure 4.29. The variable nodes positive showed far higher significance as a predictor of survival with a p-value of 0.020 than either tumour size or age which had p-values of 0.930 and 0.750 indicating no significance, a summary of the results can be seen in figure 4.21.

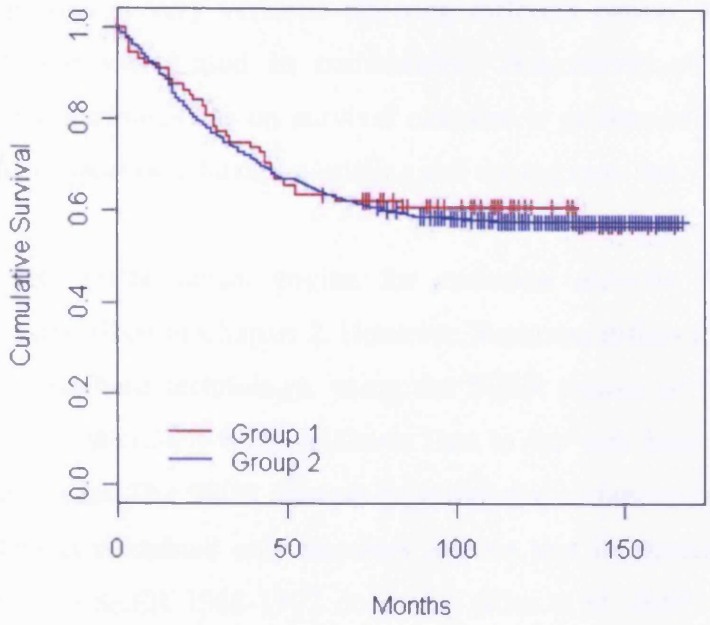
Model Summary								
Variable	coef	exp(coef)	se (coef)	z	p	exp(-coef)	lower .95	upper .95
nodes_positive	0.010	1.010	0.004	2.326	0.020	0.990	1.002	1.019
tumour_size	-0.001	0.999	0.007	-0.088	0.930	1.001	0.985	1.014
age	-0.023	0.978	0.072	-0.313	0.750	1.023	0.848	1.127

Model Statistics			
RSquare	0.054		
Max poss. RSquare	0.963		
Tests	Score	df	P
Likelihood ratio test	4.967	3	0.174
Wald test	6.090	3	0.107
Score (logrank) test	6.611	3	0.085

Figure 4.21 – Cox proportional hazard model results.

4.7.2 Case study 2: A group of white male patients are all married and range in age of between 30 and 40 years old. They have all developed colon cancer with tumours being classified as grade 2. How would they compare with a subset of patients with the same cancer attributes yet with an age range of between 40 and 50 years old?

To address this question, the two cohort colorectal search tool in Superstes can be used. The results show that the survival curve for the two groups of patients, as seen in figure 4.22, shows a similar survival curve for both cohorts of patients. Many patients were censored in group 2 as indicted by the vertical lines on the plot which could reduce the reliability of the curve. The log rank test p value of 0.755 reflects that there is not significant different between the two groups of patients.



Log Rank Test

Chi Square	df	P Value
0.098	1	0.75399

Figure 4.22 – Kaplan-Meier survival cure comparing two cohorts of patients between the ages of 30 and 40 (group 1) and an older group (group 2) between the ages of 50 and 60.

The four example patient cases outlined are a typical example of how the tool could be used. The next step would be to more carefully assess each patient variable in turn, comparing the results of the modelling after each search so determine the best combination of patient variables which is playing the most significant role in survival outcome.

4.8 Discussion

Superstes harnesses the power of web service technology in a user-friendly, online multifunction "query and report system" for correlating patient data with survival in breast and colorectal cancer. Importantly, the capabilities of Superstes, facilitate the exploration of different combinations of patient variables on survival with statistical analysis assessing the value and impact particular variables have on overall survival. Threfore, it plays a different role to other online analysis tools using the SEER patient dataset such as Adjuvant!Online and

CancerMath.net. Superstes is very versatile allowing different patient variables which could potentially be predictive when used in combination. Assessment of the value particular combinations of patient attributes has on survival outcome is performed through Kaplan-Meier survival analysis, Cox proportional hazard modeling and the log rank test.

Superstes uses the 'R' mathematical engine for statistical analysis which was also used extensively by I-10 as described in Chapter 2. However, Superstes differs from I-10 in that it uses multi-concurrent user database technology, using the SEER dataset of cancer patients which Superstes searches via a Microsoft SQL database. Due to the web based nature of Superstes, multi user ability is essential. The SEER data set for breast and colorectal cancer was transformed so that the stored dataset contained only numbers and no text information. Adherence to the standard set down by the SEER 1988-1992 document (Ries et al, 2005) [100] demonstrates a precedent to base non-standardised systems upon. This would enable subsequent pathological datasets from non-US sources to be added to the database and queried by Superstes. Providing coding such as ICD-10 does not change in the foreseeable future, Superstes should be able to process all such information accurately therefore making it an expandable and powerful resource.

The architecture of Superstes also differs from I-10 in its adoption of new web-based technologies. Thus, due to technical inheritance of certain elements of Visual Basic 6.0 used in I-10 development, it has been possible to use similar strategies in terms of interfacing 'R' with D-COM in Superstes to VB.net. Importantly, using the R-(D)-COM module to communicate with web-based visual basic.net technology was a very powerful association. This again allowed automated production of statistical analysis such as Kaplan-Meier survival curves. All analysis techniques are available at the click of a button through the web browser instead of a locally-installed application as in the case of I-10.

Microsoft visual basic.net technology has performed well for development of Superstes. The tools searches thousands of patient records in seconds and the Windows Server 2003 platform has provided a robust platform for hosting of Superstes. However the key advantage of Superstes being web based has been computer system independence and worldwide access. The web-based nature of Superstes should facilitate adoption by the greater research community of oncologists

using datasets at an international level. Clinicians can access Superstes via the internet using any web-capable operating system and any web browser such as Microsoft Internet Explorer or Mozilla Firefox.

Any change to Superstes as a result of updates or new datasets to analyse is instantly reflected in the online version as opposed to needing to distribute updated versions to users. Users can always be assured that the current version is available online without having to make any updates themselves on every machine where they use Superstes.

Future implementations of Superstes could replace the Microsoft SQL server database with a mySQL database implementation. MySQL is a license free, data base application which is well established. It is more akin to the community development spirit which 'R' brings to Superstes. A library exists in 'R' for communication with mySQL which could potentially pave the way for an alternative method of analysing the SEER dataset. This could have performance benefits from an analysis point of view especially if more datasets are added to Superstes.

4.8.1 Impact of Superstes on prognostic marker discovery in cancer research

A key strength of Superstes is the way in which users can obtain survival estimates based on actual data, rather than estimates generated by a regression analysis, for example. Superstes generates survival curves for all available data for any particular follow up period and as opposed to just a single time point. The ability to generate histograms, Kaplan Meier and Cox proportional hazard models offers users new ways of comparing and exploring biomedical data to reveal trends and display relationships within the data in a comprehensive, easy to use way, potentially resulting in the generation of new novel prognostic models. It was fortunate to have access to a high quality cancer data source such as the SEER dataset to facilitate Kaplan-Meier curves where the robustness and accuracy is critically dependent on the quality of the underlying data set (Kumar et al, 1994) [165].

Users of Superstes are not limited to a single analysis model yet can enter any prognostic factor combination they desire. Two cohorts of patients compared against each other in both breast and

colorectal cancer is a powerful ability. However due to the myriad of patient variables available, the more refined choices the user makes, the fewer patients which will match the selected categories and which could potentially result in uncertain survival estimates. However using combinations based on actual patient cohorts will select the most applicable variables initially, which will impact the number of matching patients and the confidence intervals generated. However this should not detract from the numerous combinations in which the current (and future) covariate patient parameters could be assessed and modelled and the SEER dataset resource itself explored. Superstes will prove a very useful tool.

It is also hoped Superstes will pave the way for more collaboration through showing how cancer patient datasets can be mined. The natural progression for the developed tool would be to encompass multiple types of cancer using similar data formatting of the patient variables as demonstrated by the SEER breast and colorectal datasets. Use of Superstes will add value to databases which currently cannot be queried in such a user friendly manner. The value of the information generated will hopefully demonstrate a precedent to encourage others to follow and want to add data to the system. Other types of cancer and covariate data available now or in the future from SEER could be added quickly.

The approach could also be powerful if applied to UK data sources to explore or compare treatment regimes form the UK versus the USA, for example. This has wide ranging implications of opening up a new resource to oncologists to not only check published results however allow for discovery of new covariates which will ultimately impact on driving research and discovery forward. Superstes strikes the balance between publication demands and the accessibility of data and the reluctance of many researchers to release their valuable datasets into the public domain. The interactive web based nature of Superstes facilitates explorative analyses of prognostic variables and could potentially be applied to a variety of diseases in addition to cancer.

4.8.2 An international cancer patient data resource

The successful web-based architecture of Superstes is a very powerful and versatile feature which should allow international exposure as well as allowing the capabilities of Superstes to be infinitely expandable allowing relationships to be explored in the dataset at an unprecedented level.

Usage of web service technology should allow interfacing with other systems to receive data for analysis or, as demonstrated here, to provide access to the SEER dataset. A fundamental goal of the project was to keep the tool universal yet maintain access to the data source so its value could be exploited by other applications. This has been achieved using well established SOAP web service technology. Applications which can retrieve and display returned results from a web service can access the data source.

The planned exposure that Superstes will experience in the future will be interesting to see how researchers and oncologists accept the tool and find it useful to explore new combinations of patient variables which could ultimately lead to new prognostic models. However one of the most powerful benefits of Superstes which should not be underestimated is how adoption of standardisation through programs such as SEER results in the generation of invaluable data mining tools.

Chapter 5

Assessment of survival using machine learning algorithms based upon the Nottingham Prognostic Index (NPI) covariates using the SEER dataset, R and Weka.

Chapter 5 – Assessment of survival using machine learning algorithms based upon the Nottingham Prognostic Index (NPI) covariates using the SEER dataset, R and Weka.

5.1 Background

Predicting survival of a patient who has been diagnosed with cancer is difficult. No two patients are a like – they will all differ in some way. Analysing past cases of patients who have developed cancer and then survived for differing periods could potentially uncover novel relationships. Understanding the connection between what specific attributes a patient exhibits and how long they live as a result of those attributes will help understand the mechanisms of cancer related deaths. The gap which connects a set of patient attributes and their survival can be thought of as a 'black box'. We strive to understand what happens in the 'black box' which could be thought of as nature itself – the processes which take place within the patient which ultimately govern their survival. From a statistical view point, it is valuable to determine what set of mechanisms connects a certain set of patient attributes with their resulting survival.

Prognostic indices, such as the St. Gallen criteria, the TNM system and the Nottingham Prognostic Index, can integrate information from several prognostic factors which have been validated in a number of ways and assigned to patients of different prognostic categories (Ellis et al, 2004) [89]. However, what these prognostic models do not provide are estimates for a survival per individual patient probability. The demand remains for tools that not only provide prognostic classification, but also give quantitative probabilities of survival (Fulford et al, 2007) [88]. It is important to recognise that any model intended to provide prognostic assessments should include NPI factors, as a minimum, largely due to the historic value systems such as the NPI has provided. One study showed that apart from therapy-specific models, molecular and other prognostic classifiers would have to add significant information to the NPI to be considered clinically important (Fulford et al, 2007) [88]. Consequently, it could prove difficult to find new prognostic markers from those patients who fall into the excellent prognosis group (EPG) of the NPI.

The limitations in the abilities of the NPI has been well documented (Yu et al, 2004) [8]. For example, while widely applied to inform the choice of adjuvant therapy, advances in therapy and detection policies, such as the introduction of breast cancer screening for women aged 50 (Fulford et al, 2007) [88]. Consequently, there is a continued need to explore covariate combinations, drawing on large clinical datasets such as those for breast cancer, to test new predictive models.

However it should be clear that indexes such as the NPI are forms of classifying a patient dataset. Classification as a process is arguably one of the most important analytical tasks in high throughput data analysis (Handl et al, 2005) [106]. Numerous automated classification procedures have been developed to try to map new underlying patterns in datasets and explore the 'black box' which is a cancer patient with the goal of finding important correlations between different patient variables thought to predict an outcome, such as survival.

There is lack of comparison between analysis methodologies to find a better framework for classification of patient covariate information.

It has been shown in previous chapters that significant biological relationships can be identified using high throughput analysis technologies which datamine using statistical and machine learning methodologies using *in vitro* and *in vivo* data sets. Previous chapters have focused on how biological function generated from Affymetrix array data and cancer patient dataset, such as the SEER data set, have the potential to reveal possible markers of endocrine response/failure and also clinical prognostic markers through use of I-10 or Superstes respectively. Using the correct tools, a cancer researcher can begin to determine the more robust individual associations within such biomedical data in relation to these clinically-important end-points using statistical procedures available through software, where this approach has been demonstrated in previous chapters.

5.2 Data Mining

Data mining is a knowledge discovery process. It provides structure for discovering and importantly – allows quantification of patterns hidden in large quantities of data (Cox, 2005) [116]. Patterns, in the form of classification models, can be discovered in many ways ranging from more traditional statistical methods through to more advanced machine learning approaches.

Linear regression analysis was one of the earliest forms of data mining created by Johann Gauss in 1809 due to his work on the "method of least squares" (Buhler et al, 1981) [117]. Regression analysis allows the modelling of dependent variables such as the response variable and one or more independent variables. The dependent variable is modelled as a function of the independent variable and any constants required. The best fit method is evaluated by the "method of least squares". Consequently, regression analysis can be used for prediction and modelling of events thought to cause another event.

5.2.1 Logistic regression

Logistic Regression is a type of predictive model that can be used when the target variable, for example survival, is a categorical variable with two categories – such as live/die. A logistic regression model is similar to non linear regression such as fitting a polynomial to a set of data values.

Logistic regression can be used only with two types of target variables. These are a categorical target variable that has exactly two categories (i.e., a binary which could represent alive (1) or dead (0)) or a continuous target variable that has values in the range 0.0 to 1.0 representing probability values or proportions.

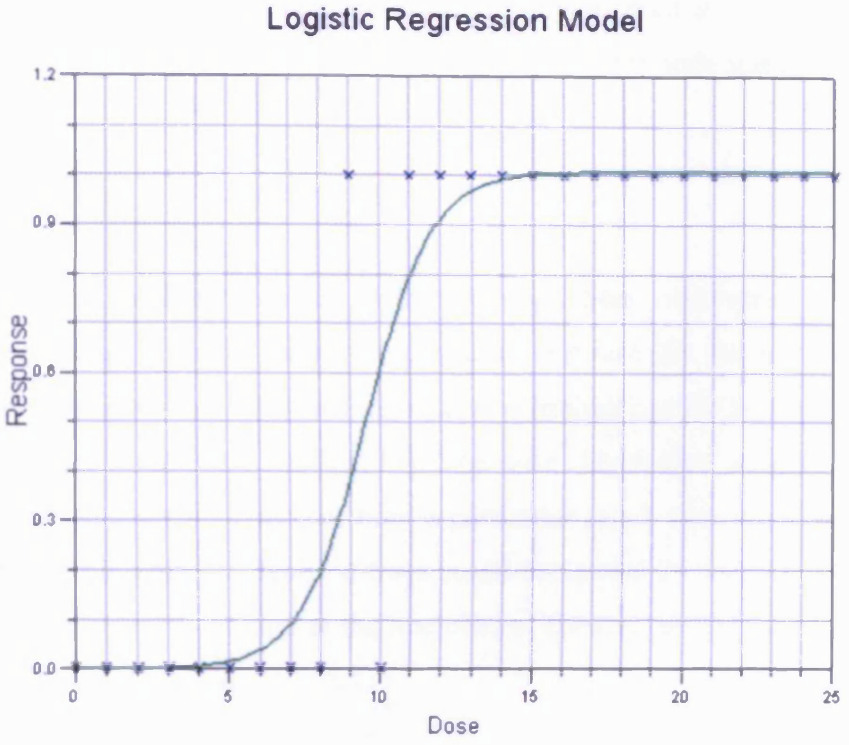


Figure 5.1 – Dose response curve – an example of a logistic regression model (Bewick et al, 2005) [118]

Logistic regression has two key advantages over linear regression (Bewick et al, 2005) [118]. Firstly, there are no limits on the values predicted by a linear regression, so the predicted response might be less than 0 or greater than 1 – which is not therefore appropriate if predicting survival.

Secondly, the response usually is not a linear function of the covariates predicting the outcome. Again a clinical trial of a drug analogy can be used to highlight the affect of a covariate on an outcome. If a small amount of a drug is given, it is likely few patients will respond. Doubling the dose to a larger amount will mostly probably not yield any positive response. However as the dosage increases beyond a certain threshold, there will become a point where the drug starts to become effective. Small increases in the dosage above a threshold may produce an increasingly positive result however, eventually a saturation level is reached which beyond, and therefore eventually increasing the dosage will not increase the response (Bewick et al, 2005) [118]. Figure 5.1 illustrates this effect which is also known as a dose response curve.

Consequently, linear regression could be used initially to predict survival outcome based on patient variables such as tumour grade, tumour size and tumour node status.

5.2.2 Decision trees

A decision tree is a predictive model in that it maps from observations about a variable to conclusions regarding its target value, in this case survival. In the tree structures, each leaf represents classifications and branches represent combinations of features that lead to those classifications (Jonsdottira et al, 2008) [119]. The term 'black box' is often applied to machine learning methods whereby it is unclear how a particular result was formed, based on the given input. To continue this analogy, decision trees could be termed as 'white boxes' as a given result is provided by the model by looking at the branches of the tree with the explanation for the result often replicated by simple math.

Decision trees can be applied to biomedical data as the concept of a decision tree is analogous to the procedure used by a clinician who will ask a patient a series of questions (or in this instance, query aspects of the clinicopathological data) until arriving at a diagnosis or prediction of prognosis, for example (Jonsdottira et al, 2008) [119]. There are a wide range of decision tree algorithms that can predict both binary and more complex multiple outcomes. One of the first was the ID3 (Iterative Dichotomiser 3). An improved version of the ID3 algorithm is the J48 decision tree (Jonsdottira et al, 2008) [119]. Key improvements include the ability of the user to choose an appropriate attribute selection measure, able to cope with training data with missing attribute values, able to cope with attributes having different costs, and the ability to cope with continuous attributes (Jonsdottira et al, 2008) [119].

An example of a basic decision tree is shown in figure 5.2. The reader starts at the left of the tree and works through the tree by answering yes or no to a particular patient variable, branching their way until arriving at a leaf with an affirmative answer (Jonsdottira et al, 2008) [119]. When a tree model is built from classifiers it can then be applied to new cases to predict their outcome. Like any model, a decision tree will predict an outcome based on the values of the input attributes.

Although possible manually on paper for simple problems, for biomedical datasets, predictions are made algorithmically due to the potential covariate complexity within such a dataset.

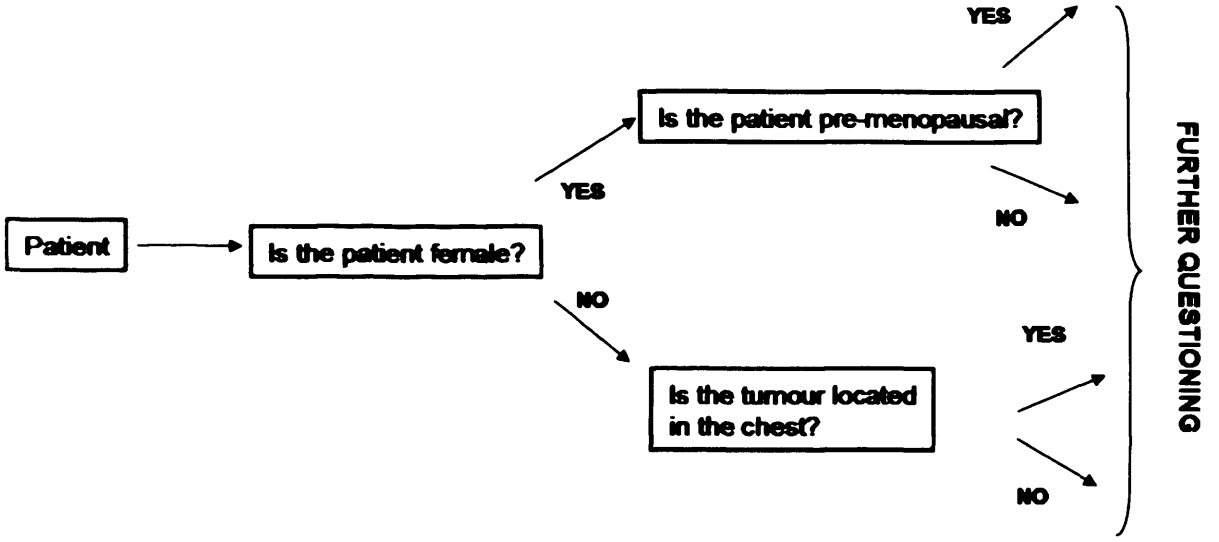


Figure 5.2: A basic decision tree. In this example, the tree determines the type of breast cancer the patient has. An important early question is gender as this has an impact on the subsequent questioning and hence the route through the tree will differ.

5.2.3 Support Vector Machine (SVM)

A Support Vector Machine (SVM) performs classification by formation of an N-dimensional hyperplane that optimally separates the data into two categories (Moguerza et al, 2006) [120]. There are different types of support vector machines however all are supervised learning methods used for classification and regression analysis. The simplest type of support vector machines is linear classification which tries to draw a straight line that separates data with two dimensions as shown in Figure 5.2. A linear classifier is also known as a 'hyperplane'. A predictor variable is called an attribute, and a transformed attribute that is used to define the hyperplane is called a feature. The task of choosing the most suitable representation is known as feature selection. A set of features that describes one case (i.e., a row of predictor values) is called a vector. So the goal of SVM modelling is to find the optimal hyperplane which separates clusters of a vector in such a way that cases with one category of the target variable are on one side of the plane and cases with

the other category are on the other size of the plane, again as illustrated in Figure 5.3 (Moguerza et al, 2006) [120]. The vectors near the hyperplane are the support vectors.

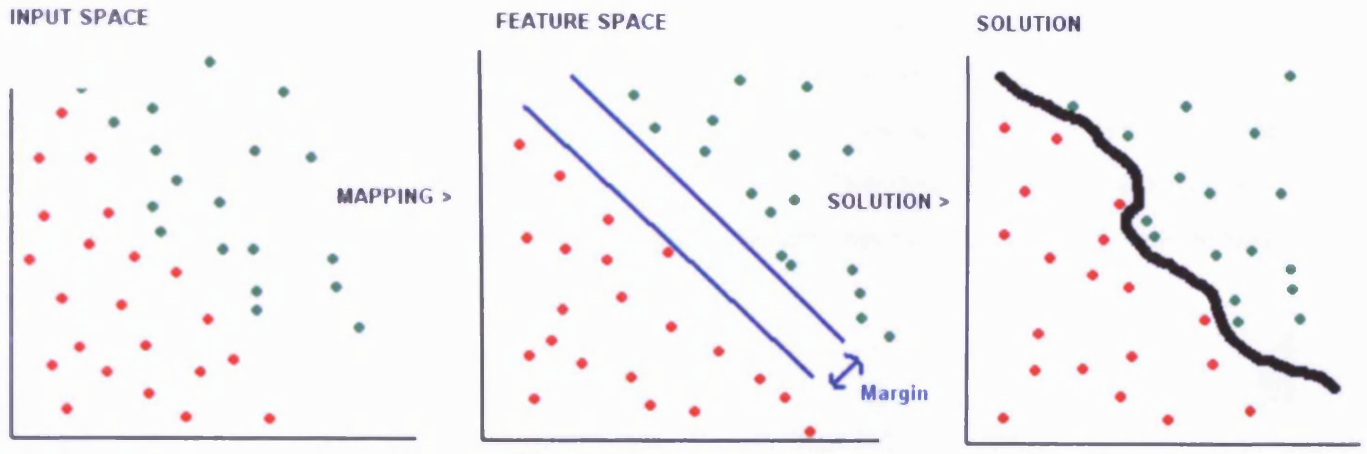


Figure 5.3 – An overview of the Support Vector Machine process illustrating the process of a linear boundary being drawn between instances of different classes – shown as red and green dots. The solution of separation is shown as a black line drawn between the red and green dots. (Moguerza et al, 2006) [120]

5.2.4 Boosting and AdaBoost

Many of the machine learning techniques previously outlined require optimisation for best results particularly in the case of Support Vector Machine. However methods exist whereby the algorithm can learn from previous analysis cycles. Boosting is such a technique. Boosting builds a series of models all of the same type such as in a decision tree format whereby new models are affected by previous model performance. The process occurs many times with the goal to improve upon the inaccuracies of models built during previous cycles. Bagging is thought to reduce variance (Yang et al, 2007) [121]. It can helps improve unstable classifiers resulting from "small" changes in training data leading to significantly different classifiers and "large" changes in accuracy, for example. Boosting brings two modifications to the learning process. Instead of a random sample of the training data, a weighted sample is instead used to focus learning on the most difficult examples. Also instead of combining classifiers with an equal vote, a weighted vote is used.

The machine learning algorithm AdaBoost, short for Adaptive Boosting, was produced by Yoav Freund and Robert Schapire (Yang et al, 2007) [121]. It is a heuristic method for solving a general class of computational problems by combining user-given so called 'black-box' procedures as previously introduced, and can be used in conjunction with many other learning algorithms to improve their performance. As in the case of boosting in general, Adaboost is adaptive in that subsequent classifiers built are altered in favour of those instances misclassified by previous classifiers (Yang et al, 2007) [121]. AdaBoost is particularly sensitive to 'noisy' data and observations which are numerically distant from the rest of the data. Cross validation applied to Adaboost results reduces the over fitting scenario where the results indicate when further training is not resulting in better generalisation.

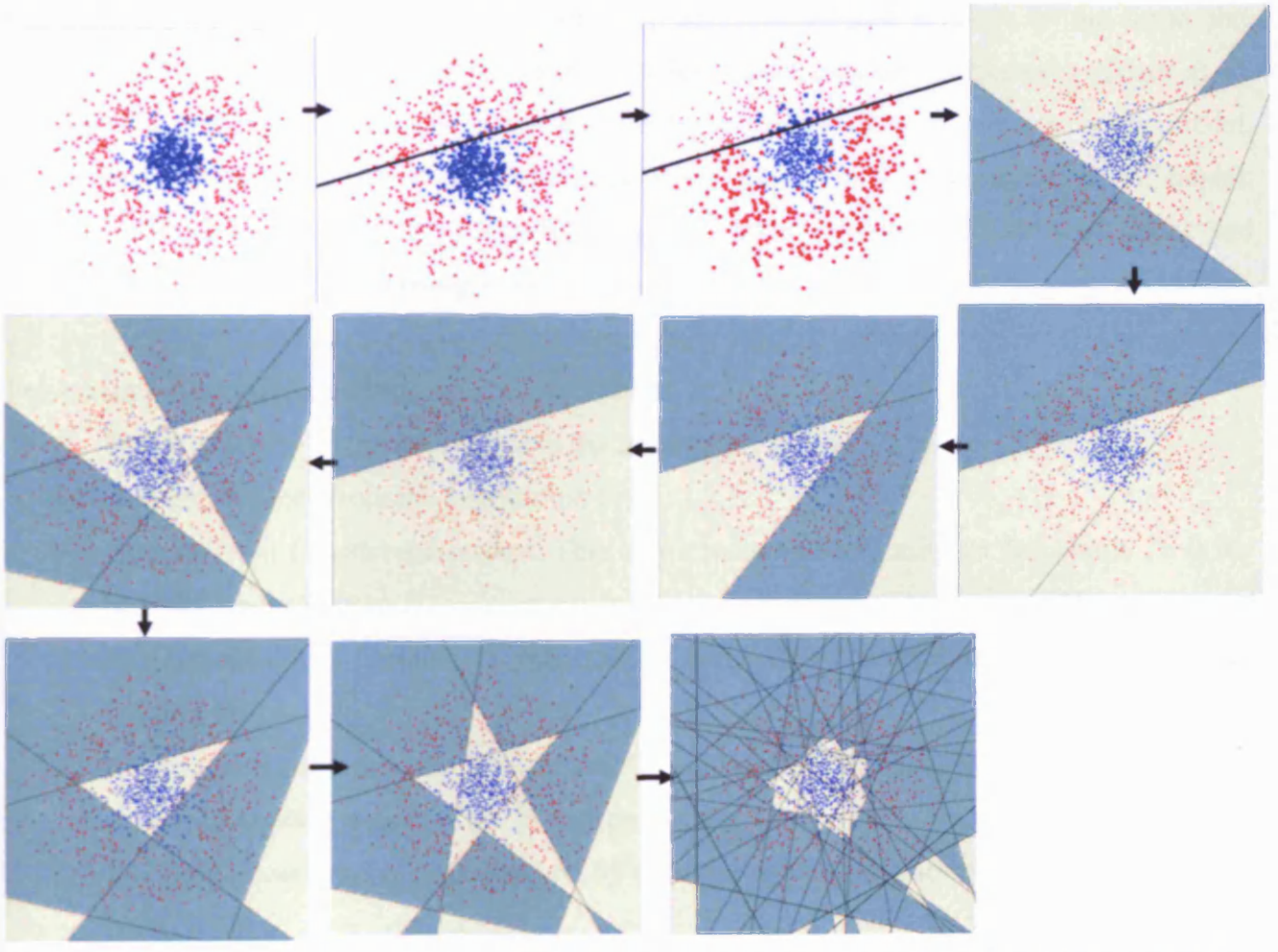


Figure 5.4: Illustration of the operation of the Adaboost algorithm to classifiy red and blue dots where successive iterations improve classification on each cycle of the algorithm based on the previous result. (Yang et al, 2007) [121]

AdaBoost calls a weak classifier repeatedly in a series of cycles as illustrated in figure 5.4. For each call a distribution of weights is updated that indicates the importance of examples in the data set for the classification (Yang et al, 2007) [121]. On each round, the weights of each incorrectly classified example are increased (or alternatively, the weights of each correctly classified example are decreased), so that the new classifier focuses more on those examples.

5.2.5 Bagging

Bagging – a short term for boot strap aggregating is a meta-algorithm to improve machine learning of classification and regression models in terms of stability and classification accuracy (Islam et al, 2008) [122]. The model works by applying random datasets of the same size generated from a training dataset by sampling with sequential case replacement (Islam et al, 2008) [122]. Bagging also reduces variance and helps to avoid over fitting of a particular model. It is usually applied to decision tree models, however can be applied to other types of model. Bagging differs from boosting mainly by building models independently of each other and offering no weights to models using a voting procedure.

For example, consider a training set D with m cases.

1. The probability of the nth sample in the training set as $P(n) = 1/m$.
2. Sample m times from the distribution $P(n)$
3. Sample from D with replacement. This way a re-sampled training set Di is built. Di is the bootstrap sample from D
4. The procedure is repeated to construct a sequence of several independent bootstrap training sets
5. A corresponding sequence of classifiers is constructed by using the same classification algorithm applied to each of the bootstrap training sets
6. The final classification is determined by each classifier voting for each class (Islam et al, 2008) [122].

degeneration in accuracy. An important feature is that it carries along an internal test set estimate of the prediction error. For every tree grown, about one-third of the cases are out of the boot strap sample.

The design of random forests is to give the user a good deal of information about the data besides an accurate prediction. Consequently, it could potentially prove to be a valuable model in exploring the 'black box' phenomenon of survival.

5.2.7 The Naive Bayes classifier

A Naive Bayes classifier assumes that the presence (or lack of presence) of a particular feature of a class is unrelated to the presence (or lack of presence) of any other feature (Chun et al, 2007) [124]. A bayesian network represents independencies over a set of variables in a given joint probability distribution. Nodes correspond to variables of interest, and arcs between two nodes represent statistical dependence between variables. Bayesian refers to Bayes' theorem on conditional probability (Chun et al, 2007) [124]. Bayes' theorem is a result in probability theory, which relates the conditional and marginal probability distributions of random variables. The probability of an event X conditional on another event Y is in general different from the probability of X conditional on Y. However, there is an explicit relationship between the two, and Bayes' theorem is the statement of that relationship (Chun et al, 2007) [124].

Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; one can work with the naive Bayes model without using any Bayesian methods (Chun et al, 2007) [124].

However even though their design can be classed as naive and use over-simplified assumptions, naive Bayes classifiers work better in many real-world situations than expected. An advantage of the naive Bayes classifier is that it only requires a small amount of training data to estimate the variables necessary for classification. As a result of independent variables being assumed, only the variances of the covariates for each class need to be determined and not the entire covariance

Bagging should only be used if the learning machine is unstable. A learning machine is said to be unstable if a small change in the training set yields large variations in the classification (Islam et al, 2008) [122].

5.2.6 Random Forest

A 'random forest' is a classifier that consists of many decision trees and outputs the class that is the mode of the classes output by individual trees (Statnikov et al, 2008) [123]. The algorithm was developed by Adele Cutler and Leo Breiman. Breiman was also responsible for the Bagging concept previously outlined. The term came from random decision forests that were first proposed by Tin Kam Ho from Bell Laboratories which later became part of AT&T in 1995. Interestingly, the same lab was responsible for development of the language 'S' plus which the statistical scripting language 'R' was based. The random forest method combines "bagging" and "random subspace method", the latter which was developed by Tin Kam Ho. This ultimately constructs a collection of decision trees with controlled variations.

There are many advantages of the random forest method. From a prediction of survival point of view, the method is known to produce highly accurate classifiers from a large number of input variables. It is also able to estimate the importance of variables in determining classification. As the 'forest' is built, it generates an internal unbiased estimate of error (Statnikov et al, 2008) [123]. The system can also balance error in class population unbalanced data sets which would be valuable for modelling survival from cancer patient datasets where there is a bias towards death at some point in time.

The key to accuracy is low correlation and bias. To keep bias low, trees are grown to maximum depth. To keep correlation low, the current version of random forest uses randomisation. Each tree is grown on a bootstrap sample of the training set. A number m is specified much smaller than the total number of variables M. At each node, m variables are selected at random out of the M, and the split is the best split on these m variables. In empirical tests, RF has proven to have low prediction error. On a variety of data sets, has been proven to be more accurate than Adaboost previously outlined. It handles hundreds and thousands of input variables with no

matrix (Chun et al, 2007) [124]. Again, this method could prove very effective in predicting the probability of survival based on certain given patient variables.

5.2.8 Supervised learning – classification to model survival

Unlike unsupervised learning, the objective of supervised classification models is error minimization. Thus, a natural cost function, the number of misclassifications, exists. Nevertheless, several metric-based classification models do not explicitly optimise this cost function, however they are based on intuitive heuristics.

Supervised learning is a broad term which can use any of the machine learning methods previously outlined. It can be applied to many situations where an example is illustrated multiple times based on a set of input criteria and the resulting outcome as a result of the input produced. The system is shown an example of what the result needs to be and what input values were given to produce that output. Ultimately when the system is given only input values, it can make the right decisions based on 'training' on how they can be combined to produce the correct output.

To achieve this, a training set is first required. The training set needs to be characteristic of the real-world use of the function. In this chapter, the outcome survival can serve as output and the input parameters correspond to patient variables such as tumour size, tumour node status and tumour grade.

The accuracy of the learned function depends strongly on how the input covariates are represented. Typically, the input object is transformed into a feature vector, which contains a number of features that are descriptive of the object. The number of features should not be too large, because of the curse of dimensionality; yet should be large enough to accurately predict the output. Consequently there could be an underlying relationship between tumour node status, tumour grade and tumour size where a relationship is already known to exist due to the Nottingham prognostic index using this combination of features. Therefore overall there are a myriad of algorithms as previously outlined which could fulfil this aim.

There is no single classifier that works best on all given problems. Once a particular learning algorithm is chosen it is applied to the training set. Parameters of the learning algorithm are then adjusted by optimising performance on a subset of the training set using cross-validation. After parameter adjustment and learning, the performance of the algorithm is measured on a test set that is separate from the training set.

Some of the classifiers previously outlined will perform better than others. The underlying data set on which the classifier is applied is very important for successful results, with a typical example dataset being a selection of patients corresponding to 10 year survival of the SEER breast cancer dataset, for example.

There are situations in survival analysis where the prediction of whether the event will eventually occur or not is of primary importance however there are also cases where modelling survival is an attempt to determine the probability of the event to occur within a specific time (Modlich et al, 2006) [12]. For example, for an oncologist to decide whether to operate on a patient with clinically localized breast cancer, the probability of cancer recurrence would be a very important decision factor (Modlich et al, 2006) [12].

Consequently it would be interesting to determine if methods outlined such as Decision Trees, Bagging, Random Forest can be used to robustly predict survival outcome (dead or alive) or survival probability using the Bayes classifier. Ten fold cross validation results would therefore indicate the best methodology as it would assess how effective one model is in comparison to another.

5.2.9 Measuring the error of a particular classifier – cross validation

In order to perform a measure of classification error, it is necessary to have test data samples independent of the learning dataset that was used to build a classifier. However, it is undesirable to hold back data from the learning dataset to use for a separate test as this could potentially weaken the learning dataset. However the classifier still needs to be tested against a dataset which

has not been used to train the classifier. A cross validation technique performs independent tests without requiring separate test datasets and without reducing the data used to build the classifier. This is achieved by the learning dataset being partitioned into a particular number of groups called "folds" (Chun et al, 2007) [124]. The number of groups that the rows are partitioned into is usually 10 as indicated in literature (Chun et al, 2007) [124]. This is particularly important to apply to assess a models performance in predicting survival.

5.2.10 Assessing an accurately predicted outcome - the confusion matrix and kappa statistic

When comparing different machine learning solutions it is very important to pay attention to the learning performance of each technique. It is important to track the 'cost' of a wrong decisions made by a particular method otherwise errors may lead to inaccurate results when applied to other datasets. A confusion matrix facilitates this 'cost' to be measured. For survival prediction, measuring this 'cost' is very important so that patients who may have survived are not incorrectly classified as having died.

The kappa statistic is a measure of agreement between predicted and observed classifications however is does not take the 'cost' into consideration of each algorithm. This gives a measure of the performance of the learning aspect of the algorithm. The kappa statistic is essentially a representation of the performance of a particular classification algorithm on overall results.

A summary of the accuracy of a particular method after 10 fold cross validation of results for different machine learning algorithms is displayed using a confusion matrix. Results are output in binary – 0 or 1. In a survival or death situation, 0 represents survival and 1 represents death. Table 5.1 summarises the structure of a confusion matrix.

		The predicted result	
		0 (Alive)	1 (Dead)
The actual result	0 (Alive)	True positive (TP)	False positive (FP)
	1 (Dead)	False positive (FP)	True negative (TN)

Table 5.1 – The format of a confusion matrix as used to display the validation of a classifier

It would be hoped that any given classifier would have the majority of patients in the true positive and true negatives column. This represents an accurate result of a classifier.

5.3 Using 'R' and Machine Learning Algorithms through 'Weka'

The power of 'R' has been demonstrated previously in Chapters 2, 3 and 4 in the context of its capability to analyse Microarray data as well as present the impact of clinical variables on survival outcome using Superstes. 'R' is used again in this Chapter to enable advanced multivariate statistical and machine-learning analysis procedures to be applied to a clinical breast cancer dataset, however in this instance interacting with a powerful data mining engine called 'Weka' written in Java (Witten et al, 2005) [125].

The Weka project, described as a set of machine-learning tools, was developed at the University of Waikato. Development began in 1993 (Witten et al, 2005) [125] and it was launched for the Java platform in 1997. There have been over a million downloads of the application to date from the University web site with the application reaching distribution version 3.4. There are many methods available in Weka to perform advanced regression, classification and clustering analysis as previously described.

To build models which predict outcome of breast cancer in Weka, variables from the patient dataset which are prognostic needed to first be identified and the data initially prepared. As in the case of the NPI, the number of nodes positive, tumour size and grade have been examined. However, methods ultimately applied from Weka have been selected so that models can deal with mixed sets of continuous and categorical predictor variables.

An 'R' library called 'RWeka' has been specifically developed by the 'R' project community by the Java application creators from the University of Auckland, New Zealand (Witten et al, 2005) [125]. This allows Weka to interface with 'R', resulting in an ability to offer powerful Weka machine-learning functionality using 'R' datasets. In addition to the machine learning algorithms

already offered in 'R' as libraries, further algorithms can be imported into 'R' via a Java interface to Weka (Witten et al, 2005) [125].

Weka and 'R' can be used to generate models using the same panel of parameters as used in the NPI to predict survival, however it also facilitates exploration to determine if there are better patient variable combinations. Validated models could reveal which combinations of tumour size, tumour node status and tumour grade are the most accurate for predicting survival of a given cohort of breast cancer patients. Ultimately, successful models generated could have the potential to influence management and thereby improve outcome of breast cancer patients based on more accurately predicting survival.

5.4 Exploring the SEER dataset using the Nottingham Prognostic Index

An initial key aim of the chapter was to validate the applicability of the SEER dataset to be used to model patient survival and also predict patient outcome.

The Nottingham Prognostic Index (NPI) is a classic example of how a simple formula and three covariates based on inherent disease characteristics– tumour size, grade and nodal status – can be used to predict patient survival, and thereby can help the clinician decide if treatment should be considered for a given patient. The scoring system was recently updated in 2007 by the original team who developed the system headed by Roger Blamey (Blamey et al, 2007) [111]. Instead of three prognostic groups there are now six prognostic groups for the NPI. Table 5.2 outlines the ranges of NPI score which correspond to each prognostic group.

Given the exploration of the SEER dataset through Superstes in Chapter 4, it is of interest to firstly assess the performance of the NPI against a non-uk dataset such as the SEER dataset. Close association of an NPI determined for a US dataset versus a UK dataset will therefore result in confidence in modelling alternative prognostic formula using the SEER dataset.

Prognostic group (PG)	NPI score range
Excellent (EPG)	2.08-2.4
Good (GPG)	2.42≤3.4
Moderate I (MPG I)	3.42≤4.4
Moderate II (MPG II)	4.42≤5.4
Poor (PPG)	5.42≤6.4
Very poor (VPG)	≥6.5

Table 5.2: Table of thresholds of NPI score with corresponding prognostic group for a given patients (Blamey et al. 2007) [111].

Before the NPI can be applied to the SEER dataset, the coding system for tumour size, grade and nodes positive status needed to be altered to mirror that which was used for the NPI. Tumour size on which the NPI was originally developed used centimetres whereas the SEER dataset uses millimetres. Grade also differs in that a UK dataset only contains Grade from level I through to III whereas the US system also contains levels IV and V. As grade IV and V represents tumours which are poorly differentiated, they can be merged into grade III to allow NPI calculation. A similar situation existed with nodes positive status where only values of 1, 2 or 3 can be accepted.

'R' can be used to calculate NPI scores for each patient in a 10 year survival subset of the SEER database. Appendix 4 shows an outline of steps taken to not only filter the dataset to bring it into line with what is expected for the NPI calculation however also to determine what proportion of patients survive within each group of the NPI.

Firstly, the number of patients whom survived 10 years and those who died is determined based on their NPI status. An overview of these results can be seen in table 5.3

Prognostic group	Number of patients who are estimated to survive 10 years	Number of patients who are estimated to not survive 10 years
EPG	6548	149
GPG	15573	792
MPG I	15395	1792
MPG II	9266	2431
PPG	4220	2512
VPG	1820	2120

Table 5.3: Proportion of SEER breast cancer patients according to their calculated NPI score and resulting prognostic group banding whom will either survive or die after 10 years.

The update to the NPI scoring system in a publication by Blamey *et al* (Blamey et al, 2007) [111] also once again used a Nottingham UK based dataset akin to the original work on the NPI in the 1980's. It is interesting to overlay the two sets of results from the SEER dataset in comparison to the UK based dataset. For the comparison, the percentage of patients that survive in each NPI group was calculated. Comparisons of the results are shown in table 5.4:

Prognostic group	SEER based NPI score patients	Nottingham UK NPI score patients
EPG	97.7	96
VPG	95.1	93
MPG I	89.6	81
MPG II	79.2	74
PPG	62.3	50
VPG	46.2	38

Table 5.4 – Comparison of the results between the NPI scored calculated for an American breast cancer dataset compared to a UK dataset.

Interestingly, the survival pattern between the two datasets is similar with better survival rates in the US based dataset compared to the UK dataset. This could be due to improved treatment and disease awareness seen in the American dataset being a more recent set of patients than that of the UK dataset. However it is also confirmation that the SEER dataset is a valid dataset upon

which improvements in the relationship between patient covariates and survival can be based to potentially improve upon the Nottingham prognostic index.

5.5 Aims of the Chapter

The analysis tools 'R' and Weka facilitate the exploration of a 10 year survival subset of the SEER breast cancer patient dataset. As a result the following aim was proposed:

1 Apply advanced predictive modelling techniques utilising multivariate statistical and machine learning procedures to generate quantitative probabilities of survival in clinical breast cancer, using an expanded set of variables based on the Nottingham prognostic index variables using the SEER clinical breast cancer data set.

2 Assess the validity of each method in terms of predicting survival and understanding the affect each covariate has on a particular outcome.

5.6 Strategy

To explore the patient data covariates from the SEER dataset of tumour grade, tumour size and tumour nodes postivity in relation to 10yr survival, the following analysis strategy was adopted using R and Weka with the goal of testing the ability of different classifier methods to improve predictive accuracy at each stage:

I. Perform logistic regression analysis and Evaluate the model built based on multiple logistic regression findings.
II. Produce a J48 decision tree to evaluate if this enables further insight into patient covariate combinations in predicting survival
III. Application of Support Vector machine to evaluate accuracy of survival status.
IV. Evaluate ability of classifier 'adaboost' to improve accuracy of the model, and validate using 10-fold cross validation using a small training set.
V. Evaluate ability of "Bagging" to further improve classifier accuracy and again produce a J48 tree.

VI. Evaluate Random Forest and regression analysis trees, to compare the accuracy of models built using these further methods.

VII. Application of the naïve Bayes classifier to predict the probability of survival as opposed to survival outcome as shown in previous methods.

The steps outlined are all performed through 'R' libraries and where a particular methodology does not exist in 'R' it is imported from Weka. Weka if they are not already scripted as part of the 'R' Weka library.

5.6.1 Predicting patient survival using different statistical and machine learning methodologies

I – Multiple logistic regression (exploring the NPI classifiers grade, size and nodal postivity in relation to 10 year survival).

To initially specify the NPI model formula in R and call the relevant covariate data to be used for testing the Weka methodologies, a special syntax must be used. The syntax used to build the expression in 'R' encompassing the potential prognostic predictor and survival status variables using the 10 year survival dataset, is shown in the code 5.1:

```
alivestatus ~ size + grade + nodespos, data = data.10yr
```

Code 5.1: The syntax for recreating the NPI equation in 'R'.

The survival variable ("alivestatus") is indicated by the tilde '~' symbol with exploratory predictor variables then added using the + symbol.

As introduced earlier, there are certain assumptions for linear regression that fail when the response variable (i.e. prognosis) is not continuous and if the response variable in the dataset is binary then consideration is needed towards generalized linear models. Logistic regression was performed using 'R' however it could also have been performed in Weka for examination of the NPI parameters in relation to 10yr survival.

Initially a working directory chosen in 'R' and the breast cancer 10 year dataset read into 'R' from SEER as shown in code 5.2.

```
>setwd("C:/TEMP")

>data.10yr <- read.table("data.10yr", header=T)
```

Code 5.2: Reading the 10 year filtered breast cancer dataset from the SEER programme into 'R'

The length command can be used to determine the alive status factor - the number of patients alive after 10 years, as shown in code 5.3.

```
> length(data.10yr$alivestatus)
[1] 15194
```

Code 5.3: Checking the number of patients in the dataset – in this case 15,194.

Therefore taking the whole dataset in account, the number of patients alive after 10 years (1) in the whole dataset was three times the number of patients that have died (0).

To create a logistic regression model, the glm() function is called in R which is used to set the family parameter to binomial and the data parameter is set to "data.10yr" (see code section 5.4). Using the summary() function, those test covariates related to the model (including the associated coefficients and significance level) can be retrieved, again as shown in code 5.4. The method glm() can provide a model that contains covariates able to predict whether a patient will survive 10 years or not based, in this instance, on the values of the covariates tumour size, tumour nodes positive and tumour grade. The parameters for this equation and corresponding statistics are reported in a matrix under the 'Coefficients:' list. The first column lists the predictor variables (size, nodespositive, grade) where factors are split according to level. The second column provides an estimate of the coefficients for the equation. The remaining columns provide the standard error for the coefficients, z statistic values (the coefficient estimate divided by the standard error) and p values highlighting whether the association between survival status and predictor variable is significant. All predictor variables tested here (grade, size and nodes positive status) proved to be reliable predictors of outcome (in accordance with NPI), each being highly

significant in the matrix. The 'Deviance Residuals' list summarises the range of differences in value between predicted and actual outcome values.

```
> lr.glm <- glm(alivestatus ~ size + nodespos + grade, data = data.10yr,
family="binomial")

> summary(lr.glm)

Call:
glm(formula = alivestatus ~ size + nodespos + grade, family = "binomial",
    data = data.10yr)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.0997  -0.6647  -0.5249  -0.3703   2.3856

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.325886   0.083432  -39.86   <2e-16 ***
size         0.027995   0.001373   20.39   <2e-16 ***
nodespos     0.136920   0.005646   24.25   <2e-16 ***
grade        0.456159   0.029992   15.21   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Dispersion parameter for binomial family taken to be 1)
    Null deviance: 16342  on 15193  degrees of freedom
Residual deviance: 14089  on 15190  degrees of freedom
AIC: 14097
Number of Fisher Scoring iterations: 4
```

Code 5.4 showing initial logistic regression step.

Evaluating the resultant Logistic Regression Model

To test the predictive capability of the model summarised in figure 5.11, (encompassing grade, size and nodes positive), the predict() function was used in relation to the probability of outcome for all cases in the data.10yr dataset. To do this, a confusion matrix needed to be initially created as shown in code 5.5.

```
> pr<-predict(lr.glm, newdata=data.10yr, type="response")
> for(i in 1:length(pr))ifelse(pr[i]>0.5, pr[i]<-1,  pr[i]<-0)
> confmat<-table(data.10yr$alivestatus, pr, dnn=c("actual","predicted"))
> confmat
      predicted
actual     0     1
     0 11313   405
     1  2702   774
```

Code 5.5: Creating a confusion matrix to assess the performance of the model.

The first cell of the top row indicates that 11313 cases have been correctly predicted as being alive after ten years. The confusion matrix as outlined previously indicates in the second cell of the top row shows 405 cases incorrectly classified as not surviving ten years even though they actually did. The first cell of the bottom row shows 2702 cases incorrectly classified as surviving ten years but who had actually died. Finally, the very last cell shows 774 cases correctly classified as dying within ten years. An estimation of the model of True Positive (TP) and True Negative (TN) was made where the accuracy of the model is accuracy = (TP+TN)/(TP+TN+FP+FN), which in 'R' gives the following value as shown in code 5.6.

```
> accuracy<-(11313+774)/nrow(data.10yr); accuracy
[1] 0.7950507

> recall<-11313/(11313+405); recall
[1] 0.9654377

> precision<-(11313)/(11313+2702); precision
[1] 0.8072065

> TNR<-774/(2702+774); TNR
[1] 0.2226697
```

Code 5.6 highlighting a summary of accuracy measures.

The accuracy of the logistic regression model based on the NPI covariates is 79.5% which suggests the model was good at predictions. The recall or true positive rate highlights that 96.4% of cases that survive ten years were correctly predicted. The "precision" or proportion of cases correctly predicted as surviving is also high at 80.7%. These figures give considerable confidence in the predictive power of the model. However, there are two problems that we have not

accounted for in these results. Examining the true negative rate ("TNR"), which is the proportion of cases correctly predicted as dying within ten years, shows a value of only 22.4%, which is clearly a very poor result.

When applying the model, it will wrongly predict that ~77% of cases with a poorer prognosis actually have a good prognosis. The reason for the bias towards accurate prediction of good prognosis patients is that the dataset was heavily biased in terms of number for these patients. Only 23% of patients in the entire dataset died within ten years, a problem that is referred to as class imbalance. A second problem is that the model needs to be tested against a different 'unseen' dataset. The model was potentially 'overfitted' to the dataset which could explain the poor in prediction of survival on any new cases which the system observed. The model thus ideally needed to be recreated by splitting the dataset. The model was then trained on one dataset and tested on another. The approach is known as the "holdout method", where alternatively a method called "k-fold cross evaluation" can be used if the dataset cannot be divided.

Class imbalance is a significant problem if the minority class has equal or more importance than the majority class. In this instance, the class representing patients that died within ten years (with smaller patient numbers) had more importance than the class stating survival (the majority class according to patient numbers). The reason for this weighting of importance is that the consequence of wrongly predicting survival for a patient who actually will die is probably much more severe (with regards to disease management) than wrongly predicting that a patient who actually survives will die. One way of accounting for this cost was to employ a "cost-sensitive" classification method which attempted to reduce the rate of false positives or false negatives according to potential consequence.

However, an alternative approach is to re-sample from the original data, creating a new dataset containing cases that confer a similar distribution for each class. This was the method adopted. A new dataset was created with all the deceased cases with the remaining alive cases under sampled. The NPI covariates were then tested upon this new dataset and cross-validated. Further procedures (e.g. a J48 decision tree, boosting) were then applied to address if it is possible to

further improve predictive accuracy of the model classifiers in the dataset as outlined earlier in the strategy section.

The approach for creating a new dataset can be seen in code 5.7.

```
> data.10yr<-data.10yr[order(data.10yr$alivestatus),]

data.10yr$alivestatus<-as.factor(data.10yr$alivestatus)
> data.10yr$grade<-as.factor(data.10yr$grade)
> data.10yr<-data.10yr[order(data.10yr$alivestatus),]

rand.0<-sample(1:11718, 3476)
alive <- data.10yr[rand.0,]
dead <- data.10yr[11718:15194,]
```

Code 5.7

The new training set was then divided into two groups which were allocated dead and alive as seen in code 5.8

```
training.split <-rbind(alive, dead)
split <- sample(nrow(training.split), floor(nrow(training.split) * 0.7))
split.train <-training.split[split, ]
split.test <-training.split[-split, ]

split.alive<-sample(nrow(alive), nrow(alive)-100)
split.dead<-sample(nrow(dead), nrow(dead)-100)
train.full <-rbind(alive[split.alive,], dead[split.dead,])
test.200 <-rbind(alive[-split.alive,], dead[-split.dead,])
```

Code 5.8

To compare the data returned against the NPI patient covariates, the functions shown in code 5.9 was applied using logistic regression using the split balanced training set.

```
lr.split<-glm(formula = alivestatus ~ size + nodespos + grade, family =
"binomial",  data = split.train)
summary(lr.split)

Call:
glm(formula = alivestatus ~ size + nodespos + grade, family = "binomial",
    data = split.train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.1258  -0.9911   0.1392   1.0500   2.0580

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.114446   0.139880 -15.116  < 2e-16 ***
size         0.031265   0.002468  12.666  < 2e-16 ***
nodespos     0.144515   0.010401  13.895  < 2e-16 ***
grade2       0.826963   0.142225   5.814 6.08e-09 ***
grade3       1.401945   0.141512   9.907  < 2e-16 ***
grade4       1.408391   0.182909   7.700 1.36e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6745.9  on 4866  degrees of freedom
Residual deviance: 5716.9  on 4861  degrees of freedom
AIC: 5728.9

Number of Fisher Scoring iterations: 5
```

Code 5.9

A predictive summary for the already-split dataset for the above model as shown in code 5.9 is shown in code 5.10.

```
pr.split<-predict(lr.split, newdata=split.test, type="response")
for(i in 1:length(pr.split))ifelse(pr.split[i]>0.5, pr.split[i]<-1,
pr.split[i]<-0)
confmat<-table(split.test$alivestatus, pr.split, dnn=c("actual","predicted"))
confmat
    predicted
actual   0   1
     0 789 292
     1 360 645

accuracy<-(816+666)/nrow(split.test); accuracy
[1] 0.7104506

precision<-(816)/(816+392); precision
[1] 0.6754967

> recall<-816/(816+212); recall
[1] 0.7937743

> TNR<-666/(666+392); TNR
[1] 0.6294896
```

Code 5.10 showing an accuracy of 71% with an increase in true negative rate to 63%.

To further examine accuracy, and determine the resulting effect of the training, 10-fold cross validation of the classifier against a breast cancer patient set comprising 200 patients was performed which originated from the SEER dataset. The procedure in 'R' of how this was performed can be seen in code 5.11

```
>lr.fold<-glm(formula = alivestatus ~ size + nodespos + grade, family =
"binomial",  data = train.full)
-----------------------------------------------
library(DAAG)

Loading required package: MASS

Attaching package: 'DAAG'

CVbinary(lr.fold)

Fold:  1 4 5 8 3 10 9 2 7 6
Internal estimate of accuracy = 0.697
Cross-validation estimate of accuracy = 0.697

pr.fold<-predict(lr.fold, newdata=test.200, type="response")
for(i in 1:length(pr.fold))ifelse(pr.fold[i]>0.5, pr.fold[i]<-1,  pr.fold[i]<-0)
confmat<-table(test.200$alivestatus, pr.fold, dnn=c("actual","predicted"))
confmat
  predicted
actual  0  1
     0 87 13
     1 41 59
```

Code 5.11 summarising with an accuracy of 72.5%.

II – J48 decision tree

Using the full dataset, all of the non-class-matched data, a J48 decision tree can also be generated to explore and potentially improve classifier accuracy as seen in code 5.12. The tree will ultimately predict survival or death over the 10 year period according to what value is reached at the end of each leaf.

```
WOW(J48)

tree.full<-J48(alivestatus ~ size + nodespos + grade, data = train.full)

tree.full

J48 pruned tree
------------------

nodespos <= 0
|   size <= 17: 0 (1890.0/435.0)
|   size > 17
|   |   size <= 40: 0 (1533.0/656.0)
|   |   size > 40: 1 (163.0/66.0)
nodespos > 0
|   nodespos <= 4
|   |   size <= 22
|   |   |   grade = 1: 0 (67.0/16.0)
|   |   |   grade = 2: 0 (385.0/159.0)
|   |   |   grade = 3: 1 (378.0/167.0)
|   |   |   grade = 4: 1 (42.0/18.0)
|   |   size > 22: 1 (993.0/299.0)
|   nodespos > 4: 1 (1302.0/218.0)

Number of Leaves  :     9

Size of the tree :      15
```

Code 5.12: J48 tree shown in summary

Decision trees as previously introduced are effective in that they provide predictive models with good accuracy yet also allow how the actual model was created to be visualized, therefore avoiding the so called 'black box' effect evident with other learning processes. The J48 decision tree generated code 5.12 is able to predict survival status, in this instance according to changes in the NPI classifiers size, grade and nodal positivity.

The overall effect of the tree on the NPI classifier accuracy can be seen in code 5.13.

```
tree.full.eval<-evaluate_Weka_classifier(tree.full, train.full, class=T)

tree.full.eval

=== Summary ===

Correctly Classified Instances        4719               69.8801 %
Incorrectly Classified Instances      2034               30.1199 %
Kappa statistic                          0.3976
Mean absolute error                      0.3995
Root mean squared error                  0.4469
Relative absolute error                 79.9011 %
Root relative squared error             89.3874 %
Total Number of Instances             6753

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.773     0.375       0.673     0.773      0.72      0.751    0
  0.625     0.227       0.733     0.625      0.675     0.751    1

=== Confusion Matrix ===

    a    b   <-- classified as
 2609  768 |    a = 0
 1266 2110 |    b = 1
```

Code 5.13: Evaluation of the J48 classifier summary results

The overall accuracy as can be seen in figure 5.20 is 69.9%. However there are additional ways to further improve classifier accuracy based around J48 decision trees, notably "boosting" and "bagging":

III – Application of Support Vector machine to evaluate accuracy of survival status.

Weka used the sequential minimal optimisation algorithm version of support vector machine, the SMO() function. Optimisation of model building parameters is significant in success of the technique however to compare the models performance against other machine learning techniques, default parameters were used.

```
> smo.full<-SMO(alivestatus ~ size + nodespos + grade,data = train.full , control
= Weka_control(K = "weka.classifiers.functions.supportVector.RBFKernel"))

> smo.full$classifer

0.1791 * <0.045226 0 0 1 0 0 > * X]
-       1       * <0.090452 0.010309 0 1 0 0 > * X]
+       1       * <0.095477 0.051546 0 0 0 1 > * X]
+       1       * <0.105528 0.010309 0 0 1 0 > * X]
-       1       * <0.060302 0 0 0 1 0 > * X]
+       1       * <0.221106 0.030928 0 1 0 0 > * X]
-       1       * <0.246231 0.010309 0 0 1 0 > * X]
-       1       * <0.070352 0 0 1 0 0 > * X]
+       1       * <0.110553 0 0 0 0 1 > * X]
```

Code 5.14– The complexity constant could be altered – by altering the flag '-C' to a different value – the default is 1 – to optimise the model.

However the classifier generated was evaluated to assess how accurate the non-optimised model performed using 10 fold cross validation as shown in code 5.15.

```
> smo.full.eval<-evaluate_Weka_classifier(smo.full, numFolds=10,train.full,
class=T)
> smo.full.eval
=== 10 Fold Cross Validation ===

=== Summary ===
Correctly Classified Instances        4620               68.414  %
Incorrectly Classified Instances      2133               31.586  %
Kappa statistic                          0.3683
Mean absolute error                      0.3159
Root mean squared error                  0.562
Relative absolute error                 63.1719 %
Root relative squared error            112.4028 %
Total Number of Instances             6753

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.697     0.329      0.679     0.697     0.688      0.684    0
  0.671     0.303      0.689     0.671     0.68       0.684    1

=== Confusion Matrix ===

    a    b   <-- classified as
 2355 1022 |    a = 0
 1111 2265 |    b = 1
```

Code 5.15 – 10 fold cross validation of the support vector machine classifier

It is clear from the validation that the technique would need to be optimised to improve the false positive rate which at 33% is at a very high cost of prediction of survival.

IV – Boosting

Instances in the training set that were previously wrongly classified were given priority by the boosting algorithm in model building. 'Adaboost' as previously introduced, is one way in which boosting can be performed as shown in code 5.16. All ten decision trees produced from applying Adaboost to the NPI classifiers in the J48 decision tree format are also shown in code 5.16 using the 'full' command and presented decreasing in model weight.

```
AdaBoostM1: Base classifiers and their weights:

J48 pruned tree
------------------

nodespos <= 1
|   size <= 20: 0 (2897.0/792.0)
|   size > 20
|   |   nodespos <= 0
|   |   |   grade = 1: 0 (46.0/13.0)
|   |   |   grade = 2
|   |   |   |   size <= 37: 0 (308.0/137.0)
|   |   |   |   size > 37: 1 (86.0/33.0)
|   |   |   grade = 3: 1 (618.0/296.0)
|   |   |   grade = 4: 1 (74.0/34.0)
|   |   nodespos > 0: 1 (423.0/147.0)
nodespos > 1
|   nodespos <= 4
|   |   size <= 20
|   |   |   grade = 1: 0 (28.0/10.0)
|   |   |   grade = 2
|   |   |   |   nodespos <= 3: 0 (134.0/54.0)
|   |   |   |   nodespos > 3
|   |   |   |   |   size <= 11: 0 (3.0)
|   |   |   |   |   size > 11: 1 (24.0/9.0)
|   |   |   grade = 3: 1 (168.0/63.0)
|   |   |   grade = 4: 0 (21.0/8.0)
|   |   size > 20: 1 (637.0/167.0)
|   nodespos > 4: 1 (1286.0/205.0)

Number of Leaves  :     15

Size of the tree :      25

Weight: 0.89

J48 pruned tree
------------------

nodespos <= 4
|   size <= 10
|   |   grade = 1
|   |   |   nodespos <= 0: 0 (169.89/30.88)
|   |   |   nodespos > 0
|   |   |   |   nodespos <= 1: 0 (6.66/1.72)
|   |   |   |   nodespos > 1: 1 (7.26/2.12)
|   |   grade = 2
|   |   |   nodespos <= 0: 0 (358.99/106.37)
|   |   |   nodespos > 0
|   |   |   |   nodespos <= 2
|   |   |   |   |   size <= 9
|   |   |   |   |   |   nodespos <= 1
|   |   |   |   |   |   |   size <= 5: 1 (4.84/1.41)
|   |   |   |   |   |   |   size > 5: 0 (10.58)
|   |   |   |   |   |   nodespos > 1
```

Code 5.16 continued overleaf…

...code 5.16 continued from previous page

```
|   |   |   |   |   |   |   size <= 7: 0 (2.12)
|   |   |   |   |   |   |   size > 7: 1 (8.68/3.53)
|   |   |   |   |   size > 9: 1 (31.17/10.58)
|   |   |   |   nodespos > 2: 0 (4.94)
|   |   grade = 3
|   |   |   size <= 3: 1 (25.62/7.06)
|   |   |   size > 3
|   |   |   |   nodespos <= 1
|   |   |   |   |   nodespos <= 0: 0 (169.44/72.06)
|   |   |   |   |   nodespos > 0: 1 (36.02/12.0)
|   |   |   |   nodespos > 1: 0 (31.48/9.17)
|   |   grade = 4: 1 (41.66/17.64)
|   size > 10
|   |   size <= 20
|   |   |   grade = 1: 0 (205.52/68.63)
|   |   |   grade = 2
|   |   |   |   nodespos <= 0
|   |   |   |   |   size <= 14: 0 (205.22/66.91)
|   |   |   |   |   size > 14: 1 (507.15/249.8)
|   |   |   |   nodespos > 0
|   |   |   |   |   nodespos <= 3: 1 (277.22/98.79)
|   |   |   |   |   nodespos > 3: 0 (27.44/10.58)
|   |   |   grade = 3
|   |   |   |   nodespos <= 1: 1 (857.06/333.77)
|   |   |   |   nodespos > 1: 0 (149.29/63.51)
|   |   |   grade = 4: 1 (118.13/40.93)
|   |   size > 20
|   |   |   nodespos <= 1
|   |   |   |   grade = 1: 0 (58.1/27.95)
|   |   |   |   grade = 2
|   |   |   |   |   nodespos <= 0
|   |   |   |   |   |   size <= 36: 1 (355.01/119.96)
|   |   |   |   |   |   size > 36: 0 (94.72/37.4)
|   |   |   |   |   nodespos > 0
|   |   |   |   |   |   size <= 41: 0 (127.72/40.22)
|   |   |   |   |   |   size > 41: 1 (14.02/3.43)
|   |   |   |   grade = 3: 0 (999.24/350.7)
|   |   |   |   grade = 4: 0 (115.1/43.04)
|   |   |   nodespos > 1
|   |   |   |   grade = 1: 0 (28.35/7.76)
|   |   |   |   grade = 2
|   |   |   |   |   size <= 47: 0 (158.27/65.62)
|   |   |   |   |   size > 47: 1 (33.28/8.58)
|   |   |   |   grade = 3: 1 (341.89/137.26)
|   |   |   |   grade = 4: 1 (56.38/27.45)
nodespos > 4
|   grade = 1
|   |   size <= 36: 0 (25.93/7.06)
|   |   size > 36: 1 (3.53)
|   grade = 2
|   |   size <= 32
|   |   |   nodespos <= 5: 0 (41.16/12.0)
|   |   |   nodespos > 5: 1 (170.55/70.34)
```

code 5.16 continued overleaf...

…code 5.16 continued from previous page

```
|   |   size > 32: 1 (116.06/25.74)
|   grade = 3: 1 (676.73/187.01)
|   grade = 4: 1 (80.57/20.59)

Number of Leaves  :     42

Size of the tree :      73


Weight: 0.6

J48 pruned tree
------------------

nodespos <= 0
|   size <= 25
|   |   grade = 1
|   |   |   size <= 12: 0 (176.67/45.87)
|   |   |   size > 12: 1 (155.2/75.53)
|   |   grade = 2: 0 (1288.91/533.93)
|   |   grade = 3
|   |   |   size <= 10: 1 (194.18/83.47)
|   |   |   size > 10: 0 (993.74/435.2)
|   |   grade = 4: 0 (148.99/69.45)
|   size > 25
|   |   grade = 1
|   |   |   size <= 65: 1 (29.39/7.66)
|   |   |   size > 65: 0 (2.19)
|   |   grade = 2: 1 (257.42/112.97)
|   |   grade = 3: 0 (430.11/202.56)
|   |   grade = 4: 0 (50.11/24.82)
nodespos > 0
|   size <= 34
|   |   grade = 1: 0 (106.02/51.4)
|   |   grade = 2: 0 (802.78/385.78)
|   |   grade = 3: 1 (1099.68/508.75)
|   |   grade = 4: 1 (141.5/67.87)
|   size > 34: 1 (876.13/359.05)

Number of Leaves  :     16

Size of the tree :      25

Weight: 0.25

J48 pruned tree
nodespos <= 7
|   grade = 1
|   |   nodespos <= 0: 0 (355.89/142.61)
|   |   nodespos > 0: 1 (111.48/47.29)
|   grade = 2
|   |   nodespos <= 0: 0 (1538.42/736.9)
|   |   nodespos > 0
```

code 5.16 continued overleaf…

...code 5.16 continued from previous page

```
|   |   |   nodespos <= 5
|   |   |   |   nodespos <= 2: 1 (526.57/251.82)
|   |   |   |   nodespos > 2
|   |   |   |   |   size <= 14: 0 (25.2/8.69)
|   |   |   |   |   size > 14: 1 (208.21/73.46)
|   |   |   nodespos > 5
|   |   |   |   nodespos <= 6
|   |   |   |   |   size <= 19
|   |   |   |   |   |   size <= 14
|   |   |   |   |   |   |   size <= 13: 1 (3.12)
|   |   |   |   |   |   |   size > 13: 0 (2.15)
|   |   |   |   |   |   size > 14: 1 (4.36)
|   |   |   |   |   size > 19: 0 (40.34/13.69)
|   |   |   |   nodespos > 6
|   |   |   |   |   size <= 57: 0 (27.26/10.41)
|   |   |   |   |   size > 57: 1 (2.93)
|   grade = 3
|   |   nodespos <= 1: 1 (2108.04/1029.64)
|   |   nodespos > 1: 0 (790.66/348.68)
|   grade = 4
|   |   nodespos <= 1: 1 (254.68/116.85)
|   |   nodespos > 1
|   |   |   nodespos <= 5: 0 (106.01/36.31)
|   |   |   nodespos > 5
|   |   |   |   size <= 75
|   |   |   |   |   size <= 20: 0 (4.21/1.46)
|   |   |   |   |   size > 20: 1 (7.32)
|   |   |   |   size > 75: 0 (2.75)
nodespos > 7
|   size <= 60
|   |   grade = 1
|   |   |   size <= 18: 0 (3.56)
|   |   |   size > 18
|   |   |   |   nodespos <= 9: 1 (2.26)
|   |   |   |   nodespos > 9: 0 (3.5/1.13)
|   |   grade = 2
|   |   |   size <= 10: 0 (20.67/5.61)
|   |   |   size > 10
|   |   |   |   size <= 20: 1 (32.36/4.3)
|   |   |   |   size > 20
|   |   |   |   |   size <= 32: 0 (78.22/33.04)
|   |   |   |   |   size > 32: 1 (34.68/11.0)
|   |   grade = 3
|   |   |   size <= 21
|   |   |   |   nodespos <= 10: 1 (30.51/11.0)
|   |   |   |   nodespos > 10: 0 (92.69/32.19)
|   |   |   size > 21: 1 (222.93/90.75)
|   |   grade = 4
|   |   |   size <= 19: 1 (5.37)
|   |   |   size > 19
|   |   |   |   nodespos <= 21
|   |   |   |   |   nodespos <= 8: 1 (2.93)
|   |   |   |   |   nodespos > 8: 0 (22.04/8.29)
```

code 5.16 continued overleaf...

...code 5.16 continued from previous page

```
|   size > 60
|   |   size <= 80
|   |   |   nodespos <= 40: 1 (33.65)
|   |   |   nodespos > 40: 0 (3.24/0.49)
|   |   size > 80: 1 (39.91/16.5)

Number of Leaves  :     36

Size of the tree :      67


Weight: 0.2

J48 pruned tree
------------------

grade = 1
|   size <= 12: 0 (190.48/69.35)
|   size > 12: 1 (281.92/130.46)
grade = 2: 1 (2563.9/1223.82)
grade = 3
|   nodespos <= 4: 0 (2671.55/1238.68)
|   nodespos > 4: 1 (632.2/294.67)
grade = 4
|   nodespos <= 5: 0 (358.88/165.51)
|   nodespos > 5: 1 (54.07/20.53)

Number of Leaves  :     7

Size of the tree :      11

Weight: 0.14

J48 pruned tree
------------------
: 0 (6753.0/3325.14)

Number of Leaves  :     1

Size of the tree :      1

Weight: 0.03

J48 pruned tree
------------------

grade = 1: 0 (469.15/219.51)
grade = 2: 0 (2567.8/1272.77)
grade = 3: 1 (3303.76/1631.94)
grade = 4: 1 (412.28/199.88)
Number of Leaves  :     4
Size of the tree :      5
Weight: 0.03
```

Code 5.16 continued overleaf...

```
...code 5.16 continued from previous page
J48 pruned tree
------------------
: 0 (6753.0/3371.23)

Number of Leaves  :      1

Size of the tree :      1


Weight: 0.0

J48 pruned tree
------------------

grade = 1: 0 (468.76/223.31)
grade = 2: 1 (2568.11/1273.25)
grade = 3: 0 (3303.93/1648.85)
grade = 4: 1 (412.2/202.72)

Number of Leaves  :      4

Size of the tree :      5


Weight: 0.02

J48 pruned tree
------------------
: 1 (6753.0/3373.17)

Number of Leaves  :      1

Size of the tree :      1


Weight: 0.0

Number of performed Iterations: 10
```

Code 5.16 J48 tree shown in its entirety as an example.

As before, predictive accuracy were estimated, and 10 fold-cross validation performed to validate the models, as shown in code 5.17. This revealed predictive accuracy of 70% as shown when the model was further validated against an additional small (200) test breast cancer set can be seen in code 5.17.

```
> boost.full.eval<-evaluate_Weka_classifier(boost.full, numFolds=10,train.full,
class=T)
> boost.full.eval
=== 10 Fold Cross Validation ===

=== Summary ===

Correctly Classified Instances        4724               69.9541 %
Incorrectly Classified Instances      2029               30.0459 %
Kappa statistic                          0.3991
Mean absolute error                      0.3831
Root mean squared error                  0.4432
Relative absolute error                 76.629  %
Root relative squared error             88.6337 %
Total Number of Instances             6753

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.704     0.305       0.698     0.704     0.701      0.767    0
  0.695     0.296       0.701     0.695     0.698      0.767    1

=== Confusion Matrix ===

    a    b   <-- classified as
 2379  998 |    a = 0
 1031 2345 |    b = 1
```

Code 5.17: 10 fold cross evaluation of results.

The results show, in code 5.18, that boosting the test set has improved the accuracy of error rate in comparison to the decision tree without applying boosting.

```
> boost.200.eval<-evaluate_Weka_classifier(boost.full, numFolds=10,test.200,
class=T)
> boost.200.eval
=== 10 Fold Cross Validation ===
=== Summary ===

Correctly Classified Instances         139               69.5    %
Incorrectly Classified Instances        61               30.5    %
Kappa statistic                          0.39
Mean absolute error                      0.3662
Root mean squared error                  0.4494
Relative absolute error                 73.2419 %
Root relative squared error             89.8702 %
Total Number of Instances              200

=== Detailed Accuracy By Class ===
TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.77      0.38       0.67      0.77      0.716      0.762    0
  0.62      0.23       0.729     0.62      0.67       0.762    1

=== Confusion Matrix ===
  a  b   <-- classified as
 77 23 |  a = 0
 38 62 |  b = 1
```

Code 5.18 showing the results of applying cross validation to the boosted classifier.

V – Bagging

Bagging, as previously introduced, is an alternative 'meta' classifier to boosting. Here, J48 decision trees were once again produced; however only the first two trees are displayed in this instance as seen in code 5.19.

```
> bag.full<-Bagging(alivestatus ~ size + nodespos + grade,data = train.full,
control = Weka_control(W = "J48"))
> bag.full
All the base classifiers:

J48 pruned tree
------------------

nodespos <= 1
|   size <= 20: 0 (2891.0/783.0)
|   size > 20
|   |   nodespos <= 0
|   |   |   size <= 49: 0 (1003.0/463.0)
|   |   |   size > 49: 1 (150.0/50.0)
|   |   nodespos > 0
|   |   |   grade = 1
|   |   |   |   size <= 35
|   |   |   |   |   size <= 25: 1 (6.0/1.0)
|   |   |   |   |   size > 25: 0 (3.0)
|   |   |   |   size > 35: 1 (4.0)
|   |   |   grade = 2
|   |   |   |   size <= 41
|   |   |   |   |   size <= 38: 1 (81.0/34.0)
|   |   |   |   |   size > 38: 0 (11.0/1.0)
|   |   |   |   size > 41: 1 (14.0/2.0)
|   |   |   grade = 3: 1 (251.0/79.0)
|   |   |   grade = 4: 1 (19.0/6.0)
nodespos > 1
|   nodespos <= 4
|   |   size <= 24
|   |   |   grade = 1: 0 (34.0/14.0)
|   |   |   grade = 2
|   |   |   |   nodespos <= 2: 0 (93.0/33.0)
|   |   |   |   nodespos > 2
|   |   |   |   |   size <= 11: 0 (8.0)
|   |   |   |   |   size > 11: 1 (79.0/31.0)
|   |   |   grade = 3
|   |   |   |   nodespos <= 2
|   |   |   |   |   size <= 11
|   |   |   |   |   |   size <= 9
|   |   |   |   |   |   |   size <= 8: 0 (4.0/1.0)
|   |   |   |   |   |   |   size > 8: 1 (4.0)
|   |   |   |   |   |   size > 9: 0 (14.0/3.0)
|   |   |   |   |   size > 11: 1 (83.0/29.0)
|   |   |   |   nodespos > 2: 1 (130.0/43.0)
|   |   |   grade = 4
|   |   |   |   size <= 19: 0 (16.0/2.0)
|   |   |   |   size > 19: 1 (10.0/3.0)
|   |   size > 24: 1 (586.0/143.0)
|   nodespos > 4: 1 (1259.0/188.0)

Number of Leaves  :     24
Size of the tree :      43
```

Code 5.19 continued overleaf…

```
...code 5.19 continued from previous page

J48 pruned tree
------------------

nodespos <= 1
|   size <= 20: 0 (2871.0/788.0)
|   size > 20
|   |   nodespos <= 0
|   |   |   size <= 37: 0 (870.0/395.0)
|   |   |   size > 37: 1 (252.0/93.0)
|   |   nodespos > 0: 1 (428.0/155.0)
nodespos > 1
|   nodespos <= 5
|   |   size <= 19
|   |   |   grade = 1: 0 (27.0/5.0)
|   |   |   grade = 2: 0 (125.0/50.0)
|   |   |   grade = 3: 1 (131.0/52.0)
|   |   |   grade = 4
|   |   |   |   size <= 15
|   |   |   |   |   nodespos <= 4
|   |   |   |   |   |   nodespos <= 2: 0 (10.0/4.0)
|   |   |   |   |   |   nodespos > 2
|   |   |   |   |   |   |   size <= 14: 0 (2.0)
|   |   |   |   |   |   |   size > 14: 1 (2.0)
|   |   |   |   |   nodespos > 4: 1 (2.0)
|   |   |   |   size > 15: 0 (7.0)
|   |   size > 19: 1 (914.0/226.0)
|   nodespos > 5: 1 (1112.0/164.0)

Number of Leaves  :     14

Size of the tree :      25
```

Code 5.19: The effect of bagging on the J48 tree.

Once again 10 fold cross validation is performed to evaluate the classifier as seen in code 5.20.

```
> bag.full.eval<-evaluate_Weka_classifier(bag.full, numFolds=10,train.full,
class=T)
> bag.full.eval
=== 10 Fold Cross Validation ===

=== Summary ===

Correctly Classified Instances        4730               70.0429 %
Incorrectly Classified Instances      2023               29.9571 %
Kappa statistic                          0.4009
Mean absolute error                      0.3965
Root mean squared error                  0.4455
Relative absolute error                 79.2935 %
Root relative squared error             89.0987 %
Total Number of Instances             6753

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.738     0.337      0.687     0.738     0.711      0.759    0
  0.663     0.262      0.717     0.663     0.689      0.759    1

=== Confusion Matrix ===

    a    b   <-- classified as
 2492  885 |    a = 0
 1138 2238 |    b = 1
```

Code 5.20: The effect of bagging and 10 fold cross validation upon the results.

The model produced by bagging was tested once again against the 200 breast cancer training set as seen in code 5.21:

```
> bag.200.eval<-evaluate_Weka_classifier(bag.full, numFolds=10,test.200, class=T)
> bag.200.eval
=== 10 Fold Cross Validation ===
=== Summary ===

Correctly Classified Instances         135               67.5    %
Incorrectly Classified Instances        65               32.5    %
Kappa statistic                          0.35
Mean absolute error                      0.3882
Root mean squared error                  0.4563
Relative absolute error                 77.6311 %
Root relative squared error             91.2663 %
Total Number of Instances              200

=== Detailed Accuracy By Class ===
TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.72      0.37       0.661     0.72      0.689      0.738    0
  0.63      0.28       0.692     0.63      0.66       0.738    1

=== Confusion Matrix ===
  a  b   <-- classified as
 72 28 |  a = 0
 37 63 |  b = 1
```

Code 5.21: Evaluation of the bagged model on a 200 patient training set.

Again an improvement is seen with the test set using this alternative method of optimising the tree results as shown in code 5.21, however improvements in accuracy over the J48 method were not observed.

VI – Creating new classifiers from Weka analysis strategies to use within 'R'

There are many analysis strategies to reveal classifiers available in the Java application Weka. Only selected classifiers are precompiled for use instantly in 'R', however using the 'make_weka_classifier' function within the R Weka package, Weka classifiers can be imported for use into 'R'. The Random Forest approach was chosen to be added to the RWeka portfolio of analysis approaches to determine classifiers. The 'I' parameters (e.g: I=1000) in the 'Weka_control' specifies however many trees to produce from the 3 covariates. 1000 trees were chosen which took an extended time to process, as seen in code 5.22.

```
> rf <- make_Weka_classifier("weka/classifiers/trees/RandomForest")

> randf.full <- rf(alivestatus ~ size + nodespos + grade, data = train.full,
control = Weka_control(I = 1000))

> randf.full

Random forest of 1000 trees, each constructed while considering 3 random
features.
Out of bag error: 0.3336
```

Code 5.22: Creating a new classifier through Weka imported in 'R'.

The 'out of bag error' is calculated by Random Forest as a result for the given dataset. This value is useful as cross validation is not required subsequently. It is created as part of the Random Forest process after this splits the data into two thirds training and one third testing sets. Due to the large number of trees examined in the procedure these are not shown individually as per previous classifier methods; however, a summary can be produced. Again this can take an extended time to process on slower computers, where the summary process is seen in code 5.23:

```
> summary(randf.full)

=== Summary ===

Correctly Classified Instances        5068               75.0481 %
Incorrectly Classified Instances      1685               24.9519 %
Kappa statistic                          0.501
Mean absolute error                      0.3269
Root mean squared error                  0.4001
Relative absolute error                 65.3899 %
Root relative squared error             80.0218 %
Total Number of Instances             6753

=== Confusion Matrix ===

    a    b   <-- classified as
 2711  666 |    a = 0
 1019 2357 |    b = 1
```

Code 5.23: Summary of random forest classifier results.

In this instance, it appears correctly classified instances are superior at 75%.

5.6.3 Summary – cross comparison of machine learning performance to predict survival or death

It is important not to forget the purpose of fitting a particular model to a dataset: to be able to accurately predict a particular outcome according to a given covariate profile. Here, it was to predict survival based on the NPI covariate formula yet analyse how altering the interactions between the covariates nodes positive, grade and tumour size impacted survival. Early exploration of the data revealed hurdles such as class imbalance which needed to be addressed, resulting in further model optimisation by considering different modelling processes (including machine learning algorithms) in this thesis. This in itself resulted in a false sense of security form an accuracy point of view as subsequent methods appeared to perform more poorly. Table 5.5 summarises the results of each process, showing the effect on accuracy of prediction given by each modelling technique.

Classifier process/Machine learning algorithm	Life	Death	Overall accuracy
Logistic regression	11313 correctly predicted as surviving 2702 incorrectly predicted as surviving	774 correctly predicted as dying 405 incorrectly predicted as dying	79.5%
Splitting data to address class imbalance on test set	789 correctly predicted as surviving 360 incorrectly predicted as surviving	645 correctly predicted as dying 292 incorrectly predicted as dead	71.0%
Further testing and re-training	87 correctly predicted as surviving 41 incorrectly predicted as surviving	59 correctly predicted as dying 13 incorrectly predicted as dead	69.7%
J48 tree	4719 correctly predicted	2034 incorrectly predicted	69.9%
Adaboost	4724 correctly predicted	2029 incorrectly predicted	69.9%
Adaboost on test set	139 correctly predicted	61 incorrectly predicted	69.5%
Bagging	4730 correctly predicted	2023 incorrectly predicted	70.0%
Bagging on test set	135 correctly predicted	65 incorrectly predicted	67.5%
Random Forest	5068 correctly predicted	1685 incorrectly predicted	75.0%

Table 5.5: Comparison of the different classifiers to predict survival in breast cancer using the SEER dataset.

VII – Probability of death or survival using NBTree – a naïve Bayes classification method

Previous modelling stages of boosting and decision tree analysis show the ability to predict outcome in an all or nothing fashion. The patient will either die or survive over the 10 year period. However in reality this is not as very informative as the models themselves are not 100% accurate and vary as previously demonstrated.

However for an oncologist, it is more meaningful to make a prognosis and determine the chances of death or survival in a given time frame chosen by themselves. In this regard, Bayes classification can be used applied. This is a method of calculating conditional probabilities and can be compared with decision tree methods. Such a method is available to be imported from within Weka for use in 'R' called NBTree. To apply NBTree to the dataset, any continuous variables need to be initially factorised in 'R', and so size is allocated over a range of one through five, and nodes positive from zero through five as seen in code 5.24. Grade is already factorised on a scale of 1-3.

```
> nbtree <- make_Weka_classifier("weka/classifiers/trees/NBTree")
> train.full.factor<-train.full
> for(i in 1:length(train.full.factor$nodespos))
if(train.full.factor$nodespos[i]>5){train.full.factor$nodespos[i]<-5}
> for(i in 1:length(train.full.factor$size))
if(train.full.factor$size[i]>5){train.full.factor$size[i]<-5}
```

Code 5.24: Importing the NBTree classifier into 'R' from Weka.

The "Train.full.factor" variables are then parsed into the script for creating decision trees in a comparable manner to other methods shown before as seen in code 5.25.

```
> nbtree.full <- nbtree(alivestatus ~ size + nodespos + grade, data =
train.full.factor)
> nbtree.full
NBTree
------------------

nodespos <= 1.5
|   nodespos <= 0.5
|   |   size <= 4.5
|   |   |   grade = 1: NB 4
|   |   |   grade = 2: NB 5
|   |   |   grade = 3: NB 6
|   |   |   grade = 4: NB 7
|   |   size > 4.5
|   |   |   grade = 1: NB 9
|   |   |   grade = 2: NB 10
|   |   |   grade = 3: NB 11
|   |   |   grade = 4: NB 12
|   nodespos > 0.5: NB 13
nodespos > 1.5: NB 14

Leaf number: 4 Naive Bayes Classifier

Class 0: Prior probability = 0.83

size:  Discrete Estimator. Counts =  24  (Total = 24)
nodespos:  Discrete Estimator. Counts =  24  (Total = 24)
grade:  Discrete Estimator. Counts =  24 1 1 1  (Total = 27)


Class 1: Prior probability = 0.17

size:  Discrete Estimator. Counts =  5  (Total = 5)
nodespos:  Discrete Estimator. Counts =  5  (Total = 5)
grade:  Discrete Estimator. Counts =  5 1 1 1  (Total = 8)

// Results truncated - showing only first two //

Number of Leaves  :     10

Size of the tree :      15
```

Code 5.25: Applying the Naïve Bayes classifier to the full patient training set.

The resultant tree is similar to that of the J48 tree shown earlier in section 5.8. However the leaves are numbered which corresponds to the Bayes classifiers which follow the tree. For example as above following leaf number four gives a probability of survival of 0.83.

The results can then be cross validated using 10 fold cross validation as shown earlier as seen in code 5.26.

```
> nbtree.full.eval<-evaluate_Weka_classifier(nbtree.full, numFolds=10,train.full,
class=T)
> nbtree.full.eval
=== 10 Fold Cross Validation ===

=== Summary ===

Correctly Classified Instances        4732               70.0726 %
Incorrectly Classified Instances      2021               29.9274 %
Kappa statistic                          0.4015
Mean absolute error                      0.3568
Root mean squared error                  0.4441
Relative absolute error                 71.3566 %
Root relative squared error             88.8219 %
Total Number of Instances             6753

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.7       0.299      0.701     0.7       0.701      0.773    0
  0.701     0.3        0.7       0.701     0.701      0.773    1

=== Confusion Matrix ===

    a    b   <-- classified as
 2364 1013 |    a = 0
 1008 2368 |    b = 1
```

Code 5.26: 10 fold cross validation of the Naïve Bayes classifier.

As can be seen from above, a reasonable accuracy of 70% is obtained. The next step was to compare against the 200 training set as shown in code 5.27.

```
> nbtree.200.eval<-evaluate_Weka_classifier(nbtree.full, numFolds=10,test.200,
class=T)
> nbtree.200.eval
=== 10 Fold Cross Validation ===

=== Summary ===

Correctly Classified Instances         132               66      %
Incorrectly Classified Instances        68               34      %
Kappa statistic                          0.32
Mean absolute error                      0.4144
Root mean squared error                  0.4701
Relative absolute error                 82.8846 %
Root relative squared error             94.0295 %
Total Number of Instances              200

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
  0.71      0.39       0.645     0.71      0.676      0.711    0
  0.61      0.29       0.678     0.61      0.642      0.711    1

=== Confusion Matrix ===

  a  b   <-- classified as
 71 29 |  a = 0
 39 61 |  b = 1
```

Code 5.27: Application of the classifier determined by the Naïve Bayes against the 200 patient training set.

Again, the correctly classified instances are slightly higher in the training set giving whole number of 70% as opposed to 66% in the test set.

However it is important to remember that the Bayes classification gives a probability within a given time frame and not an ultimate alive or dead classification.

Classifier process	Life	Death	Overall accuracy
Bayes (NBTree)	4732 correctly predicted	2021 incorrectly predicted	70%
Bayes (NBTree) on test set	132 correctly predicted	68 incorrectly predicted	66%

Table 5.6 Summary of the results of Naïve Bayes classification to assess probability of death or survival.

5.7 Discussion

5.7.1 Nottingham prognostic index applied to the SEER breast cancer dataset

The comparison of the NPI against the SEER dataset produced an interesting comparison which has not been performed before to the best of my knowledge. It is an interesting hypothesis to compare a model based on breast cancer patients from the early 1980's to that of patients twenty years later given the advances in breast cancer research. The increased use of targeted therapies, such as anti-*HER2*-selective antibodies and hormonal therapy, should result in improved survival in hormone receptor positive as well as *HER2* positive patients (Modlich et al, 2006) [12]. Consequently, regular updates of models may have been required as new data continually becomes available. Yet it was clear from the results that the SEER dataset performed better than a UK based dataset upon which the NPI was based which could be a reflection of improved treatment regimes.

Comparing models with data sets from international sources can only be beneficial in the long term to help reveal all facets of *in vivo* breast cancer. The combining of patient data could also provide more accurate explanations for rare subgroups, such as anomalies shown with race as highlighted in Chapter 4. This should be possible where measurement of prognostic indices common across international borders is possible. Blamey et al. in a recent publication speculate that improved survival over time must come from improved patient management, since the distribution of patients into the NPI subgroups has remained essentially the same since the 1980s (Blamey et al, 2007) [111]. The better results yielded by the application of the NPI criteria against the SEER dataset may also be explained by a more accurate lymph node staging in combination with more patients being detected within mammography screening. This was certainly observed in patients classified as node negative in the 1980s, but actually being node positive, may have 'stage migrated' to a higher NPI category in the 1990s (Blamey et al, 2007) [111].

5.7.2 Modelling survival

As a result of the NPI grouping a US dataset with similar success to that of a UK dataset, the SEER dataset was used to refine the covariate combinations outlined in the NPI. Potential combinations of existing covariates – namely nodes positive status, tumour grade and tumour size were analysed using different machine learning and classification statistical techniques. After overcoming problems which bias the data such as class imbalance as discussed earlier, performance for predicting life or death varied greatly. Throughout this final chapter, it should also be clear that once again that 'R' has proven a valuable data analysis tool. The ability to interface with Weka through small amounts of coding allows access to powerful machine learning techniques.

During testing, some machine learning algorithms took far longer to perform than other machine learning algorithms. Support Vector Machine (SVM) took an extended duration for processing. It would be useful in future to recreate results using parallel computing architecture and the associated speed advantages. This would be particularly useful for optimisation where multiple cycles of the algorithm will be required.

In assessing the value of one algorithm over another, difficulties in such a comparison have been widely reported. For example, assessing accuracy scores without considering statistical significance tests that take into account the specific data sampling strategy and correct for multiple testing is unsatisfactory. Robustness in predicting survival is important if any of the classifiers described are to be used routinely. However it was also important to start the analysis of the different methods with logistic regression, particularly after class imbalance was addressed. It is important not to overlook simpler models in favour of more complex algorithms in the hope they would perform better.

Evaluation of each method was performed by 10 fold cross validation in most instances. Where a tree based method was employed, training of the classifier at each node was needed in separate succession. The process continued until reaching a leaf where no further classification occurred.

An ultimate value of accuracy for each method was reported in a confusion matrix which reported the number of patients correctly and incorrectly classified.

The random forest algorithm performed better in accuracy and also performed quicker in 'R' than most of the other methods outlined. Random forest inherently does not appear to be susceptible to 'overfitting' which some of the models such as SVM can exhibit. It is highly likely that repeated optimisation of the SVM technique could improve upon accuracy of that of even the random forest method which scored the highest accuracy in testing.

Literature suggests that the standard procedure when fitting data models such as logistic regression to a dataset is to delete variables to improve accuracy. However a model such as random forest performs better with more variables.

However, decision tree learners consider only one attribute at a time, such than relevance distributed among several attributes cannot be detected. They provide explicit rules and ordering of the decision tree by means of their depth within the tree. Yet, decision trees are sensitive with respect to disturbed data, which lead to instable solutions, i.e. different resulting tree structures. One future possible solution is to combine tree generating systems with robust classification schemes like neural networks. One approach based on prototype based classifiers is BB-trees which can be used for decision system generation (Moguerza et al, 2006) [120].

For all classification methods the underlying metric plays a crucial role: the metric can be chosen in agreement with the classification task or may be contradictious in the worst case. Therefore, adaptive, non-standard metrics are required for optimum classification (Yang et al, 2007) [121]. Whereas some machine learning techniques inherently weight the data streams during learning, classifiers such as SVMs can be extended to deal with metric adaptation and non-standard metrics as demonstrated in gene expression analysis (Moguerza et al, 2006) [120]. The same is true when applied to the patient dataset as outlined.

The Bayes classifier which uses all the prognostic variables of the dataset to predict survival probability I would imagine would be of more usefulness than a decision tree to a clinical as it

takes into account all the input patient variables. Although decision tree methods such as random forest performed well, only a selection of variables will be used when following a particular path.

It would be of value to compare the different machine learning techniques against different cancer datasets, such as the colorectal dataset also available as used in Chapter 4 with Superstes. This way a more overall picture of accuracy of each technique could be judged.

Using alternative patient variables could also potentially improve classification results. Collection of patient variables using pre-operative techniques can only cover certain pieces of information which allow a diagnosis to be performed. Information collected post-operative will be more informative due to the nature of information collected due to the invasive procedures which would have been followed. It would be interesting therefore to repeat the classification methods with a pre and post operative dataset to see if the post operative variables are able to classify survival more accurately.

The benefits for a more individual patient tailored and quantitative prediction of outcome is appealing and without doubt important in making treatment decisions. It can also be seen as part of a more general trend within medicine which strives for personalised, absolute risk assessment.

Chapter 6

Discussion

Chapter 6 – Discussion

The power of biomarkers and genetic probes in understanding cancer biology has been clearly demonstrated in both *in vitro* and *in vivo* throughout this project. Many studies have tried to discover and yet improve upon genetic signatures to predict the onset of cancer or prevent resistance to life saving therapies. Computer technology and advanced mathematical analysis through pattern discovery and classification allow such genetic signatures to be revealed.

Development of the data analysis platforms throughout this study have been made possible only due to open source initiatives. The 'R' project, R-(D)-COM, OpenGL and Visual Basic.net have all played fundamental roles in creating I-10 and Superstes (Bioconductor Core, 2002) [73], Shreiner et al, 2005) [104]. The knowledge discovery tools created will facilitate cancer researchers to analyse different types of biomedical data more rapidly than they may have using past tools. This is in part due to careful user interface design coupled with powerful and versatile statistical methodologies such as 'R' provides.

Throughout the project, technologies employed in biomedical data analysis, while primarily focusing on Affymetrix array data, also introduced the value of clinical dataset analysis. This was facilitated by development of a new arsenal of tools for future gene discovery making existing technologies more accessible to the end user as well as introducing new, perhaps unfamiliar approaches. However these tools could become quickly dated if they are not designed to be flexible for future improvement. Incorporation of 'R' technology and its associated algorithm libraries facilitates the potential for future expansion and development of both 'Informatics Tenovus' and 'Superstes' applications beyond the scope of this thesis.

6.1 Aim 1 - Development of a user friendly Affymetrix Microarray suite

Detailed analysis of *in vitro* models explored using Affymetrix HGU-133A arrays, using a purpose built system, marked a new chapter of array research for the Tenovus centre for cancer research. Although the commercial data analysis platform, Genesifter, had been used extensively to explore individual probe sets revealed by established statistical techniques, I-10 was a more

adventurous step for Tenovus. I-10 facilitates exploration of visualisation methodologies on an unprecedented level in both two and three dimensions, yet utilises well developed resources of the acclaimed statistical programming environment 'R' (Dessau et al, 2008) [73]. Through careful design, I-10 allows easy access to data analysis techniques for microarray data which otherwise might have not been explored using 'R' directly. It has facilitated biologists wishing to fully embrace microarray analysis to harness the benefits of 'R' without any in-depth coding knowledge using the command line.

The benefits of combining Microsoft Excel with 'R' and 3D graphical abilities were also demonstrated in I-10. A review of existing analysis technologies – particularly those including an interface with 'R' – showed extensive use of Excel. The usage of Excel internationally is well documented and has remained the leading spreadsheet application in many disciplines throughout the world. Consequently, due to widespread familiarity of Excel especially by Tenovus researchers who use I-10, it made an obvious choice for reviewing loaded datasets and analysis results.

Before the inception of this project, no storage procedure existed through which data generated from microarrays was stored securely within Tenovus. Although most arrays were stored online for analysis with Genesifter there was no independent storage within Tenovus to enable fast retrieval of array information. Files which contain the microarray scan information, which form the basis of analysis, were stored on CD-ROM. Furthermore, the file naming convention which was adopted by the Affymetrix scanning facility was not amenable to clear combining of arrays into projects for analysis. Consequently, the Microsoft Access database produced – viewable on all computers with Microsoft Access throughout Tenovus – has allowed easier and more robust accessibility to their valuable microarray resources. This allowed microarray model system projects to be produced more rapidly and allowed I-10 to ultimately have a data analysis management system for storage of results.

The major advantage of I-10 is in its ability to be continually upgraded and expanded. The power that adding new 'R' libraries by members of the 'R' developer community brings to I-10, should not be underestimated. I-10 will always serve as a user friendly mechanism through which

powerful new libraries can be accessed by those not wishing to learn scripting of 'R' via the console.

The availability of OpenGL to be commanded in Visual Basic through the freely available API added a new dimension to data analysis graphical abilities (Shreiner et al, 2005) [104]. Although 'R' has some inbuilt 3D functionality, the added flexibility of how scatter plots generated using OpenGL could be labelled, rotated and enlarged was novel in I-10. The 3D user interface is able to visualise results from the clustering techniques PAM, PCA, K-means and Fuzzy clustering (Hartigan et al, 1979) [44], (Du et al, 2008) [45], (Bozinov et al, 2002) [46]. Although user feedback was largely positive, some users found the 3D view on data confusing, especially with large plots. However, during user workshops with Tenovus researchers who use the software, all users were clear that it provided an interesting, quick insight into potentially interesting groups within data for any given analysis question.

Contributions from the 'R' Bioconductor community – namely through the library (simpleaffy) – proved valuable in the addition of array quality control functionality (Bioconductor Core, 2002) [72]. Detailed quality analysis had not been performed by Tenovus in previous analysis via the MVA plot system. Addition of this functionality quickly proved useful when testing the system to address a particular biological question where one control MCF7 microarray model was lower in overall similarity to its corresponding replicates. The array was ultimately reproduced and shown to be an improved match on the original array, using IQR measurement assessment. This ultimately improved results when a detailed analysis was performed using I-10, allowing more significant probes to be revealed due to greater similarity between control replicates.

Examining commercial applications also proved valuable in judging what functionality was lacking in terms of recent analysis methodologies in relation to what was possible in development of a new system. Any developed system is only successful if it addresses the requirements of the user. For example, one function of Genesifter, which was required in I-10 was a profile viewer. This facilitates fast viewing of individual probe sets in a given analysis question to see true differential expression based on normalised probe intensity values. When used in context,

induced and suppressed events of individual probes in comparison to a control microarray can be individually visualised. This has also proven useful for exploration of individual clusters.

When I-10 development was complete, changes to the Microsoft XP operation caused slight problems upon installation during initial (beta) testing. It was discovered that computers other than the development computer of which I-10 was produced lacked key files which were required by I-10. The Microsoft XP windows update tool examines each computer individually. It analyses what applications are installed and what updates are therefore required – it will vary from one computer to another. Few computers will be identical. During the many months of development, the development machine was updated with Windows update recognising the presence of Visual Basic installed on the computer. Applications developed in this way on Windows require certain system files. However due to shortcomings in security found over time on Windows XP machines, only those computers with Visual Basic applications received the updated files. Therefore when I-10 was finished and installed on other machines in Tenovus, these computers lacked the updated files which caused problems when trying to run I-10. These computers which required I-10 had to be manually updated to receive the update patches after which I-10 ran successfully.

This scenario raised an interesting issue of development of applications with Visual Basic on operating systems far newer than when the programming language was developed. This is one factor advocating all future application development in a web based manner only. However it is also possible to take the cynical point of that it is a form of Microsoft discouraging the use of older applications on their newer operating systems by making it harder to run older software. Of course, an activity always marketed from the point of view of 'security'. I-10 has not been tested with Windows Vista.

Due to experience gained through production of Superstes as outlined in Chapter 4, I would recommend that future developments of I-10 migrate to a web based platform. Since the inception of the project, there have been improvements with 3D visualisation for online applications using applications such as Adobe Flex and the open source equivalent – OpenLaszlo (Openlaszlo, 2008) [126]. Both allow creation of 3D graphs as produced in I-10 without any

particular hardware requirements. Created applications are viewed through Adobe Flash Player which all current Internet browsers support. The visualisation technology used in both Flex and OpenLaszlo originated in animation development. Recently, technical changes in the open source equivalent, OpenLaszlo, such as syntax expression alterations have changed from initial releases. This has had an impact on web based applications developed with the beta release requiring radical changes to be compatible with current and future versions of the OpenLaszlo application. The commercial Adobe product 'Flex' has not been plagued by such syntax alterations (Adobe, 2008) [127]. This raises an interesting point that although open source technologies enable very powerful applications to be developed, there is a risk that future developments could impact previously developed applications as users control development. This tends not to happen in such a radical way with commercial applications due to previous commitments and responsibilities in software license agreements. However, if I-10 had been created in OpenLaszlo to harness the 3D modelling capabilities, this could have had a drastic impact on future upgrades of I-10.

Experience with Microsoft SQL server and also alternative database solutions which 'R' can interact with, such as mySQL, could prove a worthwhile upgrade to I-10. When I-10 was developed, a key aspect to consider was to avoid expensive overheads requiring new equipment. Only later in the project was access to a Windows Server possible. Although any desktop computer can be configured to act as a 'web server' which displays locally stored web based files and applications, operating system versions of Microsoft XP often called 'images' remove the capability of acting as a server for security reasons. Consequently this was another reason option a local application version of I-10 was chosen. However as University policy ever changes in relation to Information Technology, this could be an option which could be rediscovered.

It is hoped the Informatics 10 application will prove valuable for future gene discovery applied to new Affymetrix arrays and other types of bio medical data for years to come. A future step could be to include I-10 as part of the network distributed applications so it could be used by an even wider audience. The ability of I-10 to be expandable if and when new 'R' libraries are introduced is a very powerful ability. In fact as users within Tenovus feedback their ongoing experience with the application, changes and improvements will inevitably made in future releases. Many commercial analysis products may offer some functional similarity aspects due to the nature of

some of the statistical methodologies; however, very few will offer the versatility, speed and ease of use that I-10 provides.

6.2 Aim 2 - Demonstrate the capability of the developed I-10 software to identify differential gene expression in order to assist further understanding of resistance to Tamoxifen or Faslodex

Several microarray studies have previously been described in the context of deciphering response and resistance to endocrine agents in breast cancer, where these have been applied to *in vitro* models and *in vivo* breast cancer material.

For example, some research groups have compared the transcriptional impact of oestrogen and various antioestrogens including the selective estrogen receptor modulators (SERMS) tamoxifen and raloxifene as well as the pure antioestrogen faslodex (ICI182780), on *ER+* breast cancer cell lines during the responsive phase. These studies, such as that described by Frasor *et al* Frasor et al, 2004) [128] which used Affymetrix HGU-133A microarrays, have been able to discern fundamental differences among these agents. A model to define oestradiol-like (pro-survival) and antioestrogen-like (pro-apoptotic) activities of SERMs on the basis of their various gene expression profiles has been described using the more limited Atlas cDNA array platform (Levenson et al, 2002) [129]. Another *in vitro* study utilised an alternative array platform – high-density cDNA microarrays- to again assess differences in different antioestrogen impact, but where the gene signatures of the study could subsequently be exploited to screen novel hormonal antagonists that could prove more effective against breast cancer (Scafoglio et al, 2006) [130].

Some of these various *in vitro* oestrogen and antioestrogen studies have revealed tamoxifen-regulated genes that have further potential since they correlated with adverse clinical outcome (Scafoglio et al, 2006) [130], while Hayashi and Yamaguchi (Hayashi et al, 2005) [131] and also Oh *et al.* (Oh et al, 2006) [132] studied oestrogen-regulated genes in cell lines to reveal novel response predictive factors and to develop a gene expression–based survival predictor for *ER+* and/or *PgR+* breast cancer patients respectively. The latter study determined that poor prognosis patients showed increased expression of proliferation/survival genes, while those with good prognosis showed increased oestrogen-regulated gene expression in their breast cancer (Oh et al,

2006) [132]. Finally, using the Affymetrix HGU-133A chip, a better understanding of oestrogen deprivation treatment, using various aromatase inhibitors, again versus antioestrogens, has also been achieved *in vitro* at the molecular level (Itoh et al, 2005) [133], while custom cDNA microarrays are also being used to explore non-classical *ERα* (ERE-independent) target genes, giving much broader insight into *ER* interplay with further novel pathways that may contribute to breast tumor response to SERM therapy (Glidewell-Kenney et al, 2005) [134].

In total, such microarray studies of cell lines during responses to hormones and antihormones are not only aiding our understanding of antihormone mechanism at a transcriptional level, but also revealing possible new avenues to potentially maximize anti-tumour response as well as potential predictive gene signatures. Microarray expression studies, such as those ongoing in the Tenovus Centre, are also being used to study *in vitro* cell line and mouse models that have acquired resistance to the antioestrogens tamoxifen or faslodex, in order to reveal potential signatures (and possibly therapeutic targets, where genes are growth signalling-pathway related) that may be directly associated with various endocrine resistant states (Scott et al, 2007) [135], (Fan et al, 2006) [136], (Hilsenbeck et al, 1999) [137], (Huber et al, 2004) [138], (Sommer et al, 2003) [139], (Gu et al, 2002) [140]. However, to date, such studies have generally focused on signatures of individual forms of resistance, rather than also defining "shared" differential gene expression in the context of resistance to different endocrine agents, where the latter approach may prove particularly powerful in the context of seeking generic biomarkers/signalling targets for multiple forms of resistance. This latter area is of particular interest to this thesis and researchers in the Tenovus Centre.

In vivo material can be a furthermore extremely valuable resource to reveal, explore and subsequently screen predictive gene signatures for endocrine response or failure (Chanrion et al, 2008) [141], (Tozlu-Kara et al, 2007) [142]. For example, identification of a molecular signature predicting the relapse of tamoxifen-treated primary breast cancers using yet another platform – in this instance based on a 70-mer oligonucleotide microarrays, has been described by Chanrion *et al.* to identify a 36 gene molecular signature specifying a subgroup of breast cancer patients who do not gain benefits from tamoxifen treatment. Such a cohort of patients would therefore be eligible for alternative endocrine therapies and/or chemotherapy which would help the therapeutic

management of their estrogen receptor–positive cancers (Chanrion et al, 2008) [141]. A further study screened genes of interest by using a pangenomic 44K oligonucleotide microarray in a series of ten *ER+* tumors comprising five tamoxifen-treated postmenopausal patients who relapsed (distant metastasis) and five who did not relapse, matched with respect to age, tumour grade, lymph node status, and macroscopic tumor size. The study revealed the genes *HRPAP20* and *TIMELESS* as promising markers of tamoxifen resistance in women with ER alpha-positive breast tumors [142]. In some instances such gene signatures are now being examined in larger clinical studies in relation to recurrence during tamoxifen therapy (although to our knowledge not as yet in relation to faslodex failure in the clinic), for example the Oncotype DX and Rotterdam signatures as previously outlined in Chapter 1.

Clearly, microarray approaches are proving extremely popular in the context of understanding endocrine response and resistance, with some significant findings in this regard, in some instances already under test as predictive signatures in the clinic in relation to tamoxifen outcome. However, there are no standard approaches for microarray analysis, although initial experimental protocol now must be described according to MIAME recommendations. This means there is an inherent risk that gene signatures and targets discovered will be difficult to replicate in other studies. Moreover, few applications are freely available to the biological researcher that include database storage, multiple visualization, basic and advanced analysis techniques (e.g. statistical and multiple unsupervised/supervised clustering), with appropriate links to ontological analysis, presented in a comprehensive package. Such a plethora of capabilities are highly-desirable for microarray studies if the differential genes revealed are to ultimately be robust, reproducible and biologically-relevant.

A key aim of this project was thus to develop a platform with such capabilities, to improve and streamline the process whereby a researcher can reveal potential new genetic targets from array portfolios in the context of deciphering endocrine response and failure, in this instance exemplified by the Tenovus MCF7-derived cell line HG-U133A dataset. I-10 was also required to be a user-friendly platform, a critical aspect since the laboratory researcher rarely has the advanced bioinformatic or coding skills required to harness all the analysis techniques and apply them to large datasets. Such advanced analysis capabilities are particularly desirable in the

Tenovus studies that aim to find both "unique" and "shared" differential genes often across multiple replicates and resistance/treatment groups in their *in vitro* datasets. I-10 also incorporates beneficial 2D/3D graphical features not only to enhance its user-friendly format but to permit further advanced data exploration. Importantly, I-10 also allows the researcher to access algorithms capable of assisting choice of the most appropriate clustering method and to prioritize large lists of differential genes, again presenting this analysis to the researcher in a user-friendly manner. This again is highly beneficial, since large gene lists and defining their associated ontology can be daunting in the absence of a systematic prioritization strategy. I-10 should be applicable to multiple types of array data and is also able to be continually upgraded – a feature many commercial applications lack- such that state-of the art advanced analysis methods can be easily and rapidly encompassed for the user. Finally, I-10 is non-commercial, based upon open-source technologies, an important factor where analysis cost needs to be considered.

To demonstrate some of the key abilities of I-10 in the context of endocrine resistance (in this first instance to the SERM tamoxifen and the pure antioestrogen Faslodex), therefore, differential gene expression has been determined and further explored through multiple clustering procedures as applied to the Tenovus TAMR and FASR cell lines in relation to their endocrine responsive counterpart MCF-7. Using advanced data analysis techniques through 'R', thousands of potential genes were examined, prioritised and ultimately reduced to a highly significant, biologically relevant cohort. The application of I-10 has proved to be very useful in discriminating the breadth and patterns of transcriptional impact of resistance and in selecting dominant differentially expressed candidate genes across these multiple experimental groups and their associated replicate samples.

Breast cancer is believed to encompass multiple disease sub-types that have been defined according to transcriptional signature from clinical material using microarray class discovery approaches (Harris et al, 2007) [15] and emerging parallel immunocytochemical landmarks (notably *ER*, *HER2* and cytokeratin patterns (He et al, 2006) [10]. While some heterogeneity between classes is apparent, *ER*+ cancers can broadly be characterised by a "luminal" signature, while *ER*- disease includes the "*HER2*" (i.e. *HER2* amplified), "basal" and "normal" sub-types.

These sub-types appear to have different clinical behaviour, where the luminal group is again broadly associated with superior prognosis versus the "*HER2*" and "basal" sub-types, although within the luminal signature there are not only "A" but also "B" (and possibly "C") classes, where the latter has a somewhat poorer outlook (Sotiriou et al, 2003) [80]. However, it remains largely unexplored how these sub-type signatures are related to antihormonal response, although as stated above further gene signatures such as the Rotterdam tamoxifen response profile, the 21-gene set Oncotype DX (Recurrence Score), and a reported *HOXB13-IL17BR* ratio are promising in the context of predicting tamoxifen outcome, while high genomic grade (measurement of proliferation-related genes) also discriminates luminal breast cancers who will fail on tamoxifen (Sommer et al, 2003) [139]. Equally it remains unknown if breast cancer sub-type is influenced by antihormonal exposure in *ER+* patients or whether a shift in sub-type occurs on acquisition of endocrine resistance.

Using the capabilities of I-10, therefore, the phenotype of the acquired resistant TAMR and FASR models in relation to MCF-7 cells was examined before detailed differential gene expression analysis began, based on the luminal, *HER2*, basal and normal genomic signatures using the classifiers produced by Sorlie *et al* (Sorlie et al, 2003) [79] . This was carried out to assess any phenotypic shifts in sub-type which may have occurred during emergence of resistance to both Tamoxifen and Faslodex. Appendix 2 shows a summary of the presence or absence of the genes and corresponding Affymetrix probes applied to the various MCF7 derived models. In summary, the results generated showed that the MCF7 cell line is a luminal cell line (in agreement with literature Ross and Perou 2001; Lacroix and Leclerq, 2004), where interestingly this phenotype was not lost in resistance, despite gains in proliferative and aggressive cellular behaviour of the resistant models. It will be interesting in the future to explore if further signatures linked to tamoxifen failure clinically equally equate with those of the acquired antioestrogen resistant phenotypes *in vitro*, since several of these incorporate proliferation-related genes in the signature including the 21-gene set Oncotypedx (Recurrence Score) and also "genomic grade" (Sommer et al, 2003) [139].

Thus, there was no transcriptional signature shift to the amplified *HER2* class (although the TAMR model does utilize modestly increased *HER2* signalling as part of its growth mechanism

Knowlden et al, 2005) [143]. Moreover, it was interesting to note that the models were not shifting to a basal phenotype in resistance (since key probes associated with a basal phenotype were absent in all the models), particularly in the context of the FASR cell line. In contrast to the *ER+ / HER2-* parental MCF-7 cell line and the acquired TAMR model, it has recently emerged from studies in the Tenovus Centre that the FASR model has lost its *ER* positivity such that it has acquired an aggressive, *ER-/PR-/HER2-* (i.e."triple negative") phenotype (Nicholson et al, 2005) [21]. While a substantial proportion of triple negative breast cancers clinically have a basal sub-type, a recent study has indicated that a triple negative, non-basal phenotype may be more common than originally appreciated (Bertucci et al, 2008) [144]. The studies in this thesis indicate a triple negative, non-basal phenotype may also be acquired during Faslodex treatment. Importantly, no targeted therapies are yet available for aggressive triple negative disease, and hence the I-10 approach to study genetic signature of Faslodex resistance in this thesis and its findings described to date may prove particularly rewarding in the context of treating the triple negative state, whether it is present *de novo* or develops during treatment.

A Significant Analysis of Microarray (SAM) differential expression filter was then performed accepting a false discovery rate of 10%. 1070 significant genes were revealed. This number encompasses both decreases and increases in gene expression in resistance. This relatively large number of transcriptional changes is likely to underpin the elevated growth and invasion associated with the resistant models. Interestingly, the results showed an apparent enrichment of significant differential expression of genes on chromosomes 7 and 10 in resistance (with less deregulation of genes shown on chromosome 19) than would have been expected from the gene set. Clearly, it is possible that resistance may in part be driven by genetic gains or losses focused to particular chromosomes that occur during treatment. Individual gene amplification or loss have been (controversially) linked with tamoxifen resistance, including amplification of key growth signalling molecules such as *HER2* (Dowsett et al, 2001) [145] and loss/mutation of key tumour suppressor genes including *p53* (Berns et al, 2000) [146], while amplification of several genes at chromosome *11q13* has also recently been linked to Tamoxifen resistance (Bostner et al, 2007) [147].

Among the chromosomal changes described in a further endocrine resistant model from Achuthan *et al.* (Achuthan et al, 2001) [148], there were several alterations occurring at chromosome 7. Further I-10 analysis of the chromosome changes in the Tenovus FASR and TAMR cells revealed that most of the expression changes associated with chromosome 7 was increases in resistance (particularly in the FASR model). However for chromosome 10, while 78% of the changes were again expression increases in FASR cells, 63% of the changes in TAMR cells were expression decreases. Interestingly, gains in chromosome 7 copy (the location of the invasion signalling gene *Met* receptor), and allelic loss in chromosome 10, have both been associated with tumour progression (Bose et al, 1998) [149], (Hirata et al, 1998) [150]. Moreover, a lengthened homogeneous staining region (HSR: a cytogenetic indicator of gene amplification) on chromosome 7 has been described in MCF-7 cells resistant to methotrexate, where chromosomal rearrangements and numerical changes have been linked to adaptation of MCF-7-derived cell lines (Whang-Peng et al, 1983) [151].

An aspect of advanced functionality of I-10 was demonstrated early into analysis by performing principal components analysis upon the 1070 revealed genes. Rotation of the plot through 360 degrees showed a bias towards the FASR arm in general i.e. that there were more significant gene changes in this form of resistance. This is perhaps not surprising since this cell model exhibits more extreme increases in its proliferative and invasive capacity (Perou et al, 2000) [11] than the TAMR model (where both are elevated versus endocrine responsive MCF-7 cells; (Perou et al, 2000) [11]), presumably underpinned by unique or perhaps more extreme transcriptional changes. For example, we have previously noted unique up regulation in expression of the tyrosine kinase gene *Met* (the target receptor for *HGF*/scatter factor) associated with the FASR model which can contribute to its invasive behaviour. Generation of a hierarchical clustering heat map through I-10 subsequently showed that the Control MCF7 and TAMR models clustered with greater association than the FASR model. This is in keeping with the TAMR, like the MCF-7 model, retaining *ER*, where this receptor in TAMR cells remains growth-contributory (albeit through its coupling to a different cross-talk mechanism with *EGFR/HER2* (Britton et al, 2006) [152]). In contrast, the FASR model has lost its *ER* expression, and thereby any oestrogen-regulated signalling and growth, presumably in turn gaining *ER*-independent signalling pathways

(perhaps evident at the gene expression level) that could perhaps promote a different/more extreme transcriptional readout underpinning growth and progression.

Later in the development of I-10, its versatility was demonstrated by ease of integration of a new clustering analysis library – ClValid - which proved a powerful addition to advanced analysis allowing an optimal choice of clustering method and cluster number to be made. This was incorporated in order to remove uncertainty by reinforcing clustering results with statistical values. Through 3 measurements of cluster validation (i.e. internal, stability and also biological validation of clusters vs. Go ontology), ClValid allows the user to choose from different clustering techniques and compare one against the other, testing up to the user-defined number of clusters. Graphs and statistics can be produced at each cluster number with a summary generated showing optimal cluster number. Five different clustering techniques – hierarchical clustering, K-Mean, PAM, Self-organising maps and fuzzy analysis - were chosen which were all integrated into I-10 through 'R'. It was clear that some compromise was required between feasibility of testing cluster number and computing power when using the package. A maximum cluster number of 20 clusters were possible when evaluating the various clustering techniques; however only three different types of clustering could be performed in each analysis cycle, with this number of clusters again due to computational limitations. Consequently, the five clustering techniques were divided into two runs, with hierarchical clustering being performed in each cycle.

Ultimately two and twelve clusters were revealed as optimal across the different clustering techniques – the two clusters revealed were simply the two dominant "up" and "down regulated" clusters, whereas the 12 clusters were of potentially more interest since they focused upon the underlying key patterns of gene expression between the three cell model arms of the analysis. Hierarchical clustering performed consistently throughout the clustering techniques particularly with the biological validation testing, although methods such as PAM and SOM also performed well. To subsequently further explore hierarchical clustering membership, each of the 12 clusters revealed through ClValid were overlayed alongside the heat map. An average profile for each cluster was generated with the profile viewer in I-10 which showed that overall there were seven distinct profiles according to up and down regulation across the two resistant groups in relation to

the MCF-7 cells, with the remaining five being variations on each of the other profiles. Taking fold change of 1 as a cut off in either direction, the dominant profiles were: significantly increased in both TAMR and FASR cells (1 profile); increased in TAMR cells only (2 profiles) or FASR cells only (2 profiles); increased in FASR but decreased in TAMR cells (1 profile); decreased in both TAMR and FASR cells (3 profiles), decreased in TAMR cells only (1 profile) or FASR cells only (2 profiles). No genes fell into the significantly decreased in FASR and increased in TAMR only category. There are thus more possible expression profile variations that are unique to a resistant state (8 profiles) versus those where expression changes are shared in both forms of resistance (4 profiles) that may be particularly relevant in a quest to identify generic targets in the resistant states.

Potentially the overlapping clusters could thus be combined, but as there were sufficient number of probes in each of the 12 individual clusters for separate analysis these were ultimately all analyzed. However, some analysis priority was immediately placed on clusters 12 and 9 as these were the most significant "shared" induced or "shared" suppressed profiles respectively across the resistant states and were thus clearly of substantial potential relevance in the context of defining generic markers or targets for anti-oestrogen resistance. Function of the pvClust module of 'R' in I-10 was then demonstrated, potentially prioritizing those genes within each cluster for study by identifying those that were ultimately the most significant at both a 0.05 and 0.005 p value level, in this instance initially focusing on the obvious shared clusters 9 and 12 in the context of generic suppressed or induced transcriptional events associated with resistance. Subsequent use of the 3D plotting tool through I-10 revealed an extremely reproducible gene profile in the significant genes comprising clusters 9 and clusters 12, implying transcriptional coregulation of expression.

As stated above, Cluster 9 was indicative of genes in FASR and TAMR cells whose expression was suppressed in both models. This cluster contained some suppressed genes that have been shown in a further project in Tenovus to be depleted in TAMR cells but which had not previously been explored in the context of faslodex resistance. These included the carbonic anhydrase 12 gene, where higher levels have previously been associated with better prognosis breast cancer in keeping with the observation here of its expression loss in the aggressive resistant models. Of

novel interest, however, was the shared suppressed Probe *205440_s_at* that persisted in analysis through many levels of GO ontology focusing on receptors (obtained by accessing FATIGO multi-level analysis through I-10). This was subsequently taken forward into DAVID, a further ontological resource linked to I-10, to determine more ontological information.

The probe *205440_s_at*, known as the *NPY Y(1)* receptor (*NPY1R*) which is the neuropeptide Y receptor Y1, is a receptor targeted by ligands previously implicated in neuroendocrine regulation in the nervous system and GI tract. This signalling gene encodes a membrane protein that belongs to the G-protein coupled receptor 1 (rhodposin-like receptor) family and its signaling is reported to induce the expression of *CRE* containing target genes through the *CaM kinase-CREB* pathway, and inhibits *CRE* containing genes when cellular *cAMP* levels are elevated. Recently, a role of neuropeptide Y (*NPY*) in tumor biology was implied based on the high density of *NPY* receptors noted in breast and ovarian cancers. These *NPY* receptors are also a potential new molecular target for the therapy of some tumour types. In the models used in this project, however, *NPY* (1) receptor decreases in resistance suggesting a possible role in driving endocrine responsive breast cancer cells that is subsequently lost in the aggressive, proliferative resistant state during prolonged antihormone exposure. Interestingly, therefore, *NPY1R* has previously been shown to be oestrogen-regulated in MCF-7 cells but to be lost in a aggressive ER- breast cancer cell line, *MDAMB231* . Oncomine clinical transcriptome analysis outlined in chapter 3, available as a link within I-10, revealed that the *NPY* receptor is at a higher expression level in luminal, *ER+*, *PgR+* and *HER2–* clinical breast cancers at the transcriptional level, equating with its enrichment in the MCF-7 cell model. Cumulatively, these findings support a relationship between *NPY1R* and an indolent, potentially endocrine responsive phenotype. Moreover, the findings indicate that the receptor would be worthy of further exploration both as a biomarker for response (i.e. where in turn loss could associate with anti-oestrogen failure in the context of both tamoxifen and faslodex). With regards to its therapeutic targeting, it may also be valuable to pursue this in the context of the responsive state, or possibly to explore impact of re-inducing its expression in resistant cells to see if antihormone response can be restored.

The second cluster examined in detail, cluster 12, comprised TAMR and FASR induced genes. A gene of potential interest was *202412_s_at* – ubiquitin thiolesterase (*USP1*) which is reported to

play a role in DNA repair. Protein ubiquitination and deubiquitination are dynamic processes implicated in the regulation of numerous cellular pathways. Monoubiquitination of the Fanconi anemia (FA) protein *FANCD2* appears to be critical in the repair of DNA damage because many of the proteins that are mutated in FA are required for *FANCD2* ubiquitination. *USP1* is believed to deubiquitinate *FANCD2* when cells exit S phase or recommence cycling after a DNA damage insult and may thus play a critical role in the FA DNA repair pathway by recycling *FANCD2*. An Oncomine search in this thesis showed its presence in high grade *ER-* breast cancer, and it is certainly feasible that increases in enzymes that regulate DNA repair could be of substantial benefit in permitting endocrine resistant progression. This could consequently be characterized as a biomarker of resistance, and reveals endocrine resistant breast cancer could comprise a potential target for therapies impacting on DNA repair mechanisms.

FATIGO analysis also revealed Cluster 12 contained a number of significant genes whose encoded proteins are regulated by transition metal binding – namely zinc previously shown to play a role in breast cancer. Zinc is now known to be important in enhancing activation of several mitogenic tyrosine kinases (including *erbB* receptor family members, *IGF1R* and *c-Src*) and is implicated in endocrine resistant growth and invasion, where the findings in this thesis further confirm our belief that Zinc is a critical player in regulating the resistant phenotype and its progression.

Further interesting observations were made when ontological analysis was performed by implementing FATIGO (and complementing through DAVID ontology searches) through I-10, as applied to all of the 12 clusters individually but subsequently focusing on genes that persisted in analysis through many levels of GO ontology (up to FATIGO level 9). With regards to the clusters where genes were significantly induced only in the Faslodex resistant model (i.e. clusters 1,3 and 8), FATIGO highlighted "hepatocyte growth factor receptor activity" associated with the gene probe for the tyrosine kinase Met (*203510_at*) and "fibroblast growth factor receptor activity" associated with the receptor tyrosine kinases fibroblast growth factor receptor 2 (*203638_s_at*) and 4 (*211237_s_at*). As stated above, we have previously verified Met up regulation at the mRNA level and protein level in the FASR model (Hiscox et al, 2006) [153] and showed relevance of Met receptor to promotion of invasion of these cells, particularly under

conditions where there is paracrine exposure to fibroblasts producing the *Met* ligand *HGF*/scatter factor. Such data highlight the potential for *Met* in the biomarker/therapeutic target context in the faslodex resistant state, and interestingly *Met* antibodies and small molecules inhibitors are emerging in cancer that would be worthy of subsequent testing in this particular breast cancer context (Dussault et al, 2008) [154].

The ability of the I-10 analysis process (and implementation of FATIGO) to also highlight the Met probe reassures that it may comprise an equally effective analysis platform to determine previously unknown targets/biomarkers of significant functional relevance to invasion/proliferation in resistance. The FGFR up regulation identified by FATIGO is thus also of some interest, where RT-PCR studies performed in the Tenovus laboratories in parallel to this thesis have been able to very recently confirm induction of both *FGFR2* and *FGFR4* in the context of FASR cells. Again, this is of interest since *FGFR* inhibitors are emerging and there is also increasing clinical evidence of a role for *FGFR4* in endocrine resistant states (Dussault et al, 2008) [154], while the *FGFR2* gene has been shown to be amplified in 10% of breast cancer patients. This thesis indicates a priority area for testing of *FGFR* inhibitors may be following faslodex relapse and that these receptors could prove to be new markers for this state.

Examination of the clusters where genes were significantly induced only in the tamoxifen resistant model (i.e. clusters 2, 11) revealed these were enriched for genes responsible for ATPase activity and sodium ion binding, and also included "pancreatic ribonuclease activity" referring to the angiogenin gene (*205141_at*) and "zinc ion binding" referring to the peroxisome proliferator-activated receptor gamma gene (*208510_s_at*). Again, the discrimination of *PPARγ* by the I-10 analysis strategy is encouraging since this gene has also been recently verified in the Tenovus laboratories at the RT-PCR level, where manipulation of this receptor has been of some interest in the literature in the context of breast cancer treatment/prevention (Meijer et al, 2008) [155]. The novel identification of angiogenin in the thesis is also of some interest. The encoded protein has been implicated in angiogenesis and further aspects of cancer progression and has been correlated with clinical breast cancer behaviour (Eppenberger et al, 1998) [156], while there is furthermore increasing data supporting a relationship between markers of angiogenesis/hypoxia and endocrine resistant states (Qu et al, 2008) [157]. Future deciphering of the importance of

angiogenin in the TAMR cells and evaluation of angiogenic capacity of the Tenovus resistance models may highlight further biomarker/therapeutic avenues for this state.

The remaining clusters comprised either further clusters suppressed in both forms of resistance (as in cluster 9, i.e. clusters 7 and 10) or in one form of resistance only (either suppressed in FASR cells for clusters 5, 6 or suppressed in TAMR cells for cluster 4). Examining the further clusters bearing genes whose expression was decreased in both forms of resistance versus MCF-7 revealed that cluster 10 included "insulin-like growth factor receptor activity" referring to the *IGF1* receptor gene (203627_at). Endocrine responsive cells like the MCF-7 line are known to express high levels of the *IGF1R* where such signalling interacts closely with ER to promote cell growth. The Tenovus group have previously confirmed a lower level of *IGF1R* expression in the TAMR cells (Knowlden et al, 2005) [143], further reassuring that the I-10 procedure is effective in identifying robust mRNA changes, although interestingly activity of this receptor still remains substantial and growth-relevant in the TAMR cells, re-enforcing the importance of subsequently monitoring protein expression, activation and cellular function when deciphering signalling elements revealed by microarray mRNA studies.

Interestingly, "MAP kinase tyrosine/serine/threonine phosphatase activity", referring to the dual specificity phosphatases 4 and 6 (*DUSP 4* [*204015_s_at*; also known as *MKP2*] and *DUSP6* [*208892_s_at*, also known as *MKP3* or *PYST1*]), was highlighted as decreased at the expression level in clusters 5 and 6 with Faslodex resistance only. As potential phosphatase inactivators of MAP kinases and possible tumour suppressors, loss of such *DUSPS*s could enable increased *Erk1/2* MAP kinase signalling and thereby Faslodex resistant growth (Keyse et al, 2008) [158]. Several MAP kinase activity cascades, including *Erk1/2*, are thought to transmit and amplify signals involved in cell proliferation and cell survival. Such signal transduction pathways can be induced by various growth factor, steroid hormone and G protein receptor-mediated ligands as well as being environmental stress-activated. MAP kinase pathways can also exert cross-talk effects in *ER+* cells at the level of *ER*-induced transcription as well as at the level of the cell cycle. Recent studies have shown that some tamoxifen resistant breast cancer cells contain increased activation of MAP kinase family members, notably including *p38* (Gutierrez et al, 2005) [159], (Knowlden et al, 2003)[160] and the *Erk1/2* MAP kinases including in endocrine

resistant breast cancer models such as TAMR cells (Knowlden et al, 2003) [160]. A further study revealed that while *DUSP6/MKP3* expression level was increased in response to tamoxifen as a potential counterbalance to the increasing *Erk1/2* MAP kinase activity, in tamoxifen resistant cells the activity of *MKP3* was markedly decreased, allowing MAPK hyperactivation (Cui et al, 2006) [161]. Coupled with our observations here of a *MKP3* expression decline in FASR cells, such data suggest multiple regulation of *MKP3* at both the expression and activity level contributing to various endocrine resistant states. The observations made for the *DUSPs* indicate further deciphering of MAPK pathways is needed and may lead to novel biomarkers and targeting strategies for endocrine resistant cells.

Finally, FATIGO+ was used to further compare the broad anthologies of the two gene lists for cluster 9 and 12, by comparing each list against the other according to GO ontology class as shown previously for individual clusters. Results showed no ontological information present at levels 1 and 2, and only non-significant trends at level 3, where cluster 9 had more genes associated with "cell developmental processes" in its suppressed cluster, as opposed to more genes for "cellular metabolism" in the shared induced cluster 12. In this instance, the gene expression data revealed through I-10 no doubt reflect increased metabolic needs associated with highly proliferate, invasive resistant states. A further significance was recorded at level 7 in this FATIGO+ ontology analysis, where "programmed cell death" was highlighted as a more prominent ontology for some genes within the suppressed cluster 9. Again, this would be in keeping with a shift towards increased proliferation and cell survival in resistance relative to the more indolent behaviour of endocrine responsive cells such as MCF-7.

6.3 Aim 3 - • Development of a cancer patient covariate exploration analysis tool to investigate the impact of multiple prognostic factors on survival using breast and colorectal cancer patient data from a published dataset.

To better understand what factors affect survival for a cancer patient and optimise or even decide the treatment they obtain is the goal of any oncologist. This motivation gave rise to the development of the Nottingham prognostic index where many patients shared similar characteristics which resulted in a ranked survival outcome (Galea et al, 1992) [7].

The SEER dataset, which contains cancer patient data from 1972 through to 2002, was utilised to providing a high quality data set representing patients who have had different types of cancer – particularly breast and colorectal cancer (Ries et al, 2005) [100]. The coding system used to describe each patient attribute was transformed so it could be queried in a high throughput manner in an automated system. The modified patient dataset was stored in a Microsoft SQL database.

To explore the dataset, a cancer survival query tool – Superstes – was developed. Although Superstes was built on the experience gained using Visual Basic, development switched to producing a web based tool as opposed to locally installed Windows application such as I-10. The tool also used updated Visual Basic technology – Visual Basic.net which the installation issues observed that I-10 faced when installing on new computers. Usage of web service technology within Superstes provides a framework to allow the transformed SEER dataset to be shared with other research groups. Very few studies have transformed the SEER patient dataset to harness the power and information the wealth of over 20 patient attributes used in Superstes provides. Hopefully it will provide an example to other research groups as to how such analysis platforms can be built and used to explore cancer patient datasets.

One key aspect is the ability of Superstes to directly compare two patient cohorts directly. The technical frame work under which Superstes was developed, allows not only other cancer types to be queried – as in the case of colorectal cancer in addition to breast caner – however also to potentially include datasets from different international sources. However due to the protective mechanisms in place and health policy differences in terms of usage of data between the United Kingdom and the USA, the UK is behind the USA in regard to data sharing. However initiatives such as the Health Information Research Unit for Wales (HIRU) are aiming to set a precedent in collecting information from many different medical sources ranging from GP surgeries to NHS trusts to address the shortfall of available patient information [HIRU, 2008) [162].

Many different patient attribute combinations can be searched using Superstes. Four different examples were outlined in Chapter 4 to demonstrate how results were returned from single and two cohort patient groups. Logistic regression analysis has also shown to be informative in

ascertaining the significance a particular patient attribute has on survival. When a two cohort colon cancer search was performed, number of nodes positive of a tumour was shown to have a highly significant role in influencing survival having a p-value of 0.02. In future, altering the patient search query perhaps directed by an oncologist's knowledge or by literature would discover what combination of covariates are optimal in contribution to patient survival. A particular set of covariates could then analysed further using more advanced modelling techniques to fully explore their impact on survival.

The SEER dataset is widely acknowledged for its data quality (Ries et al, 2005) [100]. However the nature of Superstes could potentially show particular bias in some patient covariates. For example, minority ethnic groups are of particular interest to the SEER programme and are therefore somewhat over represented in the database. Therefore confirmation or further insight might be revealed when studying histograms of race from the dataset or using race in assessing modelling results using Superstes. A previous study by Li *et al* in 2003 examined the affect of race on breast cancer survival using the SEER breast cancer dataset. The study highlighted disparities in breast cancer diagnosis, treatment, and survival among American women from a wide-range of racial and ethnic backgrounds. The study highlighted that 50% of women from Puerto Rico were more likely to receive substandard, inappropriate treatment for breast cancer. Japanese and Chinese women had better survival rates after breast cancer while Hawaiian and Mexican women had 30% poorer survival rates when compared to non-Hispanic whites. African American, Native American, and Hispanic white women faced a 10% to 70% greater risk of dying after a breast cancer diagnosis as compared to non-Hispanic whites. Consequently, Superstes will allow greater understanding of this phenomenon with the added benefit of examining other patient attributes simultaneously. Due to Superstes enabling query of many different patient attributes from the SEER dataset, it will prove to offer terrific insight into patient survival for clinicians wishing to explore many different hypotheses, not just race alone.

Superstes demonstrates how modelling of survival – such as using the Cox proportional hazards model – is able to show how much of a significant contribution a particular patient attribute had on survival based on previous patient histories, outlined in the SEER dataset. When different machine learning techniques were evaluated in terms of predicting survival outcome in Chapter

5, the random forest method performed the best overall in the evaluation of different machine learning techniques. Consequently it is plausible to conclude that it would be the most reliable technique in measuring the resulting prognostic impact. It would be very important for an oncologist, on the basis of patient history data, to assure a particular patient has relatively benign condition and that their treatment could be easily managed possibly without treatment and therefore have a good overall survival prognosis. This fundamentally has a crucial importance to a patient's standard of living as the potential of developing side effects as a result of engaging a patient onto a particular treatment regime, may outweigh the benefits of such treatment, particularly if their condition is not as critical as that of others.

6.5 Aim 4 – Usage of advanced computational methodologies in predicting patient survival and determination of which method offers the most robust classification using the same patient attributes as used to predict prognosis using the NPI

Multivariate analysis has been long established using high throughput approaches such as microarray analysis *in vitro* models representative of different cancer types. Historically, clinicians have researched pathological factors and their affect on patient survival has been based more upon their own experience and search of the literature. More recently however, multivariate analysis studies have shown that different methodologies can be used to analyse the multitude of patient attribute information simultaneously and to learn tends in population, thus expanding the "localised" knowledge to a more "global" knowledge, which can then be accessed by other clinicians.

Chapter 5 demonstrated that many different machine learning methodologies could be applied to a cohort of breast cancer patients to model survival based on patient attributes of nodes positive status, tumour size and tumour grade. These covariates were the same as that used to predict prognosis using the NPI. An assumption was made that these three attributes could be used to predict survival outcome in the training data set for the machine learning methods to learn successfully; otherwise the performance of such techniques would be low.

The versatility of 'R' was further demonstrated in the way that it can interface with external specialised applications such as machine learning applications. The Java programming language application 'Weka' is a typical example of such an application. Using six different machine learning methodologies, a comparison of the accuracy of a classifier produced using each method was achieved. The better the model emulates what processes are occurring in the patient which result in a particular outcome based on certain patient attributes, the greater the accuracy the classifier will produce.

The fact of defining a 'typical' representative cancer patient cohort is difficult, if it even exists. The SEER dataset comprises of population-based cancer registries from a state wide, metropolitan area or rural county grouping via the National Cancer Institute (NCI) for inclusion in the SEER database. The cancer patient data is collected from health providers which range from hospitals to physician offices as well as from autopsy reports and death certificates. The SEER Program data is considered the international standard for cancer registry data quality. Consequently the patient data set which the modelling techniques utilises can be used with confidence.

Most of the benefits observed with boosting appeared to be caused by over-fitting the training data set. Although boosting is thought to generally increase accuracy, it is well documented that it leads to deterioration of classifier results with some datasets.

More understanding of attribute interactions occurring within each classifier is required in addition to being able to accurately predict an outcome. Information regarding the relationship between the attributes and their outcome when applied to a dataset is required – this can be thought of as looking inside the 'black box'. Data modellers have criticized the machine learning efforts on the grounds that the accurate predictors constructed are so complex that it is nearly impossible to use them to get insights into the underlying structure of the data. They are often thought of as large bulky incoherent single purpose methodologies. The contrary is true using decision tree analyses such as random forests. More reliable information about the inside of the 'black box' is obtained than using that revealed using the other machine learning techniques.

However the disadvantage is that it is not in the form of a simple equation, which makes routine usage more prohibitive for clinicians, for example.

The machine learning techniques show how detailed analysis of patient covariates can be assessed. Ten fold cross validation has shown which classifier performs the best using the SEER breast cancer patient cohort and using patient covariates known to have a connection with survival – namely the NPI index. It is therefore plausible that in conjunction with a tool such as Superstes, patient covariates revealed from patient searches in Superstes could be applied to the SEER dataset to determine if an altered selection of covariates improves upon the patient covariates of tumour size, grade and nodes positive status. Consequently this would encourage a clinician to generate new hypotheses and thus aim to improve the standard of current diagnostic and prognostic processes.

Ten-fold cross validation, apart from providing an overall accuracy measure, produces a Kappa statistic value for each methodology evaluated. As introduced in Chapter 5, this enables a measure of success classification of a technique. A value of 1 indicates perfect classification, and a value of 0 indicates classification at the chance rate i.e: the null hypothesis – the closer to 0 the poorer the classifier has performed. Using this value as a measure, none of the machine learning techniques performed exceptionally well. In keeping with the objectives of the study, determining which classifier performed best with when applied to the breast cancer cohort, multiple sources of information from ten fold cross validation analysis indicates that the random forest classifier performs the best of all the methods. It had the highest accuracy and also the highest kappa value of 0.501, for example. Bagging performed the worst when applied to a test set of 200 patients resulting in a score of 0.350. This performance was mirrored in the overall accuracy scores which ten fold validation summarised. However it is very easy to forget when using advanced modelling techniques exactly what the rationale for applying such systems in the first place. This is particularly important when predicting survival on a routine basis as ultimately the cost of misclassifying patients as having a good prognosis when in fact they are of a high risk of death is the ultimate mistake to make.

As models varied in performance, optimisation was needed – particularly in the case of support vector machines – and therefore their routine use for predicting survival outcome should be viewed very sceptically for day to day prognosis. As previously described in Chapter 1 for microarray classifiers, it is very easy to make bold statements regarding performance of a particular classifier; however it can be difficult to recreate similar success with a different data source. Currently, more in depth study into optimising patient covariate combinations in terms of predicting survival is required.

However when presented with new 'fashionable' machine learning techniques, it is very easy to undervalue more traditional statistical techniques such as logistic regression. Logistic regression performed well against the SEER dataset especially considering its relative simplicity to more advanced machine learning methods. Furthermore it has the advantage that the same result is produced each time based on a particular result. However this differs to the machine learning methods as because of the way in which they use random methods for sampling as part of their 'learning' processes against the dataset, the result will be slightly different each time. Oncologists are busy people and therefore they might not have the time or value the extra effort in building the machine learning models as a result of the lack of consistency in the results produced. Furthermore as the results produced were not overwhelmingly accurate, their value would have to be carefully considered.

It would be interesting to compare the different methodologies against different cancer datasets to determine if performance by any particular method is better against certain datasets or similar in performance overall. This could be achieved by potentially comparing the accuracy percentages obtained during ten fold cross validation against each classifier performance in turn. Until this is performed, it is difficult to categorically say one machine learning approach is universally better than another.

Many of the machine learning techniques – particularly Support Vector Machine (SVM) - take an extended time to process. In fact, SVM took over an hour to perform just one cycle of the algorithm. Due to the optimisation required for the SVM algorithm, results will take many hours – perhaps even days to return meaningful results with no guarantee of success. Such techniques are best suited to a super computing environment. In fact the very nature of the way classifiers

are built using some of the machine learning techniques borrows itself very well to parallel processing. Parallel processing involves a particular task – e.g: iterations of an algorithm – to be distributed among different individual computers connected together so each has an individual task to process. This greatly speeds up the running of many of the machine learning algorithms as multiple computers are working in 'parallel' with each other. Once complete, each machine returns a result and the output combined. These results in far higher throughput and analysis of datasets – a task which 'R' is not optimised to perform. In fact studies were conducted to re-compile 'R' for it to perform tasks in a super computing environment however the task ultimately was beyond the scope of this project. It would be interesting to see this achieved in a future study.

6.6 Conclusion

Freely-available programming languages and design tools have enabled the new tools outlined in this project to be created. Their usefulness has subsequently been successfully demonstrated in the context of two specific areas of current interest in breast cancer: to better understand adverse endocrine resistant states (revealing some new potential biomarkers/possible therapeutic avenues), and to be able to interrogate clinical data sets in order to derive improved classifiers for patient survival. It is acknowledged that freely available tools, particularly data analysis tools, are often difficult for some users not familiar with computer programming environments to apply. Consequently development of automated, user-friendly tools such as I-10 and Superstes will hopefully introduce many more researchers to the power of freely available technologies such as the 'R' project to achieve their bioinformatics needs.

However it is also acknowledged that development of an automated system – for example, a user interface similar to Superstes – is highly unlikely for machine learning methodologies. Due to their highly-specialised nature that requires optimisation, and potentially differing performance of machine learning methodologies between datasets, this would prove difficult to perform automatically whilst still remaining informative and user friendly, especially to a user unfamiliar with machine learning.

It is hoped that the analysis approaches and tools developed in this study will ultimately help many researchers gain a broader understanding of cancer biology in both model systems and the clinical setting. There is much hope that usage of the strategies and tools outlined in this project should expedite better understanding of results derived from high throughput technologies and large datasets which will ultimately drive cancer research forward.

References

1 Office for National Statistics. **Registrations of cancer diagnosed in 1993-1996, England and Wales.** Health Statistics Quarterly 1999; 04:59-70.

2 Thompson A, Brennan K, Cox A, Gee J, Harcourt D, Harris A, Harvie M, Holen I, Howell A, Nicholson R, Steel M, Streuli C: **Evaluation of the current knowledge limitations in breast cancer research: a gap analysis.** *Breast Cancer Res* 2008, **10**:R26.

3 Rakha EA, El-Sayed ME, Reis-Filho JS, Ellis IO: **Expression profiling technology: its contribution to our understanding of breast cancer.** *Histopathology* 2008, **52**:67-81.

4 Yaffe MJ: **Mammographic density. Measurement of mammographic density.** *Breast Cancer Res* 2008, **10**:209.

5 Kapp AV, Jeffrey SS, Langerød A, Børresen-Dale A, Han W, Noh D, Bukholm IRK, Nicolau M, Brown PO, Tibshirani R: **Discovery and validation of breast cancer subtypes.** *BMC Genomics* 2006, **7**:231.

6 Slamon DJ, Clark GM, Wong SG, Levin WJ, Ullrich A, McGuire WL: **Human breast cancer: correlation of relapse and survival with amplification of the HER-2/neu oncogene.** *Science* 1987, **235**:177-182.

7 Galea MH, Blamey RW, Elston CE, Ellis IO: **The Nottingham Prognostic Index in primary breast cancer.** *Breast Cancer Res Treat* 1992, **22**:207-219.

8 Yu K, Lee CH, Tan PH, Hong GS, Wee SB, Wong CY, Tan P: **A molecular signature of the Nottingham prognostic index in breast cancer.** *Cancer Res* 2004, **64**:2962-2968.

9 Feng Y, Sun B, Li X, Zhang L, Niu Y, Xiao C, Ning L, Fang Z, Wang Y, Zhang L, Cheng J, Zhang W, Hao X: **Differentially expressed genes between primary cancer and paired lymph node metastases predict clinical outcome of node-positive breast cancer patients.** *Breast Cancer Res Treat* 2007, **103**:319-329.

10 He P, Xiao K, Li X, Zhou L, Lu HF: **[Clinical significance of lymph vessel density marked by lymphatic ressel endothelial hyaluronic acid receptor-1 in laryngeal squamous cell carcinomas].** *Lin Chuang Er Bi Yan Hou Ke Za Zhi* 2006, **20**:828-30, 833.

11 Perou CM, Sørlie T, Eisen MB, van de Rijn M, Jeffrey SS, Rees CA, Pollack JR, Ross DT, Johnsen H, Akslen LA, Fluge O, Pergamenschikov A, Williams C, Zhu SX, Lønning PE, Børresen-Dale AL, Brown PO, Botstein D: **Molecular portraits of human breast tumours.** *Nature* 2000, **406**:747-752.

12 Modlich O, Prisack H, Bojar H: **Breast cancer expression profiling: the impact of microarray testing on clinical decision making.** *Expert Opin Pharmacother* 2006, 7:2069-2078.

13 Gnant M, Mlineritsch B, Luschin-Ebengreuth G, Kainberger F, Kässmann H, Piswanger-Sölkner JC, Seifert M, Ploner F, Menzel C, Dubsky P, Fitzal F, Bjelic-Radisic V, Steger G, Greil R, Marth C, Kubista E, Samonigg H, Wohlmuth P, Mittlböck M, Jakesz R: **Adjuvant endocrine therapy plus zoledronic acid in premenopausal women with early-stage breast cancer: 5-year follow-up of the ABCSG-12 bone-mineral density substudy.** *Lancet Oncol* 2008, **9**:840-849.

14 Dahabreh IJ, Linardou H, Siannis F, Fountzilas G, Murray S: **Trastuzumab in the adjuvant treatment of early-stage breast cancer: a systematic review and meta-analysis of randomized controlled trials.** *Oncologist* 2008, **13**:620-630.

15 Harris LN, You F, Schnitt SJ, Witkiewicz A, Lu X, Sgroi D, Ryan PD, Come SE, Burstein HJ, Lesnikoski B, Kamma M, Friedman PN, Gelman R, Iglehart JD, Winer EP: **Predictors of resistance to preoperative trastuzumab and vinorelbine for HER2-positive early breast cancer.** *Clin Cancer Res* 2007, **13**:1198-1207.

16 Belkhiri A, Dar AA, Peng D, Razvi MH, Rinehart C, Arteaga CL, El-Rifai W: **Expression of t-DARPP Mediates Trastuzumab Resistance in Breast Cancer Cells.** *Clin Cancer Res* 2008, :.

17 van 't Veer LJ, Dai H, van de Vijver MJ, He YD, Hart AAM, Mao M, Peterse HL, van der Kooy K, Marton MJ, Witteveen AT, Schreiber GJ, Kerkhoven RM, Roberts C, Linsley PS, Bernards R, Friend SH: **Gene expression profiling predicts clinical outcome of breast cancer.** *Nature* 2002, **415**:530-536.

18 Edwards BK, Brown ML, Wingo PA, Howe HL, Ward E, Ries LAG, Schrag D, Jamison PM, Jemal A, Wu XC, Friedman C, Harlan L, Warren J, Anderson RN, Pickle LW: **Annual report to the nation on the status of cancer, 1975-2002, featuring population-based trends in cancer treatment.** *J Natl Cancer Inst* 2005, **97**:1407-1427.

19 Lisztwan J, Pornon A, Chen B, Chen S, Evans DB: **The aromatase inhibitor letrozole and IGF-IR inhibitors synergistically induce apoptosis in in vitro models of estrogen-dependent breast cancer.** *Breast Cancer Res* 2008, **10**:R56.

20 Frasor J, Chang EC, Komm B, Lin C, Vega VB, Liu ET, Miller LD, Smeds J, Bergh J, Katzenellenbogen BS: **Gene expression preferentially regulated by tamoxifen in breast cancer cells and correlations with clinical outcome.** *Cancer Res* 2006, **66**:7334-7340.

21 Nicholson RI, Hutcheson IR, Hiscox SE, Knowlden JM, Giles M, Barrow D, Gee JMW: **Growth factor signalling and resistance to selective oestrogen receptor modulators**

and pure anti-oestrogens: the use of anti-growth factor therapies to treat or delay endocrine resistance in breast cancer. *Endocr Relat Cancer* 2005, **12 Suppl 1**:S29-36.

22 Ross JS, Hatzis C, Symmans WF, Pusztai L, Hortobágyi GN: **Commercialized multigene predictors of clinical outcome for breast cancer.** *Oncologist* 2008, **13**:477-493.

23 Fei R, Shaoyang L: **Combination antigene therapy targeting c-myc and c-erbB(2) in the ovarian cancer COC(1) cell line.** *Gynecol Oncol* 2002, **85**:40-44.

24 Brown PO, Botstein D: **Exploring the new world of the genome with DNA microarrays.** *Nat Genet* 1999, **21**:33-37.

25 Robinson MD, Speed TP: **A comparison of Affymetrix gene expression arrays.** *BMC Bioinformatics* 2007, **8**:449.

26 Affymetrix: *Affymetrix Microarray Suite User Guide.* Affymetrix; 2001.

27 Black MA, Doerge RW: **Calculation of the minimum number of replicate spots required for detection of significant gene expression fold change in microarray experiments.** *Bioinformatics* 2002, **18**:1609-1616.

28 Pedotti P, 't Hoen PAC, Vreugdenhil E, Schenk GJ, Vossen RH, Ariyurek Y, de Hollander M, Kuiper R, van Ommen GJB, den Dunnen JT, Boer JM, de Menezes RX: **Can subtle changes in gene expression be consistently detected with different microarray platforms?.** *BMC Genomics* 2008, **9**:124.

29 Brazma A, Hingamp P, Quackenbush J, Sherlock G, Spellman P, Stoeckert C, Aach J, Ansorge W, Ball CA, Causton HC, Gaasterland T, Glenisson P, Holstege FC, Kim IF, Markowitz V, Matese JC, Parkinson H, Robinson A, Sarkans U, Schulze-Kremer S, Stewart J, Taylor R, Vilo J, Vingron M: **Minimum information about a microarray experiment (MIAME)-toward standards for microarray data.** *Nat Genet* 2001, **29**:365-371.

30 Lee M, Xiang CC, Trent JM, Bittner ML: **Performance characteristics of 65-mer oligonucleotide microarrays.** *Anal Biochem* 2007, **368**:70-78.

31 Lim WK, Wang K, Lefebvre C, Califano A: **Comparative analysis of microarray normalization procedures: effects on reverse engineering gene networks.** *Bioinformatics* 2007, **23**:i282-8.

32 Hubbell E, Liu W, Mei R: **Robust estimators for expression analysis.** *Bioinformatics* 2002, **18**:1585-1592.

33 Irizarry RA, Hobbs B, Collin F, Beazer-Barclay YD, Antonellis KJ, Scherf U, Speed TP: **Exploration, normalization, and summaries of high density oligonucleotide array probe level data.** *Biostatistics* 2003, **4**:249-264.

34 Wu Z, Irizarry R, Gentleman R, Murillo F, Spencer F: **A Model Based Background Adjustment for Oligonucleotide Expression Arrays.** *Dept. of Biostatistics Working Papers* 2004.

35 Li C, Wong WH: **Model-based analysis of oligonucleotide arrays: expression index computation and outlier detection.** *Proc Natl Acad Sci U S A* 2001, **98**:31-36.

36 Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns.** *Proc Natl Acad Sci U S A* 1998, **95**:14863-14868.

37 Thomas JG, Olson JM, Tapscott SJ, Zhao LP: **An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles.** *Genome Res* 2001, **11**:1227-1236.

38 Tusher VG, Tibshirani R, Chu G: **Significance analysis of microarrays applied to the ionizing radiation response.** *Proc Natl Acad Sci U S A* 2001, **98**:5116-5121.

39 Loots GG, Chain PSG, Mabery S, Rasley A, Garcia E, Ovcharenko I: **Array2BIO: from microarray expression data to functional annotation of co-regulated genes.** *BMC Bioinformatics* 2006, **7**:307.

40 Reimers M, Heilig M, Sommer WH: **Gene discovery in neuropharmacological and behavioral studies using Affymetrix microarray data.** *Methods* 2005, **37**:219-228.

41 Quackenbush J: **Computational analysis of microarray data.** *Nat Rev Genet* 2001, **2**:418-427.

42 Reis-Filho JS, Westbury C, Pierga J: **The impact of expression profiling on prognostic and predictive testing in breast cancer.** *J Clin Pathol* 2006, **59**:225-231.

43 Kaufman L: *Finding Groups in Data. An Introduction to Cluster Analysis.* Wiley; 1990.

44 Hartigan A: *Applied Statistics.* ; 1979.

45 Du Z, Wang Y, Ji Z: **PK-means: A new algorithm for gene clustering.** *Comput Biol Chem* 2008, :.

46 Bozinov D, Rahnenführer J: **Unsupervised technique for robust target separation and analysis of DNA microarray spots through adaptive pixel clustering.** *Bioinformatics* 2002, **18**:747-756.

47 Kohonen T, Kaski S, Lagus K, Salojarvi J, Honkela J, Paatero V, Saarela A: **Self organization of a massive document collection**. *IEEE Trans Neural Netw* 2000, **11**:574-585.

48 Hathaway RJ, Bezdek JC: **Fuzzy c-means clustering of incomplete data**. *IEEE Trans Syst Man Cybern B Cybern* 2001, **31**:735-744.

49 Rogers, S; Williams, R; Campbell, C. **Bioinformatics using computational intelligence paradigms**. Springer; Cambridge (MA): 2005. *Class prediction with microarray datasets*. pp. 119–42.

50 Wang A: *Gene selection for microarray data analysis using principal component analysis.*

51 Yu T, Ye H, Chen Z, Ziober BL, Zhou X: **Dimension reduction and mixed-effects model for microarray meta-analysis of cancer**. *Front Biosci* 2008, **13**:2714-2720.

52 Rencher AC, **Methods of Multivariate Analysis (Second Edition)**. John Wiley & Sons; 2003

53 **Mathworks Inc, 1994-2009. A Demonstration Of Multidimensional Scaling [http://www.mathworks.com/products/demos/statistics/mdscaledemo.html]**

54 Dennis GJ, Sherman BT, Hosack DA, Yang J, Gao W, Lane HC, Lempicki RA: **DAVID: Database for Annotation, Visualization, and Integrated Discovery**. *Genome Biol* 2003, **4**:P3.

55 Al-Shahrour F, Minguez P, Tárraga J, Medina I, Alloza E, Montaner D, Dopazo J: **FatiGO +: a functional profiling tool for genomic data. Integration of functional annotation, regulatory motifs and interaction data with microarray experiments.** *Nucleic Acids Res* 2007, **35**:W91-6.

56 Oeder S, Mages J, Flicek P, Lang R: **Uncovering information on expression of natural antisense transcripts in Affymetrix MOE430 datasets**. *BMC Genomics* 2007, **8**:200.

57 Mlecnik B, Scheideler M, Hackl H, Hartler J, Sanchez-Cabo F, Trajanoski Z: **PathwayExplorer: web service for visualizing high-throughput expression data on biological pathways**. *Nucleic Acids Res* 2005, **33**:W633-7.

58 Joshua AM, Boutros PC: **Web-based resources for clinical bioinformatics**. *Methods Mol Med* 2008, **141**:309-329.

59 Dahlquist KD: **Using GenMAPP and MAPPFinder to view microarray data on biological pathways and identify global trends in the data**. *Curr Protoc Bioinformatics* 2004, **Chapter 7**:Unit 7.5.

60 **Online Electronic Journal Search Tool- NCBI** [http://www.ncbi.nlm.nih.gov/pubmed/]

61 **Genecards** [http://www.genecards.org/]

62 Chen H, Sharp BM: **Content-rich biological network constructed by mining PubMed abstracts**. *BMC Bioinformatics* 2004, **5**:147.

63 Wang L, Chen G, Lu D, Chiang H, Xu Z: **[Global gene response to GSM 1800 MHz radiofrequency electromagnetic field in MCF-7 cells]**. *Zhonghua Yu Fang Yi Xue Za Zhi* 2006, **40**:159-163.

64 Korenberg M: *Interpreting Microarray Results With Gene Ontology and MeSH* . Humana Press; 2007.

65 Al-Shahrour F, Carbonell J, Minguez P, Goetz S, Conesa A, Tárraga J, Medina I, Alloza E, Montaner D, Dopazo J: **Babelomics: advanced functional profiling of transcriptomics, proteomics and genomics experiments**. *Nucleic Acids Res* 2008, **36**:W341-6.

66 Fröhlich H, Fellmann M, Sültmann H, Poustka A, Beißbarth T: **Predicting Pathway Membership via Domain Signatures**. *Bioinformatics* 2008, :.

67 Bairoch A, Apweiler R: **The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1998**. *Nucleic Acids Res* 1998, **26**:38-42.

68 **Online Microarray Analysis Suite From Vizxlabs** [http://www.genesifter.net/]

69 **Dmt Application From Affymetrix**
[http://www.affymetrix.com/products/software/specific/dmt.affx]

70 Simon R, Peng Lam A.**BRB Array-Tools Users Guide (version 3.7)**
[http://linus.nci.nih.gov/BRB-ArrayTools.html]

71 (Cheng et al, 2003). Cheng Li and Wing Hung Wong (2003) **DNA-Chip Analyzer (dChip). In The analysis of gene expression data: methods and software.** Edited by G Parmigiani, ES Garrett, R Irizarry and SL Zeger. Springer, New York. 120-141

72 Bioconductor Core: **An Overview of Projects in Computing for Genomic Analysis Biocore Technical Report 1**. 2002, :.

73 Dessau RB, Pipper CB: **["R"--project for statistical computing]**. *Ugeskr Laeger* 2008, **170**:328-330.

74 Sørlie T, Perou CM, Tibshirani R, Aas T, Geisler S, Johnsen H, Hastie T, Eisen MB, van de Rijn M, Jeffrey SS, Thorsen T, Quist H, Matese JC, Brown PO, Botstein D, Eystein Lønning P, Børresen-Dale AL: **Gene expression patterns of breast carcinomas**

distinguish tumor subclasses with clinical implications. *Proc Natl Acad Sci U S A* 2001, **98**:10869-10874.

75 Calza S, Hall P, Auer G, Bjöhle J, Klaar S, Kronenwett U, Liu ET, Miller L, Ploner A, Smeds J, Bergh J, Pawitan Y: **Intrinsic molecular signature of breast cancer in a population-based cohort of 412 patients.** *Breast Cancer Res* 2006, **8**:R34.

76 Mullins M, Perreard L, Quackenbush JF, Gauthier N, Bayer S, Ellis M, Parker J, Perou CM, Szabo A, Bernard PS: **Agreement in breast cancer classification between microarray and quantitative reverse transcription PCR from fresh-frozen and formalin-fixed, paraffin-embedded tissues.** *Clin Chem* 2007, **53**:1273-1279.

77 Makretsov NA, Huntsman DG, Nielsen TO, Yorida E, Peacock M, Cheang MCU, Dunn SE, Hayes M, van de Rijn M, Bajdik C, Gilks CB: **Hierarchical clustering analysis of tissue microarray immunostaining data identifies prognostically significant groups of breast carcinoma.** *Clin Cancer Res* 2004, **10**:6143-6151.

78 Charafe-Jauffret E, Ginestier C, Monville F, Finetti P, Adélaïde J, Cervera N, Fekairi S, Xerri L, Jacquemier J, Birnbaum D, Bertucci F: **Gene expression profiling of breast cell lines identifies potential new basal markers.** *Oncogene* 2006, **25**:2273-2284.

79 Sorlie T, Tibshirani R, Parker J, Hastie T, Marron JS, Nobel A, Deng S, Johnsen H, Pesich R, Geisler S, Demeter J, Perou CM, Lønning PE, Brown PO, Børresen-Dale A, Botstein D: **Repeated observation of breast tumor subtypes in independent gene expression data sets.** *Proc Natl Acad Sci U S A* 2003, **100**:8418-8423.

80 Sotiriou C, Neo S, McShane LM, Korn EL, Long PM, Jazaeri A, Martiat P, Fox SB, Harris AL, Liu ET: **Breast cancer classification and prognosis based on gene expression profiles from a population-based study.** *Proc Natl Acad Sci U S A* 2003, **100**:10393-10398.

81 Perreard L, Fan C, Quackenbush JF, Mullins M, Gauthier NP, Nelson E, Mone M, Hansen H, Buys SS, Rasmussen K, Orrico AR, Dreher D, Walters R, Parker J, Hu Z, He X, Palazzo JP, Olopade OI, Szabo A, Perou CM, Bernard PS: **Classification and risk stratification of invasive breast carcinomas using a real-time quantitative RT-PCR assay.** *Breast Cancer Res* 2006, **8**:R23.

82 Chanrion M, Fontaine H, Rodriguez C, Negre V, Bibeau F, Theillet C, Hénaut A, Darbon J: **A new molecular breast cancer subclass defined from a large scale real-time quantitative RT-PCR study.** *BMC Cancer* 2007, 7:39.

83 Abd El-Rehim DM, Ball G, Pinder SE, Rakha E, Paish C, Robertson JFR, Macmillan D, Blamey RW, Ellis IO: **High-throughput protein expression analysis using tissue microarray technology of a large well-characterised series identifies biologically distinct classes of breast cancer confirming recent cDNA expression analyses.** *Int J Cancer* 2005, **116**:340-350.

84 Korsching E, Packeisen J, Agelopoulos K, Eisenacher M, Voss R, Isola J, van Diest PJ, Brandt B, Boecker W, Buerger H: **Cytogenetic alterations and cytokeratin expression patterns in breast cancer: integrating a new model of breast differentiation into cytogenetic pathways of breast carcinogenesis.** *Lab Invest* 2002, **82**:1525-1533.

85 Diallo-Danebrock R, Ting E, Gluz O, Herr A, Mohrmann S, Geddert H, Rody A, Schaefer K, Baldus SE, Hartmann A, Wild PJ, Burson M, Gabbert HE, Nitz U, Poremba C: **Protein expression profiling in high-risk breast cancer patients treated with high-dose or conventional dose-dense chemotherapy.** *Clin Cancer Res* 2007, **13**:488-497.

86 Farmer P, Bonnefoi H, Becette V, Tubiana-Hulin M, Fumoleau P, Larsimont D, Macgrogan G, Bergh J, Cameron D, Goldstein D, Duss S, Nicoulaz A, Brisken C, Fiche M, Delorenzi M, Iggo R: **Identification of molecular apocrine breast tumours by microarray analysis.** *Oncogene* 2005, **24**:4660-4671.

87 Bild AH, Yao G, Chang JT, Wang Q, Potti A, Chasse D, Joshi M, Harpole D, Lancaster JM, Berchuck A, Olson JAJ, Marks JR, Dressman HK, West M, Nevins JR: **Oncogenic pathway signatures in human cancers as a guide to targeted therapies.** *Nature* 2006, **439**:353-357.

88 Fulford LG, Reis-Filho JS, Ryder K, Jones C, Gillett CE, Hanby A, Easton D, Lakhani SR: **Basal-like grade III invasive ductal carcinoma of the breast: patterns of metastasis and long-term survival.** *Breast Cancer Res* 2007, **9**:R4.

89 Ellis IO, Bartlett J, Dowsett M, Humphreys S, Jasani B, Miller K, Pinder SE, Rhodes A, Walker R: **Best Practice No 176: Updated recommendations for HER2 testing in the UK.** *J Clin Pathol* 2004, **57**:233-237.

90 Konecny G, Pauletti G, Pegram M, Untch M, Dandekar S, Aguilar Z, Wilson C, Rong H, Bauerfeind I, Felber M, Wang H, Beryt M, Seshadri R, Hepp H, Slamon DJ: **Quantitative association between HER-2/neu and steroid hormone receptors in hormone receptor-positive primary breast cancer.** *J Natl Cancer Inst* 2003, **95**:142-153.

91 Gruvberger SK, Ringnér M, Edén P, Borg A, Fernö M, Peterson C, Meltzer PS: **Expression profiling to predict outcome in breast cancer: the influence of sample selection.** *Breast Cancer Res* 2003, **5**:23-26.

92 Huang J, Tan P, Thiyagarajan J, Bay B: **Prognostic significance of glutathione S-transferase-pi in invasive breast cancer.** *Mod Pathol* 2003, **16**:558-565.

93 Fan C, Oh DS, Wessels L, Weigelt B, Nuyten DSA, Nobel AB, van't Veer LJ, Perou CM: **Concordance among gene-expression-based predictors for breast cancer.** *N Engl J Med* 2006, **355**:560-569.

94 Taylor K, Vichova P, Jordan N, Hiscox S, Hendley R, Nicholson R: **ZIP7-mediated intracellular zinc transport contributes to aberrant growth factor signaling in anti-hormone resistant breast cancer cells.** *Endocrinology* 2008, :.

95 Ross DT, Scherf U, Eisen MB, Perou CM, Rees C, Spellman P, Iyer V, Jeffrey SS, Van de Rijn M, Waltham M, Pergamenschikov A, Lee JC, Lashkari D, Shalon D, Myers TG, Weinstein JN, Botstein D, Brown PO: **Systematic variation in gene expression patterns in human cancer cell lines.** *Nat Genet* 2000, **24**:227-235.

96 Neve RM, Chin K, Fridlyand J, Yeh J, Baehner FL, Fevr T, Clark L, Bayani N, Coppe J, Tong F, Speed T, Spellman PT, DeVries S, Lapuk A, Wang NJ, Kuo W, Stilwell JL, Pinkel D, Albertson DG, Waldman FM, McCormick F, Dickson RB, Johnson MD, Lippman M, Ethier S, Gazdar A, Gray JW: **A collection of breast cancer cell lines for the study of functionally distinct cancer subtypes.** *Cancer Cell* 2006, **10**:515-527.

97 Ertel A, Verghese A, Byers SW, Ochs M, Tozeren A: **Pathway-specific differences between tumor cell lines and normal and tumor tissue cells.** *Mol Cancer* 2006, **5**:55.

98 Nicholson RI, Hutcheson IR, Knowlden JM, Jones HE, Harper ME, Jordan N, Hiscox SE, Barrow D, Gee JMW: **Nonendocrine pathways and endocrine resistance: observations with antiestrogens and signal transduction inhibitors in combination.** *Clin Cancer Res* 2004, **10**:346S-54S.

99 Harris L, Fritsche H, Mennel R, Norton L, Ravdin P, Taube S, Somerfield MR, Hayes DF, Bast RCJ: **American Society of Clinical Oncology 2007 update of recommendations for the use of tumor markers in breast cancer.** *J Clin Oncol* 2007, **25**:5287-5312.

100 Ries LAG, Melbert D, Krapcho M, Stinchcomb DG, Howlader N, Horner MJ, Mariotto A, Miller BA, Feuer EJ, Altekruse SF, Lewis DR, Clegg L, Eisner MP, Reichman M, Edwards BK (eds).: **SEER Cancer Statistics Review.** *National Cancer Institute* 1975-2005, :.

101 **Global Operating System Web Statistics** [http://www.w3counter.com/globalstats.php]

102 Baier T, Neuwirth E. **R (D)-Com Server V1.35** [http://cran.rproject.org/contrib/extra/dcom/RSrv135.html]

103 **W3C Working Group Note 11 Web Services Architecture** [http://www.w3.org/TR/ws-arch/] Feb 2004.

104 Shreiner D, Woo M, Neider J, Davis T. **OpenGL Programming Guide: The Official Guide to Learning OpenGL,** Version 2 (5th Edition). Addison-Wesley Professional

105 Mariani TJ, Budhraja V, Mecham BH, Gu CC, Watson MA, Sadovsky Y: **A variable fold change threshold determines significance for expression microarrays.** *FASEB J* 2003, **17**:321-323.

106 Handl J, Knowles J, Kell DB: **Computational cluster validation in post-genomic data analysis.** *Bioinformatics* 2005, **21**:3201-3212.

107 Brock G, Pihur V, Datta S, Somnath D: **clValid: An R Package for Cluster Validation.** *Journal of Statistics software* 2008, **25**:1-19.

108 Sheriff S, F Qureshy A, T Chance W, Kasckow JW, Balasubramaniam A: **Predominant role by CaM kinase in NPY Y(1) receptor signaling: involvement of CREB [corrected].** *Peptides* 2002, **23**:87-96.

109 Körner M, Waser B, Reubi JC: **High expression of neuropeptide y receptors in tumors of the human adrenal gland and extra-adrenal paraganglia.** *Clin Cancer Res* 2004, **10**:8426-8433.

110 Nijman SMB, Huang TT, Dirac AMG, Brummelkamp TR, Kerkhoven RM, D'Andrea AD, Bernards R: **The deubiquitinating enzyme USP1 regulates the Fanconi anemia pathway.** *Mol Cell* 2005, **17**:331-339.

111 Blamey RW, Ellis IO, Pinder SE, Lee AHS, Macmillan RD, Morgan DAL, Robertson JFR, Mitchell MJ, Ball GR, Haybittle JL, Elston CW: **Survival of invasive breast cancer according to the Nottingham Prognostic Index in cases diagnosed in 1990-1999.** *Eur J Cancer* 2007, **43**:1548-1555.

112 Ioannidis JPA, Rosenberg PS, Goedert JJ, O'Brien TR: **Commentary: meta-analysis of individual participants' data in genetic epidemiology.** *Am J Epidemiol* 2002, **156**:204-210.

113 English translation - Hess V: **[Adjuvant!Online--an Internet-based decision tool for adjuvant chemotherapy in early breast cancer].** Ther Umsch 2008, 65:201-205.

114 Chen, L.E, Nolan, E James, S: **Comparison with Adjuvant! Online.** 2008 Laboratory for Quantitative Medicine Technical Report July 7, 2008. Harvard Medical School, Boston, Massachusetts

115 Chen D, Schell MJ, Chen JJ, Fulp WJ, Eschrich S, Yeatman T: **A predictive risk probability approach for microarray data with survival as an endpoint.** *J Biopharm Stat* 2008, **18**:841-852.

116 Earl Cox: **Fuzzy modelling and genetic algorithms for data mining and exploration.** Elsevier; 2005.

117 Buhler W. **Gauss: A Biographical Study.** P11-20. Springer, 1981.

118 : Bewick V, Cheek L, Ball J. **Statistics review 14: logistic regression.** In *Crit Care* 2005, **9**:112-118.

119 Jonsdottira T, Thora E, Sigurdssona H: *The feasibility of constructing a Predictive Outcome Model for breast cancer using the tools of data mining.* ; 2008.

120 Moguerza JM, Munoz A. **Support vector machines with applications.** In Statist. Sci 2006, 21:322-336..

121 Yang M, Ji YQG: **Unifying multi-class adaboost algorithms with binary base learners under the margin framework.**. In *Pattern Recognition Letters* 2007, **28**:631-643.

122 Islam MM, Yao X, Shahriar Nirjon SMS, Islam MA, Murase K: **Bagging and boosting negatively correlated neural networks.** *IEEE Trans Syst Man Cybern B Cybern* 2008, **38**:771-784.

123 Statnikov A, Wang L, Aliferis CF: **A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification.** *BMC Bioinformatics* 2008, **9**:319.

124 Chun J, Schnabel F, Ogunyemi O: **Assessing a Bayesian risk prediction model in a high-risk breast cancer population.** *AMIA Annu Symp Proc* 2007, :913.

125 Witten I, Frank E, Trigg L, Hall, M, Holmes, G, Cunningham J: **Weka: Practical Machine Learning Tools and Techniques with Java Implementations.** , :.

126 **Openlaszlo - A Platform For Developing Rich Internet Applications** [http://www.openlaszlo.org/]

127 **Adobe Flex 3** [http://www.adobe.com/products/flex/]

128 Frasor J, Stossi F, Danes JM, Komm B, Lyttle CR, Katzenellenbogen BS: **Selective estrogen receptor modulators: discrimination of agonistic versus antagonistic activities by gene expression profiling in breast cancer cells.** *Cancer Res* 2004, **64**:1522-1533.

129 Levenson AS, Svoboda KM, Pease KM, Kaiser SA, Chen B, Simons LA, Jovanovic BD, Dyck PA, Jordan VC: **Gene expression profiles with activation of the estrogen receptor alpha-selective estrogen receptor modulator complex in breast cancer cells expressing wild-type estrogen receptor.** *Cancer Res* 2002, **62**:4419-4426.

130 Scafoglio C, Ambrosino C, Cicatiello L, Altucci L, Ardovino M, Bontempo P, Medici N, Molinari AM, Nebbioso A, Facchiano A, Calogero RA, Elkon R, Menini N, Ponzone R, Biglia N, Sismondi P, De Bortoli M, Weisz A: **Comparative gene expression profiling reveals partially overlapping but distinct genomic actions of different antiestrogens in human breast cancer cells.** *J Cell Biochem* 2006, **98**:1163-1184.

131 Hayashi S, Yamaguchi Y: **Estrogen signaling and prediction of endocrine therapy.** *Cancer Chemother Pharmacol* 2005, **56 Suppl 1**:27-31.

132 Oh DS, Troester MA, Usary J, Hu Z, He X, Fan C, Wu J, Carey LA, Perou CM: **Estrogen-regulated genes predict survival in hormone receptor-positive breast cancers.** *J Clin Oncol* 2006, **24**:1656-1664.

133 Itoh T, Karlsberg K, Kijima I, Yuan Y, Smith D, Ye J, Chen S: **Letrozole-, anastrozole-, and tamoxifen-responsive genes in MCF-7aro cells: a microarray approach.** *Mol Cancer Res* 2005, **3**:203-218.

134 Glidewell-Kenney C, Weiss J, Lee E, Pillai S, Ishikawa T, Ariazi EA, Jameson JL: **ERE-independent ERalpha target genes differentially expressed in human breast tumors.** *Mol Cell Endocrinol* 2005, **245**:53-59.

135 Scott DJ, Parkes AT, Ponchel F, Cummings M, Poola I, Speirs V: **Changes in expression of steroid receptors, their downstream target genes and their associated co-regulators during the sequential acquisition of tamoxifen resistance in vitro.** *Int J Oncol* 2007, **31**:557-565.

136 Fan M, Yan PS, Hartman-Frey C, Chen L, Paik H, Oyer SL, Salisbury JD, Cheng ASL, Li L, Abbosh PH, Huang TH, Nephew KP: **Diverse gene expression and DNA methylation profiles correlate with differential adaptation of breast cancer cells to the antiestrogens tamoxifen and fulvestrant.** *Cancer Res* 2006, **66**:11954-11966.

137 Hilsenbeck SG, Friedrichs WE, Schiff R, O'Connell P, Hansen RK, Osborne CK, Fuqua SA: **Statistical analysis of array expression data as applied to the problem of tamoxifen resistance.** *J Natl Cancer Inst* 1999, **91**:453-459.

138 Huber M, Bahr I, Krätzschmar JR, Becker A, Müller E, Donner P, Pohlenz H, Schneider MR, Sommer A: **Comparison of proteomic and genomic analyses of the human breast cancer cell line T47D and the antiestrogen-resistant derivative T47D-r.** *Mol Cell Proteomics* 2004, **3**:43-55.

139 Sommer A, Hoffmann J, Lichtner RB, Schneider MR, Parczyk K: **Studies on the development of resistance to the pure antiestrogen Faslodex in three human breast cancer cell lines.** *J Steroid Biochem Mol Biol* 2003, **85**:33-47.

140 Gu Z, Lee RY, Skaar TC, Bouker KB, Welch JN, Lu J, Liu A, Zhu Y, Davis N, Leonessa F, Brünner N, Wang Y, Clarke R: **Association of interferon regulatory factor-1, nucleophosmin, nuclear factor-kappaB, and cyclic AMP response element binding with acquired resistance to Faslodex (ICI 182,780).** *Cancer Res* 2002, **62**:3428-3437.

141 Chanrion M, Negre V, Fontaine H, Salvetat N, Bibeau F, Mac Grogan G, Mauriac L, Katsaros D, Molina F, Theillet C, Darbon J: **A gene expression signature that can**

predict the recurrence of tamoxifen-treated primary breast cancer. *Clin Cancer Res* 2008, **14**:1744-1752.

142 Tozlu-Kara S, Roux V, Andrieu C, Vendrell J, Vacher S, Lazar V, Spyratos F, Tubiana-Hulin M, Cohen P, Dessen P, Lidereau R, Bièche I: **Oligonucleotide microarray analysis of estrogen receptor alpha-positive postmenopausal breast carcinomas: identification of HRPAP20 and TIMELESS as outstanding candidate markers to predict the response to tamoxifen.** *J Mol Endocrinol* 2007, **39**:305-318.

143 Knowlden JM, Hutcheson IR, Barrow D, Gee JMW, Nicholson RI: **Insulin-like growth factor-I receptor signaling in tamoxifen-resistant breast cancer: a supporting role to the epidermal growth factor receptor.** *Endocrinology* 2005, **146**:4609-4618.

144 Bertucci F, Finetti P, Cervera N, Esterni B, Hermitte F, Viens P, Birnbaum D: **How basal are triple-negative breast cancers?.** *Int J Cancer* 2008, **123**:236-240.

145 Dowsett M, Harper-Wynne C, Boeddinghaus I, Salter J, Hills M, Dixon M, Ebbs S, Gui G, Sacks N, Smith I: **HER-2 amplification impedes the antiproliferative effects of hormone therapy in estrogen receptor-positive primary breast cancer.** *Cancer Res* 2001, **61**:8452-8458.

146 Berns EM, Foekens JA, Vossen R, Look MP, Devilee P, Henzen-Logmans SC, van Staveren IL, van Putten WL, Inganäs M, Meijer-van Gelder ME, Cornelisse C, Claassen CJ, Portengen H, Bakker B, Klijn JG: **Complete sequencing of TP53 predicts poor response to systemic therapy of advanced breast cancer.** *Cancer Res* 2000, **60**:2155-2162.

147 Bostner J, Ahnström Waltersson M, Fornander T, Skoog L, Nordenskjöld B, Stål O: **Amplification of CCND1 and PAK1 as predictors of recurrence and tamoxifen resistance in postmenopausal breast cancer.** *Oncogene* 2007, **26**:6997-7005.

148 Achuthan R, Bell SM, Roberts P, Leek JP, Horgan K, Markham AF, MacLennan KA, Speirs V: **Genetic events during the transformation of a tamoxifen-sensitive human breast cancer cell line into a drug-resistant clone.** *Cancer Genet Cytogenet* 2001, **130**:166-172.

149 Bose S, Wang SI, Terry MB, Hibshoosh H, Parsons R: **Allelic loss of chromosome 10q23 is associated with tumor progression in breast carcinomas.** *Oncogene* 1998, **17**:123-127.

150 Hirata K, Tagawa Y, Kashima K, Kidogawa H, Deguchi M, Tsuji T, Ayabe H: **Frequency of chromosome 7 gain in human breast cancer cells: correlation with the number of metastatic lymph nodes and prognosis.** *Tohoku J Exp Med* 1998, **184**:85-97.

151 Whang-Peng J, Lee EC, Kao-Shan CS, Seibert K, Lippman M: **Cytogenetic studies of human breast cancer lines: MCF-7 and derived variant sublines.** *J Natl Cancer Inst* 1983, **71**:687-695.

152 Britton DJ, Hutcheson IR, Knowlden JM, Barrow D, Giles M, McClelland RA, Gee JMW, Nicholson RI: **Bidirectional cross talk between ERalpha and EGFR signalling pathways regulates tamoxifen-resistant growth.** *Breast Cancer Res Treat* 2006, **96**:131-146.

153 Hiscox S, Jordan NJ, Jiang W, Harper M, McClelland R, Smith C, Nicholson RI: **Chronic exposure to fulvestrant promotes overexpression of the c-Met receptor in breast cancer cells: implications for tumour-stroma interactions.** *Endocr Relat Cancer* 2006, **13**:1085-1099.

154 Dussault I, Bellon SF: **c-Met inhibitors with different binding modes: two is better than one.** *Cell Cycle* 2008, **7**:1157-1160.

155 Meijer D, Sieuwerts AM, Look MP, van Agthoven T, Foekens JA, Dorssers LCJ: **Fibroblast growth factor receptor 4 predicts failure on tamoxifen therapy in patients with recurrent breast cancer.** *Endocr Relat Cancer* 2008, **15**:101-111.

156 Eppenberger U, Kueng W, Schlaeppi JM, Roesel JL, Benz C, Mueller H, Matter A, Zuber M, Luescher K, Litschgi M, Schmitt M, Foekens JA, Eppenberger-Castori S: **Markers of tumor angiogenesis and proteolysis independently define high- and low-risk subsets of node-negative breast cancer patients.** *J Clin Oncol* 1998, **16**:3129-3136.

157 Qu Z, Van Ginkel S, Roy AM, Westbrook L, Nasrin M, Maxuitenko Y, Frost AR, Carey D, Wang W, Li R, Grizzle WE, Thottassery JV, Kern FG: **Vascular endothelial growth factor reduces tamoxifen efficacy and promotes metastatic colonization and desmoplasia in breast tumors.** *Cancer Res* 2008, **68**:6232-6240.

158 Keyse SM: **Dual-specificity MAP kinase phosphatases (MKPs) and cancer.** *Cancer Metastasis Rev* 2008, **27**:253-261.

159 Gutierrez MC, Detre S, Johnston S, Mohsin SK, Shou J, Allred DC, Schiff R, Osborne CK, Dowsett M: **Molecular changes in tamoxifen-resistant breast cancer: relationship between estrogen receptor, HER-2, and p38 mitogen-activated protein kinase.** *J Clin Oncol* 2005, **23**:2469-2476.

160 Knowlden JM, Hutcheson IR, Jones HE, Madden T, Gee JMW, Harper ME, Barrow D, Wakeling AE, Nicholson RI: **Elevated levels of epidermal growth factor receptor/c-erbB2 heterodimers mediate an autocrine growth regulatory pathway in tamoxifen-resistant MCF-7 cells.** *Endocrinology* 2003, **144**:1032-1044.

161 Cui Y, Parra I, Zhang M, Hilsenbeck SG, Tsimelzon A, Furukawa T, Horii A, Zhang Z, Nicholson RI, Fuqua SAW: **Elevated expression of mitogen-activated protein kinase**

phosphatase 3 in breast tumors: a mechanism of tamoxifen resistance. *Cancer Res* 2006, **66**:5950-5959.

162 **Health Information Research Unit (Hiru)** [http://www.wales.nhs.uk/]

163 Bassett D. **Fish are rising.** *Genome Biol.* 2001; 2(7): reports 4016.1–4016.2

164 Jelen B. **The spreadsheet at 25.** May 2005. p 23-50.

165 Bland JM, Altman DG. **The logrank test.** *BMJ* 2004;328:1073

166 Kumar D, Klefsjo B. **Proportional hazards model: a review.** *Reliability engineering & systems safety.* 1994, vol. 44, no2, pp. 177-188

167 Stratton J, **Apple switches to Intel processors.** 2005. http://www.hoboes.com/Mimsy/?ART=199

Appendix 1 – Preparation of MCF7 cell line for Affymetrix microarray analysis

The human oestrogen-dependant, breast carcinoma cell lines MCF-7 were cultured in RPMI 1640 with L-glutamine medium supplemented with 5% of foetal bovine serum (FBS) as standard tissue culture conditions. RPMI and FBS were from GIBCO BRL Life Technologies (Paisley, UK).

RNA for microarray analysis was isolated using the RNAeasy mini system (Qiagen) RNA samples were processed for Affymetrix GeneChip® hybridization using the MessageAmpTM aRNA Kit (Ambion).

For each sample, 5ug of total RNA was used to generate double stranded cDNA and was purified according to the MessageAmpTM aRNA Kit (Ambion) protocol.

In vitro transcription (IVT) reactions to produce antisense RNA were carried out in 20ul reactions consisting of:
1.5 µl cDNA
2 µl T7 10×Reaction Buffer
3.75 µl 10mMbiotin-11-CTP (Perkin-Elmer)
3.75 µl 10mM biotin-16-UTP (Perkin-Elmer)
2 µl 75mM ATP
2 µl 75mM GTP
1.5 µl 75mM CTP
1.5 µl 75mM UTP
2 µl T7 enzyme mix

The reaction was incubated for 371°C for 6 h.

The IVT reactions were DNase I treated and purified according to the manufacturer's protocol.

Sample fragmentation, hybridization and GeneChip® array washing and staining were carried out according to the GeneChip® Expression Analysis Technical Manual (Affymetrix).

The stained arrays were scanned using a GeneChip® Scanner 3000 (Affymetrix).

Appendix 2 – Phenotype status of the MCF7 models - control, TAMR and FASR Affymetrix microarrays

P = Present
M = Marginal
A = Absent

The description of the arrayed models has been shortened. For example:
C1 = Control model replicate 1
T1 = Tamoxifen resistant model replicate 1
F1 = Faslodex resistant model replicate 1
with numbering indicating each respective replicate.

Sorlie erb2

AffyID	Description	C1	C2	C3	T1	T2	T3	F1	F2	F3
210930_s_at	erbB2	P	P	P	P	P	P	A	P	P
216836_s_at	erbB2	P	P	P	P	P	P	P	P	P
203497_at	PPAR binding protein	P	A	A	A	P	A	P	P	P
203496_s_at	PPAR binding protein	A	A	A	A	A	A	A	A	A
213043_s_at	Thyroid hormone receptor associated protein4	P	P	P	P	P	P	P	P	P
202991_at	STARD3	P	M	M	P	P	P	M	P	P
210761_s_at	GRB-7 growth factor receptor bound-7	P	P	P	P	P	P	P	P	P
202039_at	TIAF1 Tgfb1-induced antiapoptosis factor	P	P	A	P	P	A	P	P	P
211899_s_at	TRAF4 TNF receptor associated factor 4	P	P	P	P	P	P	P	P	P
202871_at	TRAF4 TNF receptor associated factor 4	P	P	P	P	P	P	P	P	P
201350_at	flotillin 2	P	P	P	P	P	P	P	P	P
211299_s_at	flotillin 2	P	P	P	P	P	P	P	P	P
218464_s_at	FLJ 10700	P	A	A	P	M	A	A	A	A
211988_at	SMARCE1	P	P	P	P	P	P	P	P	P
211989_at	SMARCE1	P	M	P	P	P	A	P	P	P
202606_s_at	TLK1(mod)	P	P	P	P	P	P	P	P	P
211077_s_at	TLK1(mod)	A	A	A	A	M	A	A	P	A
210379_s_at	TLK1(mod)	A	A	A	A	A	A	A	A	A

Sorlie Basal

AffyID	Description	C1	C2	C3	T1	T2	T3	F1	F2	F3
202935_s_at	SRY sex determining region Y-box	A	M	A	P	P	A	A	A	A
202936_s_at	SRY sex determining region Y-box	A	A	A	A	P	A	A	A	A
203398_s_at	UDP-n-acetyl-alpha-D-galactosamine	A	A	A	A	A	A	A	A	A
203397_s_at	UDP-n-acetyl-alpha-D-galactosamine	P	A	P	P	P	P	P	P	P
203256_at	P-cadherin 3	P	P	P	A	P	A	P	P	P

207517_at	laminin gamma 2	A	A	A	A	A	A	A	A	A
202267_at	laminin gamma 2	A	A	A	A	A	A	A	A	A
202504_at	ATDC	A	A	A	A	A	A	A	A	A
211002_s_at	ATDC	A	A	A	A	A	A	A	A	A
211001_at	ATDC	A	A	A	A	A	A	A	A	A
205157_s_at	keratin 17	A	A	A	A	A	A	A	A	A
212236_x_at	keratin 17	A	A	A	A	A	A	A	A	A
201820_at	keratin 5	A	A	A	A	A	A	A	A	A
206393_at	troponin	A	A	A	A	A	A	A	A	A
213060_s_at	chitinase 3 like-2	A	A	A	A	A	A	A	A	A
203021_at	secretory protease inhibitor antlock proteinase	A	A	A	P	P	P	A	A	A
209290_s_at	nuclear factor I/B	P	A	P	A	P	A	P	P	P
211467_s_at	nuclear factor I/B	A	A	A	A	A	A	A	A	A
211466_at	nuclear factor I/B	A	A	A	A	A	A	A	A	A
206538_at	mRAS est	A	A	A	A	A	A	A	A	A
209908_s_at	TGFbeta 2	P	A	A	A	A	A	A	A	A
220407_s_at	TGFbeta 2	P	A	A	A	A	A	A	A	A
209909_s_at	TGFbeta 2	P	P	P	A	A	A	A	A	A
202966_at	calpain-like protease	A	A	A	A	A	A	A	A	A
202965_s_at	calpain-like protease	A	A	A	A	A	A	A	A	A
217387_at	calpain-like protease	A	A	A	A	A	A	A	A	A
208086_s_at	dystrophin muscular dystrophy	A	A	A	A	A	A	A	A	A
203881_s_at	dystrophin muscular dystrophy	A	A	A	A	A	A	A	A	A
207660_at	dystrophin muscular dystrophy	A	A	A	A	A	A	A	A	A
205029_s_at	fatty acid binding protein 7	A	A	A	A	A	A	A	A	A
205030_at	fatty acid binding protein 7	A	A	A	A	A	P	A	A	A
216192_at	fatty acid binding protein 7	A	A	A	A	A	A	A	A	A
204470_at	GRO oncogene alpha	A	A	A	A	A	P	A	A	A
203074_at	ANXA8	P	P	P	P	P	P	P	P	P
210605_s_at	MFGE8	A	A	A	P	A	A	A	A	A
823_at	CX3CL1	A	A	A	A	A	A	P	A	P
203706_s_at	FZD7	P	P	P	P	P	P	A	A	A
203705_s_at	FZD7	P	P	P	P	P	P	P	A	A
203687_at	CX3CL1	A	A	A	A	A	A	A	A	A

Sorlie Normal

AffyID	Descrip	C1	C2	C3	T1	T2	T3	F1	F2	F3
206488_s_at	CD36 collagen 1 receptor	A	A	A	A	A	A	P	P	P
209555_s_at	CD36 collagen 1 receptor	A	A	A	A	A	A	A	P	P
209554_at	CD36 collagen 1 receptor	A	A	A	A	A	A	A	A	A
214091_s_at	glutathione peroxidase 3	P	P	P	P	P	P	P	P	P
201348_at	glutathione peroxidase 3	P	P	P	P	P	P	P	P	P
213706_at	glycerol-3-phosphate dehydrogenase 1	A	A	A	A	A	A	A	A	A
204997_at	glycerol-3-phosphate dehydrogenase 1	A	A	A	A	A	A	A	A	A
203549_s_at	lipoprotein lipase	A	A	A	A	A	A	A	A	A
203548_s_at	lipoprotein lipase	A	A	A	A	A	A	A	A	A
214505_s_at	four and a half LIM domains	A	A	A	A	A	A	A	A	A
210299_s_at	four and a half LIM domains	A	A	A	A	A	A	A	A	A
201539_s_at	four and a half LIM domains	M	P	P	P	P	A	A	A	P

210298_x_at	four and a half LIM domains	A	A	A	A	A	A	A	A	A
201540_at	four and a half LIM domains	P	P	P	P	P	P	A	P	A
219140_s_at	retinol-binding protein 4	A	A	A	A	A	A	A	A	A
204894_s_at	vascular-adhesion protein 1	A	A	A	A	A	A	A	A	A
209663_s_at	integrin alpha 7	A	A	A	A	A	A	A	A	A
216331_at	integrin alpha 7	A	A	A	A	A	A	A	A	A
209613_s_at	alcohol dehydrogenase-2 class 1 beta	A	A	A	A	A	A	A	A	A
209612_s_at	alcohol dehydrogenase-2 class 1 beta	A	A	A	A	A	A	A	A	A
209614_at	alcohol dehydrogenase-2 class 1 beta	A	A	A	A	A	A	A	A	A
202595_s_at	leptin receptor overlappin	P	P	P	P	P	P	P	P	P
202594_at	leptin receptor overlappin	P	P	P	P	P	P	P	P	P
206955_at	aquaporin	A	A	A	A	A	A	A	A	A
219398_at	CICE 30KDa protein	P	A	P	P	P	P	P	P	P
204151_x_at	AKR1C1	P	P	P	P	P	P	P	P	P
216594_x_at	AKR1C1	P	P	P	P	P	P	P	P	P
201963_at	FACL2	P	A	P	P	P	P	P	P	P
207275_s_at	FACL2	A	A	A	P	P	P	A	P	P

Sorlie Luminal

AffyID	Description	C1	C2	C3	T1	T2	T3	F1	F2	F3
205355_at	acyl-coenzyme A dehydrogenase	A	A	A	A	A	A	A	M	P
205225_at	estrogen receptor	P	P	P	P	P	P	P	P	P
215552_s_at	estrogen receptor	P	A	A	A	A	A	A	A	A
211233_x_at	estrogen receptor	A	A	A	A	A	A	A	A	A
211235_s_at	estrogen receptor	A	A	A	A	A	A	A	A	A
211234_x_at	estrogen receptor	A	A	A	A	A	A	A	A	A
211627_x_at	estrogen receptor	A	A	A	A	A	A	A	A	A
217190_x_at	estrogen receptor	A	A	A	A	A	A	A	A	A
207672_at	estrogen receptor	A	A	A	A	A	A	A	A	A
204623_at	trefoil factor 3 intestinal	P	P	P	P	P	P	A	P	A
209604_s_at	GATA binding protein 3	P	P	P	P	P	P	P	P	P
209602_s_at	GATA binding protein 3	P	P	P	P	P	P	P	P	P
209603_at	GATA binding protein 3	P	P	P	A	A	A	P	P	P
200670_at	x-box binding protein	P	P	P	P	P	P	P	P	P
204667_at	FOX A1 - hepatocyte nuclear factor 3	P	P	P	P	P	P	P	P	P
202088_at	LIV-1	P	P	P	P	P	P	P	P	P
202089_s_at	LIV-1	P	P	P	P	P	P	P	P	P
210085_s_at	Annexin A9	P	P	P	P	P	P	P	P	P
211712_s_at	Annexin A9	P	P	P	P	P	P	P	P	P
214440_at	n-acetyl transferase 1	P	P	P	P	P	P	P	P	P
210480_s_at	Myosin VI	A	A	A	A	A	A	A	A	A
203215_s_at	Myosin VI	A	A	A	A	A	A	A	A	A
203216_s_at	Myosin VI	P	P	P	M	P	P	P	P	P
212692_s_at	LRBA	P	P	P	P	P	P	P	P	P
214109_at	LRBA	P	P	P	P	P	P	P	P	P
208615_s_at	PTP4A2	P	P	P	P	P	P	P	P	P
208617_s_at	PTP4A2	P	A	P	P	P	P	P	P	P
216988_s_at	PTP4A2	P	P	P	P	P	P	P	P	P
219197_s_at	SCUBT2	P	P	P	A	A	A	A	A	A

Appendix 3 – SEER cancer patient dataset coding alterations for database storage and data analysis

Table A3.1 outlines the original coding of different breast cancer patient variables and the new coding nomenclature adopted where a change was required. Table A3.2 shows the coding and changes adopted for the colorectal cancer dataset.

Breast cancer patient variables	Original coding	New coding
Race	1, 2, 3 or 4	No change
Year	e.g: 1976	No change
Histology	8010, 8050, 8070, 8140, 8201, 8211, 8480, 8500, 8501, 8503, 8510, 8520, 8521, 8522, 8530, 8541	No change
Tumour site	500, 501, 502, 503, 504, 505, 506, 508, 509	No change
Tumour grade	I, II, III or IV	1,2,3 or 4
Cause of death	Textual – varies according to type	Changed to alive or dead over a ten year period 1 or 0
Nodes examined	0 to 75	No change
Positive nodes	0 to 75	No change
Tumour extent	10, 20, 30, 40, 50, 70	No change
Tumour size	0 to 50	No change
Age	e.g: 45	No change
Surgery received	0, 1, 2, 10, 20, 30, 40, 48, 50, 58, 60, 80, 90	No change
Radiation received	0, 1 and 3	No change
Radiation sequence surgery	1,2,3,4,5,6,7	No change
Martial status	1,2,3,4 or 5	No change
Number of primaries	1,2,3 or 4	No change
PgR status	1,2 or 3	No change
Er status	1,2 or 3	No change
Survival time in months	e.g: 32	No change
Patient ID	e.g: 98493	No change

Table A3.1: Summary of the existing and modified SEER breast cancer patient dataset coding.

Colorectal cancer patient variables	Original coding	New coding
Race	1, 2, 3 or 4	No change
Sex	1 or 2	No change
Year	e.g 1954	No change
Histology	8010, 8050, 8070, 8140, 8201, 8211, 8480, 8500, 8501, 8503, 8510, 8520, 8521, 8522, 8530, 8541	No change
Tumour site	1,2,3,4,5,6,7,8,9,10,11 or 12	No change
Tumour grade	I, II, III or IV	1,2,3 or 4
Cause of death	Textual – varies according to type	Changed to alive or dead over a ten year period 1 or 0
Nodes examined	0 to 75	No change
Positive nodes	0 to 75	No change
Tumour extent	00, 10, 11, 12, 20, 30, 40, 60, 65, 70	No change
Tumour size	0 and 50	No change
Age	e.g: 76	No change
Surgery received	1,2,3,4,5,6,7,8 or 9	No change
Radiation received	1,2,3,4,5,6,7,8 or 9	No change
Martial status	1,2,3,4 or 5	No change
Number of Primaries	1,2,3 or 4	No change
Radiation sequence surgery	1,2,3,4,5,6 or 7	No change
Survival time in months	e.g: 45	No change
Patient ID	e.g:56433	No change

Table A3.2: Summary of the existing and modified SEER colorectal cancer patient dataset coding.

Appendix 4 – Transforming the SEER breast cancer dataset to calculate a Nottingham prognostic index value for each patient

The 'R' statistical programming environment was used to compare the NPI prognostic index categories against the SEER breast cancer dataset as outlined in Chapter 5.

A data frame to hold the survival data and NPI scores was initially created:

```
> data.npi<-data.all
> npi<-c(rep(0,nrow(data.npi)))
> data.npi<-cbind(data.npi, npi)
```

The scoring system for grade to be altered as grade in the UK it ranges between 1 and 3 whereas the SEER dataset had values up to 4 which needed to be included in those cases classified with a grade of 3.

The same procedure was performed for the number of nodes positive.

```
>data.npi$grade<-ifelse(data.npi$grade==4,3,data.npi$grade)

>data.npi$nodespos<-ifelse(data.npi$nodespos<1, 1,
ifelse(data.npi$nodespos<4, 2, 3))
```

The NPI scores were then calculated using the NPI formula changing the tumour size from millimetres as used in the SEER dataset to cm as used in the NPI.

```
>data.npi$npi<-as.numeric(data.npi$nodespos) + as.numeric(data.npi$grade) +
(data.npi$size/10)*0.2
```

NPI scores according to the prognostic categories were assigned to the results. This was performed in separate stages first calculating the number who survive and then the percentage and overlay according to the NPI prognostic grouping. For example: EPG = Excellent prognosis group.

```
> data.npi$npi<-ifelse(data.npi$npi<=2.4, 1,ifelse(data.npi$npi <=3.4, 2,
ifelse(data.npi$npi<=4.4, 3, ifelse(data.npi$npi<=5.4, 4,
ifelse(data.npi$npi<=6.4, 5, 6)))))
> data.npi$npi<-as.factor(data.npi$npi)
> groups<-c(rep(0,6))
> npi.table<-table(data.npi$npi, data.npi$alivestatus)
percent.surviving<-100*(npi.table [,1]/( npi.table [,1]+ npi.table [,2]))
> groups<-as.data.frame(percent.surviving, row.names=c("EPG", "GPG", "MPG1",
"MPG2", "PPG", "VPG"))
```

The results and discussion are shown in Chapter 5, table 5.3 and table 5.4