ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

Direct calculation of out-of-sample predictions in multi-class kernel FDA

Matthias Treder

School of Computer Science & Informatics - Cardiff University
Cardiff CF24 3AA - United Kingdom

Abstract. After a two-class kernel Fisher Discriminant Analysis (KFDA) has been trained on the full dataset, matrix inverse updates allow for the direct calculation of out-of-sample predictions for different test sets. Here, this approach is extended to the multi-class case by casting KFDA in an Optimal Scoring framework. In simulations using 10-fold cross-validation and permutation tests the approach is shown to be more than 1000x faster than retraining the classifier in each fold. Direct out-of-sample predictions can be useful on large datasets and in studies with many training-testing iterations.

1 Introduction

For two-class kernel FDA (KFDA), a two-stage analytical approach allows for the direct calculation of out-of-sample predictions without explicitly training the model on the training data [1, 2]. In the first stage, a model is trained on the whole dataset and in-sample predictions are calculated for all instances. In the second stage, these predictions are updated to out-of-sample predictions via multiplication with a submatrix of a projection matrix. The approach is particularly efficient if the test sets are relatively small, e.g. in 10-fold cross-validation. Additional computational benefits arise in permutation tests, since the kernel matrix needs to be inverted only once irrespective of the number of permutations.

Here, this approach is extended to the multi-class case by casting multi-class FDA in a regression framework using its relationship to Canonical Correlation Analysis (CCA) and Optimal Scoring [3]. The paper is structured as follows. First, the direct calculation of out-of-sample predictions for two-class KFDA is reviewed. Subsequently, multi-class FDA is rewritten as an Optimal Scoring (OS) problem. Using OS, the direct calculation of out-of-sample predictions for multi-class KFDA is derived and the complexity of the algorithm is determined. Cross-validation and permutation testing experiments are conducted to compare the direct approach to the standard approach (retraining the classifier on each training fold).

2 Method

2.1 Direct out-of-sample predictions for two-class KFDA

For two classes, the direct approach can be developed by casting non-kernelised FDA as a linear regression problem. Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ be the predictor matrix (n

= number of instances; p = number of predictors), and $\mathbf{y} \in \mathbb{R}^n$ be the class labels. Let $\widetilde{\mathbf{X}} = [\mathbf{X}, \mathbf{1}_n]$ be the predictor matrix augmented with a column of ones corresponding to the bias term. Let $\mathbf{H} \in \mathbb{R}^{n\times n}$ be the regularized 'hat' matrix that maps $\mathbf{y}$ onto the *in-sample discriminant scores* $\hat{\mathbf{y}}^{\text{in}} = \mathbf{Hy}$. It is calculated as $\mathbf{H} = \widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}} + \lambda\mathbf{I}_{p+1})^{-1}\widetilde{\mathbf{X}}^\top = \widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\top(\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\top + \lambda\mathbf{I}_n)^{-1}$ where $\lambda \in \mathbb{R}$ is the regularization term and $\mathbf{I}_{p+1}$ and $\mathbf{I}_n$ are identity matrices of size n or $(p + 1)$. The left formulation is more efficient if $n > p + 1$ and vice versa. Let the subscript Te refer to the instances in the test set and $t < n$ be its size. As shown in [1, 2], the *out-of-sample discriminant scores* on the test set, denoted as $\hat{\mathbf{y}}^{\text{out}}_{\text{Te}} \in \mathbb{R}^t$, can be calculated directly using

$$\hat{\mathbf{y}}^{\text{out}}_{\text{Te}} = (\mathbf{I} - \mathbf{H}_{\text{Te}})^{-1}\,(\hat{\mathbf{y}}^{\text{in}}_{\text{Te}} - \mathbf{H}_{\text{Te}}\mathbf{y}_{\text{Te}}) \tag{1}$$

where $\mathbf{H}_{\text{Te}}$ is a matrix containing the rows and columns of $\mathbf{H}$ that correspond to the test instances. This result directly generalizes to KFDA[1]. Let $\mathbf{K} \in \mathbb{R}^{n\times n}$ be a kernel matrix. The equivalent of the 'hat' matrix in KFDA is the matrix $\mathbf{G} = \mathbf{K}(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}$ and the out-of-sample discriminant scores on the test set are given by $\hat{\mathbf{y}}^{\text{out}}_{\text{Te}} = (\mathbf{I} - \mathbf{G}_{\text{Te}})^{-1}\,(\hat{\mathbf{y}}^{\text{in}}_{\text{Te}} - \mathbf{G}_{\text{Te}}\mathbf{y}_{\text{Te}})$.

2.2 Multi-class FDA as Optimal Scoring

In standard multi-class FDA with c classes, a new instance gets assigned to the closest class centroid in the subspace spanned by the columns of $\mathbf{W} \in \mathbb{R}^{p\times(c-1)}$ [3, 4]. $\mathbf{W}$ is the solution to the generalised eigenvalue problem $\mathbf{S}_b\,\mathbf{W} = \mathbf{S}_w\,\mathbf{W}\mathbf{\Lambda}$, where $\mathbf{\Lambda}$ is a diagonal matrix of eigenvalues and $\mathbf{S}_b$ and $\mathbf{S}_w$ are defined as $\mathbf{S}_b = \sum_{j\,\in\{1,2,\ldots,c\}} n_j\,(\mathbf{m}_j - \overline{\mathbf{m}})(\mathbf{m}_j - \overline{\mathbf{m}})^\top$ and $\mathbf{S}_w = \sum_{j\,\in\{1,2,\ldots,c\}}\sum_{i\in\mathcal{C}_j}(\mathbf{x}_i - \mathbf{m}_j)(\mathbf{x}_i - \mathbf{m}_j)^\top$. Here, n_j is the number of instances in class j, $\mathbf{x}_i$ is the i-th instance, $\mathbf{m}_j$ is the j-th class mean, $\overline{\mathbf{m}}$ is the sample mean, and $\mathcal{C}_j$ is the set of indices of instances in class j. $\mathbf{W}$ is scaled such that $\mathbf{W}^\top\mathbf{S}_w\mathbf{W} = \mathbf{I}$.

For more than two classes FDA is not equivalent to linear regression, but it is equivalent to Optimal Scoring (OS) introduced next [3]. Let $\mathbf{Y} \in \mathbb{R}^{n\times c}$ be the class indicator matrix whose (i,j)-th element is 1 if instance i belongs to class j and 0 otherwise. Let $\boldsymbol{\beta} \in \mathbb{R}^{p+1}$ be the regression weights, $\boldsymbol{\theta} \in \mathbb{R}^c$ be the vector of optimal scores, and Tr refer to the rows or columns corresponding to the training set. Then the response vector of optimal scores on the training data can be written as $\mathbf{Y}_{\text{Tr}}\,\boldsymbol{\theta}$, and the Optimal Scoring problem is given by

$$\underset{\boldsymbol{\beta},\theta}{\arg\min}\; ||\widetilde{\mathbf{X}}_{\text{Tr}}\,\boldsymbol{\beta} - \mathbf{Y}_{\text{Tr}}\,\boldsymbol{\theta}||_2^2 \quad \text{(Optimal Scoring)} \tag{2}$$

where $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ are jointly optimised. The additional constraint $n^{-1}||\mathbf{Y}_{\text{Tr}}\,\boldsymbol{\theta}||^2 = 1$ avoids trivial solutions. [3] show that this optimisation problem can be broken up into two successive steps.

Step 1 (regression): A multivariate regression is performed on $\mathbf{Y}_{\mathrm{Tr}}$, yielding $\widetilde{\mathbf{B}} = \arg\min ||\widetilde{\mathbf{X}}_{\mathrm{Tr}}\,\mathbf{B} - \mathbf{Y}_{\mathrm{Tr}}||_F^2$, where $||\cdot||_F$ is the Frobenius norm and $\widetilde{\mathbf{B}} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_c]$. Denote the regression scores for training and test sets as $\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Tr}} = \widetilde{\mathbf{X}}_{\mathrm{Tr}}\,\widetilde{\mathbf{B}}$ and $\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Te}} = \widetilde{\mathbf{X}}_{\mathrm{Te}}\,\widetilde{\mathbf{B}}$. Unlike for two classes, an additional rotation/scaling step is required to transform these regression scores into discriminant scores.

Step 2 (rotation and scaling): An eigendecomposition of $(\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Tr}})^\top \mathbf{Y}_{\mathrm{Tr}}$ is performed to find the optimal score vector. Let $\boldsymbol{\Theta} \in \mathbb{R}^{c\times(c-1)}$ be the eigenvectors, also called optimal scores, where the column corresponding to the trivial eigenvalue 0 (if $\mathbf{X}_{\mathrm{Tr}}$ is centered) or 1 (otherwise) has been removed. Let $\alpha_1^2, \alpha_2^2, ..., \alpha_{c-1}^2$ be the corresponding eigenvalues. Let $\mathbf{B}$ be the submatrix of $\widetilde{\mathbf{B}}$ with the intercept omitted. Then the columns of $\mathbf{B}\boldsymbol{\Theta}$ point in the same directions as the discriminant weights obtained in multi-class FDA but their scaling differs. To scale the weights, they are right-multiplied with the diagonal matrix $\mathbf{D}_{ii} = \sqrt{\alpha_i^2(1-\alpha_i^2)/N}$. The normalisation $\sqrt{N}^{-1}$ does not appear in the original definition of $\mathbf{D}$ ([3], p. 83) but is necessary because the within-class scatter matrix has been used here whereas [3] use the covariance matrix. The relationship between the multi-class FDA weights $\mathbf{W}$ and the Optimal Scoring results is then given by

$$\mathbf{W} = \mathbf{B}\boldsymbol{\Theta}\mathbf{D}. \tag{3}$$

2.3 Direct out-of-sample predictions for multi-class KFDA

In the Optimal Scoring formulation of FDA, the multivariate regression in step 1 can be skipped by adopting Eq. (1) and calculating the regression scores directly. A similar derivation additionally yields the scores on the training data $\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Tr}}$,

$$\begin{aligned}\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Te}} &= (\mathbf{I} - \mathbf{H}_{\mathrm{Te}})^{-1}\,(\hat{\mathbf{Y}}^{\mathrm{in}}_{\mathrm{Te}} - \mathbf{H}_{\mathrm{Te}}\mathbf{Y}_{\mathrm{Te}})\\ \hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Tr}} &= \hat{\mathbf{Y}}^{\mathrm{in}}_{\mathrm{Tr}} - \mathbf{H}_{\mathrm{Tr,Te}}\,\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Te}}\end{aligned} \tag{4}$$

where $\hat{\mathbf{Y}}^{\mathrm{in}} = \mathbf{HY}$ is the matrix of regression scores using all data as training data and $\hat{\mathbf{Y}}^{\mathrm{in}}_{\mathrm{Tr}}$ and $\hat{\mathbf{Y}}^{\mathrm{in}}_{\mathrm{Te}}$ are its submatrices corresponding to training and test set. Step 2 cannot be skipped, but it involves eigenanalysis of a $c \times c$ matrix which is cheap if the number of classes c is small (e.g. < 10). The out-of-sample discriminant scores are then obtained via Eq. (3) by projecting the test data onto $\mathbf{W}$

$$\begin{aligned}\mathbf{X}_{\mathrm{Te}}\,\mathbf{W} &= \mathbf{X}_{\mathrm{Te}}\,\mathbf{B}\boldsymbol{\Theta}\mathbf{D}\\ \Leftrightarrow\ \hat{\mathbf{Y}}^{\mathrm{out}}_{\mathrm{Te}} &= \hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Te}}\,\boldsymbol{\Theta}\mathbf{D}\end{aligned} \tag{5}$$

where $\hat{\mathbf{Y}}^{\mathrm{out}}_{\mathrm{Te}} \in \mathbb{R}^{t\times(c-1)}$ is the matrix of desired out-of-sample discriminant scores. Eq. (5) illustrates that the scores are obtained by rotating $\hat{\mathbf{Y}}^{\mathrm{reg}}_{\mathrm{Te}}$ using $\boldsymbol{\Theta}$ and scaling it using $\mathbf{D}$. Lastly, the equivalence between CCA and multi-class FDA also applies in the kernel case: $\mathbf{H}$ is simply replaced by its kernelised version $\mathbf{G}$ [5]. These results are compiled in Algorithm 1.

Algorithm 1 Direct out-of-sample predictions for multi-class KFDA

```
Input: kernel 'hat' matrix $\mathbf{G}$, class indicator matrix $\mathbf{Y}$
$\hat{\mathbf{Y}}^{\text{in}} \leftarrow \mathbf{GY}$
for all test sets Te do
    (step 1)
    $\hat{\mathbf{Y}}^{\text{reg}}_{\text{Te}} \leftarrow (\mathbf{I} - \mathbf{G}_{\text{Te}})^{-1} (\hat{\mathbf{Y}}^{\text{in}}_{\text{Te}} - \mathbf{G}_{\text{Te}}\mathbf{Y}_{\text{Te}})$
    $\hat{\mathbf{Y}}^{\text{reg}}_{\text{Tr}} \leftarrow \hat{\mathbf{Y}}^{\text{in}}_{\text{Tr}} - \mathbf{G}_{\text{Tr,Te}}\, \hat{\mathbf{Y}}^{\text{reg}}_{\text{Te}}$
    (step 2)
    $(\mathbf{\Theta}, \mathbf{D}) \leftarrow \texttt{eig}((\hat{\mathbf{Y}}^{\text{reg}}_{\text{Tr}})^\top \mathbf{Y}_{\text{Tr}}/N_{\text{Tr}})$
    $\hat{\mathbf{Y}}^{\text{out}}_{\text{Te}} \leftarrow \hat{\mathbf{Y}}^{\text{reg}}_{\text{Te}} \mathbf{\Theta D}$
end for
Output: out-of-sample discriminant scores for different test sets $\{\hat{\mathbf{Y}}^{\text{out}}_{\text{Te}}\}$
```

2.4 Computational complexity

The asymptotic computational complexity for training KFDA is quantified as number of floating point operations. Let n_{Tr} and n_{Te} be the number of training and test instances, respectively. In the standard KFDA approach, the complexity of calculating the kernel matrix is at least $\mathcal{O}(n_{\text{Tr}}^2 p)$, depending on the kernel. This is followed by an eigenvalue decomposition of the centered kernel matrix $\mathcal{O}(n_{\text{Tr}}^3)$. In cross-validation, this is repeated k times for an overall complexity of $\mathcal{O}(kn_{\text{Tr}}^2 p + kn_{\text{Tr}}^3)$.

In the direct approach, calculating and inverting the full kernel matrix costs $\mathcal{O}(n^2p + n^3)$. This is the most expensive step, but it is required only once. Calculating the regression fits $\hat{\mathbf{Y}}^{\text{reg}}_{\text{Tr}}$ and $\hat{\mathbf{Y}}^{\text{reg}}_{\text{Te}}$ in every cross-validation iteration costs $\mathcal{O}(kn_{\text{Te}}^3 + kn_{\text{Tr}}n_{\text{Te}}^2)$. This is followed by the calculation and eigendecomposition of $(\hat{\mathbf{Y}}^{\text{reg}}_{\text{Tr}})^\top \mathbf{Y}_{\text{Tr}}$, $\mathcal{O}(kc^2 n_{\text{Tr}} + c^3)$. Finally the discriminant scores are calculated, $\mathcal{O}(kc^2 n_{\text{Te}})$. Assuming that c is fixed this yields an overall complexity of $\mathcal{O}(n^2p + n^3 + kn_{\text{Te}}^3 + kn_{\text{Tr}}n_{\text{Te}}^2)$.

2.5 Experiments and results

Is the direct calculation out-of-sample scores faster than the standard approach? To answer this question, experiments were conducted using multi-class FDA with simulated data. Data was sampled from a multivariate normal distribution. Class centroids were randomly placed on the surface of a unit hypersphere. A common covariance matrix was sampled from a Wishart distribution. Simulations were performed for cross-validation and permutation tests.

Cross-validation. 10-fold cross-validation was used with data being split into 5 classes or 10 classes with equal class proportions. The number of instances was either 100 or 1000, the number of features was varied from 10 to 1000 in logarithmic steps.

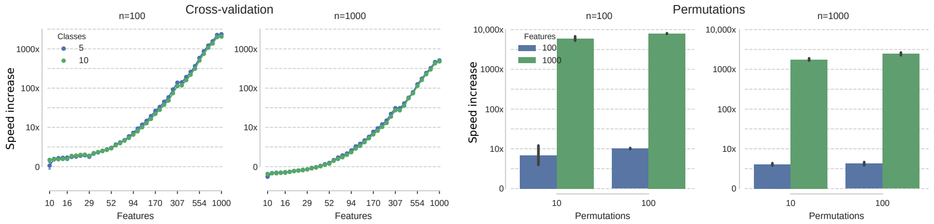

Fig. 1: Results of the simulations. On the y-axis, speed increase by the direct approach is plotted on a logarithmic scale. Left plots: Speed increase for cross-validation for 5 or 10 classes as a function of the number of features. Right plots: Permutations.

Permutation tests. Here, cross-validation was repeated multiple times using shuffled class labels. The number of features was fixed to 100 or 1000. The number of permutations was limited to 10 or 100 to keep overall computation time tractable. Since the projection matrix ($\mathbf{G}$ or $\mathbf{H}$) is independent of the class labels and only needs to be calculated once, large speed gains are expected.

For every combination of parameters, the simulation was repeated 20 times for cross-validation and 10 times for the permutations. In every iteration, the same random data and folds were used in both approaches to increase comparability. As target measure, speed increase was used. It was defined as the time required by the standard approach divided by the time required by the direct approach. Analyses were performed in MATLAB (Natick, USA) using a Thinkpad X1 Carbon with 16 GB of RAM and an Intel Core i7-6600U CPU @ 2.60GHz × 4 processor. Results are depicted in Figure 1.

Cross-validation. An analysis of variance (ANOVA) revealed significant effects of n ($F = 1023.97; p < .001$), features ($F = 38270.22; p < .001$) but not classes ($p = .15$). There was a significant features × n interaction ($F = 125.74; p < .001$) signifying a smaller effect of features for larger n.

Permutation tests. An ANOVA revealed significant effects of n ($F = 366.2; p < .001$), permutations ($F = 27.4; p < .001$), and features ($F = 16970.31; p < .001$). There was a significant n × features interaction ($F = 24.93; p < .001$) signifying a smaller effect of features for larger n.

3 Discussion

A direct approach for calculating out-of-sample predictions in multi-class KFDA has been presented. Simulations revealed a speed gain of the direct approach compared to the standard approach (retraining a classifier on every fold). Cross-validation was up to 1000x faster and permutation testing was up to 10,000x faster.

Complexity. In the kernelised case, the standard approach requires an eigenvalue decomposition of a $n_{\text{Tr}} \times n_{\text{Tr}}$ matrix (n_{Tr} = size of training set). In contrast, the direct calculation requires the inversion of a $n_{\text{Te}} \times n_{\text{Te}}$ matrix (n_{Te} = size of test set). Additionally, an inversion of a $n \times n$ matrix ($n =$ total size of data set) is needed once, irrespective of the number of train/test folds. Hence, large speed gains are possible if the test set is small relative to the training set, e.g. in k-fold cross-validation with $k \geq 5$.

Permutation tests. A particularly appealing application is permutation tests, a popular non-parametric statistical tool [6, 7]. The computationally most expensive part of the direct approach is the inversion of the $n \times n$ kernel matrix. However, since the kernel matrix is independent of the class labels, this inversion only needs to be performed once for the first permutation. The result can then be stored and re-used for all other permutations.

What if n is large? If the kernel matrix is too large to be stored in memory or the inversion is too costly at $\mathcal{O}(n^3)$, a simple remedy is to reduce the size of the kernel matrix by subsampling or instance averaging in kernel space [8]. Alternatively, the Nyström method yields a low-rank approximation to the kernel matrix $\mathbf{K}_r = \mathbf{L}\mathbf{L}^\top$ with $\mathbf{L} \in \mathbb{R}^{n \times r}$, $r \ll n$. This allows for the kernel 'hat' matrix to be approximated as $\mathbf{G}_r = \lambda^{-1}\,\mathbf{L}\,\left[\mathbf{I}_r - \mathbf{L}^\top\mathbf{L}\,(\mathbf{L}^\top\mathbf{L} + \lambda\mathbf{I}_r)^{-1}\right]\,\mathbf{L}^\top =: \lambda^{-1}\,\mathbf{L}\mathbf{R}\mathbf{L}^\top$ (compare Eq. (5) in [9]). The inversion required for calculating $\mathbf{G}_r$ has a complexity of $\mathcal{O}(r^3)$ and $\mathbf{R} \in \mathbb{R}^{r \times r}$ can be stored in memory for r sufficiently small. The submatrices of $\mathbf{G}_r$ necessary for the approach developed in this paper can be extracted efficiently by selecting the respective rows of $\mathbf{L}$, without ever requiring the full $n \times n$ matrix.

References

[1] Tapio Pahikkala, Jorma Boberg, and Tapio Salakoski. Fast n-Fold Cross-Validation for Regularized Least-Squares. In *Proceedings of SCAI'0*, pages 83–90, 2006.

[2] R Bharat Rao, Glenn Fung, and Romer Rosales. On the Dangers of Cross-Validation. An Experimental Evaluation. In *Proceedings of the 2008 SIAM International Conference on Data Mining*, pages 588–596, 2008.

[3] Trevor Hastie, Andreas Buja, and Robert Tibshirani. Penalized Discriminant Analysis. *The Annals of Statistics*, 23(1):73–102, 2 1995.

[4] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The Elements of Statistical Learning. In *The Elements of Statistical Learning*. Springer New York Inc., New York, NY, USA, 2009.

[5] M. Yamada, A. Pezeshki, and M.R. Azimi-Sadjadi. Relation between kernel CCA and kernel FDA. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005*, volume 2, pages 226–231. IEEE, 2005.

[6] Carsten Allefeld, Kai Görgen, and John-Dylan Haynes. Valid population inference for information-based imaging: From the second-level t-test to prevalence inference. *NeuroImage*, 141:378–392, 11 2016.

[7] Markus Ojala and Gemma C Garriga. Permutation Tests for Studying Classifier Performance. *Journal of Machine Learning Research*, 11:1833–1863, 2010.

[8] Matthias S. Treder. Improving SNR and Reducing Training Time of Classifiers in Large Datasets via Kernel Averaging. *Lecture Notes in Computer Science*, 11309:239–248, 2018.

[9] Farhad Pourkamali-Anaraki, Stephen Becker, and Michael B. Wakin. Randomized Clustered Nystrom for Large-Scale Kernel Machines. In *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3960–3067, 4 2018.