Workshop

An introduction to R

Andy Buerki

Overview

R is a software tool for statistical analysis. It is today the tool of choice for quantitative linguists and is used increasingly by linguists and scientists/scholars of all specialisations due to its flexibility and expandability to cover virtually any known statistical procedure. In this workshop we are going to use R through an interface called R Studio which facilitates an enhanced user experience. The workshop aims to introduce participants to ways in which R can be used to conduct a range of common statistical analyses. The focus is on how to conduct analyses in R rather than on statistical procedures per se. A full set of handouts and transcripts will enable participants to follow up on topics discussed in the workshop and review everything that was covered.

Topics

2.10 -3.00	Basics	Elements of the R Studio interface Importing and exporting data into and out of R Data manipulation in R: displaying, partially displaying, copying and creating data objects
	Descriptive statistics	Data summarisation functions, checking distribution Data visualisation functions: producing and exporting plots
3.10 – 4.00	Inferential statistics	Correlation Chi-square tests t-tests ANOVA Regression and multiple regression

Please note that due to time constraints, some of these topics will be covered only very superficially, but a full set of support materials will enable participants to follow up examples in greater detail.

Prerequisites

No prior knowledge of R is assumed. A general understanding of statistical data analysis and advanced computer skills will be helpful, but not essential.

Software installation

For the workshop, university laptops with R and R Studio pre-installed will be supplied. *To install R and R Studio on other university-owned computers running Windows*, there is an installer in Cardiff Apps > Cardiff Apps > School Applications > ENCAP . To install the software on a private computer, download and install, in this order, R (http://www.stats.bris.ac.uk/R/) *and* R Studio (http://www.rstudio.com/). Both R and R Studio are free.

Reading List

No preparation is required for the workshop, but for keen participants, I would recommend Mizumoto and Plonsky (2015) for some background on R and its advantages.

Baayen, H. (2008). Analysing Linguistic Data: A Practical Introduction to Statistics using R. Cambridge: Cambridge University Press. Despite the title, this is in fact a very advanced resource book by one of the best known quantitative linguists.

Crawley, M. (2013) The R Book, 2nd ed. Chichester: Wiley and Sons [e-book available through the library] This is a very comprehensive reference book on R and statistical analyses using R. It is not written with linguistic data in mind, but is still useful as a reference work.

Gries, S. T. (2013). Statistics for Linguistics with R: A practical introduction. Berlin: De Gruyter. [P138.5.G7] Written by one of the foremost quantitative linguists. This is no easy read, but still more introductory than Baayen.

Johnson, K. (2008). Quantitative Methods in Linguistics. Malden, MA: Blackwell. [P138.5.J6] A book with some very interesting applications of statistical procedures to linguistic data in the fields of phonetics, psycholinguistics, sociolinguistics, historical linguistics and syntax. I don't get the organisation of this book, but maybe you do.

Levshina, N. (forthcoming). How to do Linguistics with R: Data exploration and statistical analysis, Amsterdam: Benjamins. Hopefully this will be a easier-to-follow book while still dealing with advanced topics. Natalia Levshina is at Leuven where they have an excellent quantitative linguistics research group.

Mizumoto, A. and Plonsky, L. (2015). R as a lingua franca: Advantages of using R for quantitative research in applied linguistics. *Applied Linguistics*, *2015*(advance access).

R Basics

First off, R is unforgiving about typos, so unless names of objects and everything else is typed exactly right, we will get errors or unexpected results.

1 Creating and removing objects

Objects are created using arrows to a name

`AGE<-c(37,24,30,46) or c(8,6,5,10)->SCORES`

`c("m","f","m","f")->GENDER`

make a data frame out of existing variables

`our.data<-data.frame(GENDER, AGE, SCORES)`

you can always create and edit a data frame in Excel, export it as .csv file and import it into R (it is worth making sure headers are imported correctly)

We remove objects like this: **`rm(X)`**

where 'X' is the object to be removed. The object disappears irretrievably after this command.

2 Exporting data frames

`write.csv(X, file="FILENAME.csv",col.names=F)`

X is the name of the data frame, FILNAME.csv is the name of the file you want to create.

3 Editing data frames

Again, if you feel more confident doing this in Excel, that's fine, just export and re-import the data frame into R

`fix(X)` # where X is the name of the data frame. Make changes in the window that comes up, save and close. You can change the name of variables by clicking on them. Some edits (like removing or re-ordering columns or rows) cannot be done with fix(). See '6 change data frames' below for how to do such things.

4 Navigating data frames

Often we want to display only certain parts of a data frame, either because the whole thing is too big or because we want to use data in a sub-part in a certain function. Here's how to pick out subsets of values from a data frame (all commands are relative to the data frame our.data displayed on the right)

	AGE	GENDER	SCORES
1	37	m	8
2	24	f	6
3	30	m	5
4	46	f	10

Picking out values WITHOUT column names and row numbers (this only picks out the values themselves and this is usually what you want if you use the values as input to a function):

- we use the '$' sign after the name of data frame to specify the column name
- we can further specify the rows to be displayed in square brackets []

`our.data$AGE` # displays the values of the variable AGE inside our.data

`our.data$AGE[c(1,4)]`

displays the values in row 1 AND 4 of the variable AGE inside our.data

our.data$AGE[1:3]

displays the values in row 1 TO 3 of the variable AGE inside our.data

our.data$AGE[our.data$GENDER == "m"]

displays the values of AGE where GENDER is 'm'

We can now put those values we pick out into a function like mean():

mean(our.data$AGE[our.data$GENDER == "f"])

displays the mean age of males in our data

Picking out values WITH column names and row numbers

(this usually causes errors if used in functions, but is fine otherwise):

- we provide either only column names or numbers in [] or row numbers COMMA column names/numbers.
- we can leave out row or column names/numbers if we want all rows or columns

our.data[1] or **our.data["AGE"]** # first column only

first and second columns:

our.data[1:2] or **our.data[c(1,2)]** or **our.data[c("AGE","GENDER")]**

our.data[c(1,4),] # rows 1 AND 4 of all columns

our.data[c(1,2,4),c("GENDER","SCORES")]

rows 1, 2 and 4 of columns 2 and 3

our.data[c(4,3,2,1),c(2,1,3)]

rows 1 to 4 in reverse order and columns 2, 1 and 3 in that order

5 Copy data frames (it's a good idea to make a backup copy before changing data frames)

To copy a data frame (for backup for example) we can export it (see above) or just put it it under a new name

our.data->bkup.our.data

now 2 identical data frames exist under our.data and bkup.our.data

6 Change the order of variables or cases in data frames

Again, you can to export to Excel and re-importing into R if you feel that is easiest. In R, to change the order of columns and rows, deleting columns, rows, etc., we simply display what we want in the new data frame (see 'picking out values WITH column names and row numbers' under 'navigating data frames') and then put it into a new name or the same name if we want to replace the data frame:

our.data[c(4,3,2,1),c(1,3)]->new.name

the new data frame will be new.name

our.data[c(4,3,2,1),c(1,3)]->our.data

this overwrites/replaces our.data

To add a new column, we just tell R what data to put where, e.g.

our.data$AGE*2 # we display each value in AGE, multiplied by 2

our.data$AGE*2->our.data$DBL.AGE # we put it into a column called DBL.AGE in our.data

7 Getting an overview

These functions give an overview of a data frame:

```
length(X)      # gives the number of columns (or other elements) in X
str(X)         # displays information about the data frame X
summary(X)     # displays a summary of the data frame X
```

8 Converting variables between character, factor and numeric

Here is how we can make certain R uses the correct type for a variable

```
c(1,2,3,4,5)-> a
```

This creates a vector with numbers 1 to 4. This will automatically be a numeric type

```
as.character(a)-> a # now the type is changed to character
as.factor(a)-> a    # now the type is changed to factor (= categorical variable)
as.numeric(a)-> a   # now the type is changed back to numeric (= interval variable)
```

To create an ordinal variable, we might do this

```
ranks=c("first","third","second","first","third")
```

created a vector of character type

```
ordered(ranks,c("first","second","third"))
```

created a vector of type ordered factor (=ordinal variable). The 'ordered' function takes the data vector first, then then you need to indicate the ordering after the comma.

Descriptive statistics

See transcript for application examples to actual data.

1 Visualising

Tool	Suitable Variables (levels of measurement, how many variables, other things to consider)	R

frequency tables	one or more categorical variables (typically, although ordinal and interval/ratio variables possible)	`table(X)`
		`table(X,Y)`
		`prop.table(table(X,Y))`

scatterplot	typically two interval/ratio variables, although ordinal variables can be plotted here as well	`plot(X, Y)`
		
		`abline(lm(Y~X))`

barplot	one or more categorical variables	`barplot(table(X,Y),`
		`beside=T,`
		`legend=c("a","b"))`

histogram	one interval/ratio variable that is continuous	`hist(X, breaks=10)`
line graph	on the x-axis you need a variable at least on an ordinal level, typically involving time periods on the y-axis you can either have the values of an interval/ratio variable for frequencies of a categorical one	`plot(type="l", X,Y)` `lines(type="l",X,Z)`
pie chart	one categorical variable	`pie(table(X),` `labels=c("a","b"))`

boxplot	one or more interval/ratio variables (those are given as X, Y and Z respectively in the code)	`boxplot(X,Y,Z)`
		`text(1:3, mean(X),`
		`mean(Y),`
		`mean(Z),c("+","+","+")`

R commands to adjust graphs

`xlim=c(0,10), ylim=c(0,10)` # to set the minimum and maximum values for x-axis (xlim) or y-axis (ylim)

`xaxt="n", yaxt="n"` # suppress the drawing of x-axis (xaxt) or y-axis (yaxt); usually because we want to add those later using axis(), see below

`main="main title"` # to supply a main title for the graph; **`xlab="name", ylab="name"`** # to name the x-axis (xlab) or y-axis (ylab)

`col=c("white", "grey20", "grey60", "grey80", "black")` # to define the colours with which variables are drawn. Include as many colours as you have variables)

Andy Buerki 10/2015

In combination with plot(): **`type="l"`** # this indicates: "l" = line (as opposed to points), "b" = both lines and points, "s"= stairs, "h" = histogram-type lines

In combination with plot(): **`pch=1`** # point character; try out values 1 to 25 to see the different styles

`lty=1` # line type, you can try out different values and see what they look like

`lwd=1` # the weight of lines drawn, a higher number draws a bolder line

To add an axis: (while drawing the plot, use xaxt="n" / yaxt="n" to suppress the automatic axes)

```
axis(1, at=c(1,2,3), labels=c("a","b","c"))
```

1=x-axis, 2=y-axis — where (at which values) on the axis to place tick marks — 'labels' labels the tick marks with the labels provided

2 Summarising

Tool	Suitable Variables	R
mean	normally distributed interval/ratio variable (you can calculate it for non-normally distributed variables, but it is not very meaningful in that case)	`mean(X)`
median	ordinal variable or higher level of measurement	`median(X)`
mode	categorical variable or higher level of measurement	`sort(table(X))`
range	ordinal variable or higher level of measurement	`range(X)` `diff(range(X))`
interquartile range	ordinal variable or higher level of measurement	`quantile(X)` `quantile(X)[4]-quantile(X)[2]`
standard deviation	interval/ratio variable	`sd(X)`
variance	interval/ratio variable	`var(X)`

Inferential Statistics

Hypothesis testing procedure

a. formulate H0 and H1
b. set significance level (also called alpha level)
c. get an overview using descriptive stats
d. test assumptions of stat. procedure to be used
e. calculate p-values for H0
f. decide if result is significant (that is, whether to reject the H0)

Levels of measurement

- ratio scale (for present purposes no different from interval scale)
- Interval scale (values are scaled with equidistant intervals, e.g. 4 is twice as much as 2)
- Ordinal scale (values are ordered but not necessarily w/ equal intervals, e.g. 4th place is not (necessarily) twice 2nd place)
- Nominal / categorical scale (values cannot be ordered, just different, e.g. 'male' vs. 'female')
- Frequencies: typically need to be treated as frequencies of categories, but can occasionally be abstracted into a 'measure' of an interval scale, e.g. number of occurrences of the word 'blue' in a text.

Overview of common statistical procedures and their R commands

For further details see handouts. In code examples, X is the name of the first variable, Y the name of the second, Z of the third.

Tool	Applications	Assumptions	R
Correlation	correlations between 2 variables	two variables (ordinal scale or above) (this is Spearman's rho)	`cor.test(X,Y, method="spearman")`
		two interval/ratio variables; normally distributed (this is Pearson's r)	`cor.test(X,Y)`

Partial correlation	correlations between 2 variables while controlling for a third variable	three interval/ratio variables; normally distributed (if using Pearson's r). In the R command, X is the first variable, Y the second, and Z is the one that needs to be controlled for. Here also, Spearman's rho can be calculated if normality doesn't hold by specifying method as spearman	`install.packages("ppcor") ; library("ppcor")`
			
			`pcor.test(X, Y, Z)`
			or
			`pcor.test(X, Y, Z, method="spearman")`

Chi-squared for goodness of fit	comparing frequencies to see if they differ	one categorical variable = one variable holding frequency counts	`chisq.test(X)`
Chi-squared for independence	to check if frequencies in cross tables are independent	We need to be able to assemble a contingency table with real counted frequencies of occurrence of two categorical variables	`matrix(c(650,233,392,623) ,nrow=2)->TABLE; chisq.test(TABLE)`

one-sample t-test	compare sample mean to a known mean	one interval/ratio variable; normally distributed (X) and one known mean of a normally distributed interval/ratio variable (M)	`t.test(X, mu=M)`
t-test (independent)	compare means of two independent samples	two interval/ratio variables; normally distributed = one interval/ratio variable (normally distr.) and one binary variable (w/ 2 categories)	`t.test(X,Y)`
t-test (paired)	compare means of two samples with paired values	two interval/ratio variables; normally distributed (= 1 int/ratio & 1 binary); values in variable X are paired up with values in variable Y	`t.test(X,Y,paired=T)`
one-way ANOVA	compare means of three or more independent variables	three or more interval/ratio variables; normally distributed (= 1 int/ratio & 1 categorical variable with at least 3 categories)	`aov(X~Y, data=NAME_OF_DATAFRAME)->result; anova(result); TukeyHSD(result)`

two-way ANOVA	compare means of 3+ variables, classified differently	sets of three or more interval/ratio variables; normally distributed (= 1 int/ratio & 2+ categorical variables with at least 2 categories)	`install.packages("car"); library("car");`
			
			`aov(X ~ Y*Z, data=DATAFRAME)->result; Anova(result, type="III")`

Simple linear regression	predicting values of one variable using another variable (= showing that one variable causes values of the other to change)	typically we need two ratio/interval scale variables, but we can predict a ratio/interval variable from a categorical variable, too. Further assumptions apply.	`lm(X~Y)->model` `summary(model)`

Multiple linear regression	predicting values of a variable using several other variables	typically one ratio/interval variable as the dependent variable, then combinations of categorical and/or ratio/interval variables as independent variables. Further assumptions apply.	`lm(X~Y * Z)->model`
			
			`summary(model)`

In general, you get numbers in scientific notation as p-value, type:

`options(scipen=9999)`

then re-run the command.

Example Correlation

Example data: http://goo.gl/KxruY4 (name: Alcohol)

- learners of Esperanto at level B2
- given different amounts of alcohol
- then given a speaking test
- researcher wants to know if there is a relationship between
  - amount of alcohol
  - score in the speaking test

Analysis: significance of Spearman's rho

Assumptions: ordinal scale or better

Alternatives: if normally distributed we could use Pearson's r

R-command: **cor.test(Alcohol$ALCO,Alcohol$SCORE,method="spearman")**

R output:

```
   Spearman's rank correlation rho

data:  Alcohol$ALCO and Alcohol$SCORE
S = 307.7129, p-value = 3.242e-07
alternative hypothesis: true rho is not equal to 0
sample estimates:
     rho
0.847968
```

Reporting:

"There was a significant positive correlation ($r_s = 0.85$, $p < 0.001$) between amount of alcohol consumed and the scores on the speaking test."

Example chi-squared test for goodness of fit

Example data: http://goo.gl/tjiHA7 (name: Augen)

- frequencies of '*blaue[n] Augen*' (blue eyes) in a sample of German books from 1900 to 2000.
- source: Google Books (https://books.google.com/ngrams/)
- researcher wants to know if frequency fluctuations year on year are within the sort of fluctuation one would get due to chance

Assumptions: frequency data

R-command: **chisq.test(Augen$AUGEN)**

R output:

```
Chi-squared test for given probabilities

data:  AUGEN$AUGEN
X-squared = 194.169, df = 20, p-value < 2.2e-16
```

2.2e-16 is scientific notation, to convert: `options(scipen=9999)`, then re-run the command.

Reporting:

"A chi-squared test showed differences were statistically significant (chi-squared 194.169, df = 20, p < 0.001)."

Example chi-squared test for independence

Example data: in two samples of American speech, the following were found:

	-iŋ	-in
African American English	388	671
General American	530	221

- researcher wants to know if the pronunciation variation (-iŋ vs. –in) depends on the speech variety (AAE or GA) or is independent of it.

Assumptions: frequency data, no expected values should be smaller than 5.

R-commands: **matrix(c(388,530,671,221),nrow=2)-> ING; chisq.test(ING)**

R output:

```
Pearson's Chi-squared test with Yates' continuity
correction

data:  ING
X-squared = 201.0784, df = 1, p-value <
0.00000000000000022
```

Reporting:

"A chi-square test for independence (with Yates' continuity correction) indicated that occurrences of –ing and –in were not equally distributed across African American English and General American (chi-squared 201.08, df = 1, p < 0.001)."

Example t-test

Example data: http://goo.gl/waCNtA (name: Formant)

- frequencies in Hz of the first formant (F1) of male and female subjects
- researcher wants to know if there is a difference in F1 frequencies between females and males
- data taken from Gries (2013)

Analysis: t-test for independent samples

Assumptions: normal distribution, interval scale variables, no paired data

Alternatives: if NOT normally distributed, a Mann-Whitney U Test (aka Wilcoxon test) can be used.

R-command: **t.test(Formant$HZ_F1[Formant$SEX == "M"], Formant$HZ_F1[Formant$SEX == "F"])**

R output:

```
   Welch Two Sample t-test

data:  Formant$HZ_F1[Formant$SEX == "M"] and
Formant$HZ_F1[Formant$SEX == "F"]
t = -2.4416, df = 112.195, p-value = 0.01619
alternative hypothesis: true difference in means is not equal0
95 percent confidence interval:
 -80.758016  -8.403651
sample estimates:
mean of x mean of y
 484.2740  528.8548
```

Reporting:

“A t-test showed that the mean F1 frequency of males (M = 484.3, SD = 87.9) was significantly different from that of females (M = 528.9, SD = 110.8), t(112) = 2.44, p = 0.0162.”

Example one-way ANOVA with post-hoc test

Example data: http://goo.gl/qqnUAi (name: Reaction)

- reaction times in word recognition task for words of 3 different levels of familiarity
- researcher wants to know if the reaction times are different for words of differing familiarity

Assumptions: normal distribution of reaction times

Alternatives: if normally distributed we could use a Kruskal Wallis test

R-commands: **aov(RT~FAMILIARITY, data=Reaction)->results; anova(results)**

R output:

```
  Analysis of Variance Table

Response: RT
            Df Sum Sq Mean Sq F value    Pr(>F)
FAMILIARITY  2  33553 16776.6   7.982 0.0009481 ***
Residuals   52 109294  2101.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Reporting:

"An ANOVA showed that reaction times differed significantly by familiarity, $F(2,52) = 7.98$, $p < 0.001$."

Post-hoc test: Tukey HSD (or pairwise t-tests with Bonferroni correction of sig.-level)

R-command: **TukeyHSD(results)**

R output:

```
    Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = RT ~ FAMILIARITY, data = Reaction)

$FAMILIARITY
            diff       lwr       upr     p adj
lo-hi   71.90008  26.74512 117.05503 0.0009599
med-hi  22.25916 -15.34568  59.86401 0.3341263
med-lo -49.64091 -87.24575 -12.03607 0.0068048
```

Reporting:

"An A Tukey HSD post-hoc test showed significant differences between low and high familiarity ($p < 0.001$) as well as low and medium familiarity ($p = 0.007$), but the difference between medium and high familiarity was not significant ($p = 0.334$)."

Example simple linear regression

Example data: http://goo.gl/KxruY4 (name: Alcohol)

- learners of Esperanto at level B2
- given different amounts of alcohol
- then given a speaking test
- researcher wants to know if we can predict the test scores based on the the amount of alcohol

Assumptions: linear relationship between X and Y, errors must be normally distributed

R-command: **lm(Alcohol$SCORE~Alcohol$ALCO)-> result; summary(result)**

R output:

```
   Call:
lm(formula = Alcohol$SCORE ~ Alcohol$ALCO)

Residuals:
    Min      1Q  Median      3Q     Max
-6.7855 -1.8190 -0.6919  1.6810  8.1877

Coefficients:
             Estimate Std. Error t value   Pr(>|t|)
(Intercept)    6.6785     1.3367   4.996 0.00006047 ***
Alcohol$ALCO   1.0134     0.1497   6.772 0.00000107 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1

Residual standard error: 3.146 on 21 degrees of freedom
Multiple R-squared:  0.6859,  Adjusted R-squared:  0.6709
F-statistic: 45.85 on 1 and 21 DF,  p-value: 0.000001068
```

Testing assumptions: plot to see if it's linear (should have done that anyways); plot residuals (errors): `hist(result$residuals)` or test if error (residuals) are normally distributed using a Shapiro-Wilk test:

`shapiro.test(result$residuals)` – if this test is NOT significant, we should be fine because the residuals are then NOT distributed significantly differently from normal.

Reporting:

"A simple linear regression showed that the amount of alcohol predicted scores in the speaking test (adjusted $R^2 = 0.671$, df = 21, $p < 0.001$)"

Example multiple linear regression

Example data: http://goo.gl/KxruY4 (name: Alcohol)

- learners of Esperanto at level B2
- given different amounts of alcohol
- then given a speaking test and a personality test showing the degree to which a subject has an extrovert personality (scores 1 to 20)
- the researcher wants to know if we can predict the test scores based on the amount of alcohol and scores on the extrovert personality test.

Assumptions: linear relationship between X and Y/Z, errors must be normally distributed

R-command: **lm(Alcohol$SCORE~Alcohol$ALCO+Alcohol$EXTR)-> result; summary(result)**

R output:

```
   Call:
lm(formula = Alcohol$SCORE ~ Alcohol$ALCO + Alcohol$EXTR)

Residuals:
    Min      1Q  Median      3Q     Max
-6.9958 -2.2081  0.0195  1.6808  7.8425

Coefficients:
             Estimate Std. Error t value   Pr(>|t|)
(Intercept)    5.3491     1.8512   2.890    0.00906 **
Alcohol$ALCO   1.0253     0.1498   6.843 0.00000119 ***
Alcohol$EXTR   0.1111     0.1072   1.036    0.31251
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1

Residual standard error: 3.14 on 20 degrees of freedom
Multiple R-squared:  0.7019,  Adjusted R-squared:  0.6721
F-statistic: 23.54 on 2 and 20 DF,  p-value: 0.000005544
```

Testing assumptions: same as simple linear regression

Reporting: "When speaking test scores were predicted using a multiple regression, it was found that the amount of alcohol consumed was a significant predictor ($p < .001$), but extroversion was not ($p = 0.31$). The overall model fit was adjusted $R^2 = 0.672$, $df = 2,21$, $p < 0.001$."