PERGAMON

Pattern Recognition 36 (2003) 2585–2592

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Efficient leave-one-out cross-validation of kernel Fisher discriminant classifiers

Gavin C. Cawley*, Nicola L.C. Talbot

School of Information Systems, University of East Anglia, Norwich NR4 7TJ, UK

Received 9 September 2002; accepted 16 April 2003

Abstract

Mika et al. (in: Neural Network for Signal Processing, Vol. IX, IEEE Press, New York, 1999; pp. 41–48) apply the "*kernel trick*" to obtain a non-linear variant of Fisher's linear discriminant analysis method, demonstrating state-of-the-art performance on a range of benchmark data sets. We show that leave-one-out cross-validation of kernel Fisher discriminant classifiers can be implemented with a computational complexity of only $\mathcal{O}(\ell^3)$ operations rather than the $\mathcal{O}(\ell^4)$ of a naïve implementation, where ℓ is the number of training patterns. Leave-one-out cross-validation then becomes an attractive means of model selection in large-scale applications of kernel Fisher discriminant analysis, being significantly faster than conventional k-fold cross-validation procedures commonly used.

Keywords: Model selection; Cross-validation; Kernel Fisher discriminant analysis

1. Introduction

The now familiar "*kernel trick*" [1] has been used to derive non-linear variants of many linear methods borrowed from classical statistics (e.g. Refs. [2,3]), including ridge-regression [4], principal component analysis [5] and canonical correlation analysis [6] as well as more recent developments such as the maximal margin classifier [7] (giving rise to the support vector machine [8]). These methods have come to be known collectively as "*kernel machines*" and have attracted considerable interest in the machine learning research community due to a combination of conceptual elegance, mathematical tractability and state-of-the-art performance on real world as well as benchmark problems. One such method, the kernel Fisher discriminant (KFD) classifier [1], implements the well-known Fisher linear discriminant [9] in a feature space induced by a Mercer kernel [10], giving rise to a non-linear pattern recognition method demonstrating an impressive level of performance on a range of benchmark data sets. An important advantage of many kernel methods, including the kernel Fisher discriminant, is that the optimal model parameters are given by the solution of a convex optimisation problem with a single, global optimum. However, optimal generalisation still depends on the selection of an appropriate kernel function and the values of regularisation [11] and kernel parameters, an activity known as model selection. For kernel Fisher discriminant networks this is most frequently performed by a lengthy optimisation of a simple k-fold cross-validation estimate of an appropriate performance statistic. In this paper, we set out a fast implementation of the leave-one-out cross-validation procedure, providing a more efficient means of model selection for kernel Fisher discriminant classifiers than the conventional k-fold cross-validation approach and evaluate its performance on a range of standard benchmark machine learning problems.

The remainder of this paper is structured as follows: Section 2 provides a summary of the strengths and limitations of leave-one-out cross-validation for the purposes of model selection. Section 3 gives a full description of kernel Fisher discriminant analysis, establishing the notation used throughout. An efficient implementation of the leave-one-out cross-validation procedure for kernel Fisher

* Corresponding author. Tel.: +44 (0)1603 593258; fax: +44 (0)1603 593345.

E-mail address: gcc@sys.uea.ac.uk (G.C. Cawley).

0031-3203/03/$30.00
doi:10.1016/S0031-3203(03)00136-5

discriminant networks is given in Section 4. A comparison of model selection procedures based on k-fold and leave-one-out cross-validation schemes, over a range of standard benchmark learning problems, is then presented in Section 5. Finally the works are summarised in Section 6.

2. Strengths and limitations of leave-one-out cross-validation

Cross-validation [12] is often used to estimate the generalisation ability of a statistical classifier (i.e. the performance on previously unseen data). Under cross-validation, the available data are divided into k disjoint sets; k models are then trained, each on a different combination of $k-1$ partitions and tested on the remaining partition. The k-fold cross-validation estimate of a given performance statistic is then simply the mean of that statistic evaluated for each of the k models over the corresponding test partitions of the data. Cross-validation thus makes good use of the available data as each pattern used is used both as training and test data. Cross-validation is therefore especially useful where the amount of available data is insufficient to form the usual training, validation and test partitions required for split-sample training, each of which adequately represents the true distribution of patterns belonging to each class. The most extreme form of cross-validation, where k is equal to the number of training patterns is known as leave-one-out cross-validation, and has been widely studied due to its mathematical simplicity.

A property of the leave-one-out cross-validation estimator, often cited as being highly attractive for the purposes of model selection (e.g. Refs. [13,14]), is that it provides an almost unbiased estimate of the generalisation ability of a classifier:

Lemma 1 (Bias of leave-one-out cross-validation [15,16]). *Leave-one-out cross-validation gives an almost unbiased estimate of the probability of test error, i.e.*

$$E\{p_{error}^{\ell-1}\} = E\left\{\frac{\mathscr{L}(\boldsymbol{x}_1, y_1, \boldsymbol{x}_2, y_2, \ldots, \boldsymbol{x}_\ell, y_\ell)}{\ell}\right\}, \tag{1}$$

where $p_{error}^{\ell-1}$ is the probability of test error for a classifier trained on a sample of size $\ell-1$ and $\mathscr{L}(\boldsymbol{x}_1, y_1, \boldsymbol{x}_2, y_2, \ldots, \boldsymbol{x}_\ell, y_\ell)$ measures the number of leave-one-out errors for a classifier trained on a set of input-target pairs, $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{\ell}$, of size ℓ. The leave-one-out estimator is almost unbiased in the sense that the expectations are taken over samples of size $\ell-1$ on the left-hand side of Eq. (1) and of size ℓ on the right.

However, a model selection criterion need not give an unbiased estimate of the generalisation performance. For example, adding a fixed constant to the leave-one-out estimator would not alter the outcome of the model selection procedure, but would no longer provide an unbiased estimate of the test error. The principal requirement of a practical model selection criterion is that it should be strongly correlated with the true generalisation error, such that the minimum of the selection criterion reliably coincides with the minimum of the true generalisation error.

Empirical studies have shown that in some cases model selection based on k-fold cross-validation out performs selection procedures based on the leave-one-out estimator as the latter is known to exhibit a comparatively high variance. For large data sets, however, it could be argued that the variances of k-fold and leave-one-out estimators are likely to be similar:

Lemma 2 (Variance of k-fold cross-validation [17]). *Assuming the training algorithm for a classifier system is stable with regard to the perturbation of the training data introduced during the cross-validation procedure (i.e. the perturbation of the training data does not change the decision rule obtained), the variance of the k-fold estimate of the accuracy of the inducer is independent of k.*

A straightforward corollary of Lemma 2 is that provided the data set is sufficiently large such that the inducer is stable, the variance of k-fold and leave-one-out cross-validation estimates coincide. Most kernel machines (including kernel Fisher discriminant analysis) are trained by minimising a regularised loss functional, comprised of a sum of independent terms representing the loss for each training pattern. It seems reasonable to suggest then, that such models will become stable for sufficiently large data sets, at least in the case of the leave-one-out estimator, as the effect of removing a single term from the loss functional becomes diminishingly small as the size of the training data becomes large.

Leave-one-out cross-validation is normally restricted to applications where the amount of training data available is severely limited, such that even a small perturbation of the training data is likely to result in a substantial change in the fitted model. In this case, it makes good sense to adopt a leave-one-out cross-validation strategy as it minimises the perturbation to the data in each trial. Leave-one-out cross-validation is rarely adopted in large-scale applications simply because it is computationally expensive. The training algorithms for kernel machines, including that for the kernel Fisher discriminant, typically have a computational complexity of $\mathcal{O}(\ell^3)$, where ℓ is the number of training patterns. In this case, the leave-one-out cross-validation process has a computational complexity of $\mathcal{O}(\ell^4)$, which quickly becomes impractical as the number of training patterns increases. Note however that minimising an upper bound on the leave-one-out error has proved an effective means of model selection for support vector machines (e.g. Refs. [13,14]).

Since there exist theoretical and experimental justification both for and against the use of leave-one-out cross-validation in model selection, we provide an experimental comparison of leave-one-out and k-fold cross-validation

in this study. We further demonstrate that, in the case of kernel Fisher discriminant models, the leave-one-out cross-validation procedure can be implemented with a computational complexity of only $\mathcal{O}(\ell^3)$ operations, the same as that of the basic training algorithm, and by extension of the k-fold cross-validation procedure. Experiments show that the proposed leave-one-out cross-validation process is actually *faster* than k-fold cross-validation (for any value of k), overcoming the prohibition against leave-one-out cross-validation in large-scale applications.

3. Kernel Fisher discriminant analysis

Assume we are given training data $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_\ell\} = \{\mathcal{X}_1, \mathcal{X}_2\} \subset \mathbb{R}^d$, where $\mathcal{X}_1 = \{\boldsymbol{x}_1^1, \boldsymbol{x}_2^1, \ldots, \boldsymbol{x}_{\ell_1}^1\}$ is a set of patterns belonging to class $\mathscr{C}_1$ and similarly $\mathcal{X}_2 = \{\boldsymbol{x}_1^2, \boldsymbol{x}_2^2, \ldots, \boldsymbol{x}_{\ell_2}^2\}$ is a set of patterns belonging to class $\mathscr{C}_2$; Fisher's linear discriminant (FLD) attempts to find a linear combination of input variables, $\boldsymbol{w} \cdot \boldsymbol{x}$, that maximises the average separation of the projections of points belonging to $\mathscr{C}_1$ and $\mathscr{C}_2$, whilst minimising the within class variance of the projections of those points. The Fisher discriminant is given by the vector $\boldsymbol{w}$ maximising

$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^{\mathrm{T}} \boldsymbol{S}_B \boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}} \boldsymbol{S}_W \boldsymbol{w}}, \tag{2}$$

where $\boldsymbol{S}_B$ is the between class scatter matrix $\boldsymbol{S}_B = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^{\mathrm{T}}$, $\boldsymbol{m}_j = \ell_j^{-1} \sum_{i=1}^{\ell_j} \boldsymbol{x}_i^j$ and $\boldsymbol{S}_W$ the within class scatter matrix

$$\boldsymbol{S}_W = \sum_{i \in \{1,2\}} \sum_{j=1}^{\ell_i} (\boldsymbol{x}_j^i - \boldsymbol{m}_i)(\boldsymbol{x}_j^i - \boldsymbol{m}_i)^{\mathrm{T}}.$$

The innovation introduced by Mika et al. [1] is to construct Fisher's linear discriminant in a fixed feature space $\mathscr{F}(\boldsymbol{\phi} : \mathcal{X} \to \mathscr{F})$ induced by a positive definite *Mercer* kernel $\mathscr{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defining the inner product $\mathscr{K}(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x}) \cdot \boldsymbol{\phi}(\boldsymbol{x}')$ (see e.g. Ref. [2]). Let the kernel matrices for the entire data set, $\boldsymbol{K}$, and for each class, $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ be defined as follows:

$$\boldsymbol{K} = [k_{ij} = \mathscr{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j=1}^{\ell} \quad \text{and}$$

$$\boldsymbol{K}_i = [k_{jk}^i = \mathscr{K}(\boldsymbol{x}_j, \boldsymbol{x}_k^i)]_{j,k=1}^{j=\ell, k=\ell_i}.$$

The theory of reproducing kernels [18,19] indicates that $\boldsymbol{w}$ can then be written as an expansion of the form

$$\boldsymbol{w} = \sum_{i=1}^{\ell} \alpha_i \boldsymbol{\phi}(\boldsymbol{x}_i). \tag{3}$$

The objective function (2) can also be written such that the data $\boldsymbol{x} \in \mathcal{X}$ appear only within inner products, giving

$$J(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{M} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{N} \boldsymbol{\alpha}}, \tag{4}$$

where $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1}^{\ell}$, $\boldsymbol{M} = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^{\mathrm{T}}$, $\boldsymbol{m}_i = \boldsymbol{K}_i \boldsymbol{u}_i$, $\boldsymbol{u}_i$ is a column vector containing ℓ_i elements with a common value of ℓ_i^{-1} and

$$\boldsymbol{N} = \sum_{i \in \{1,2\}} \boldsymbol{K}_i (\boldsymbol{I} - \boldsymbol{U}_i) \boldsymbol{K}_i^{\mathrm{T}},$$

where $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{U}_i$ is a matrix with all elements equal to ℓ_i^{-1}. The coefficients, $\boldsymbol{\alpha}$, of expansion (3) are then given by the leading eigenvector of $\boldsymbol{N}^{-1}\boldsymbol{M}$. Note that $\boldsymbol{N}$ is likely to be singular, or at best ill-conditioned, and so a regularised solution is obtained by substituting $\boldsymbol{N}_\mu = \boldsymbol{N} + \mu \boldsymbol{I}$, where μ is a regularisation constant. To complete the kernel Fisher discriminant classifier, $f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{\phi}(\boldsymbol{x}) + b$, the bias, b, is given by

$$b = -\boldsymbol{\alpha} \frac{\ell_1 \boldsymbol{M}_1 + \ell_2 \boldsymbol{M}_2}{\ell}.$$

Xu et al. [20] show that the parameters of the kernel Fisher discriminant classifier are also given by the solution of the following system of linear equations:

$$\begin{bmatrix} \boldsymbol{K}\boldsymbol{K} + \mu \boldsymbol{I} & \boldsymbol{K}\boldsymbol{1} \\ (\boldsymbol{K}\boldsymbol{1})^{\mathrm{T}} & \ell \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{1} \end{bmatrix} \boldsymbol{y}, \tag{5}$$

where $\boldsymbol{1}$ is a column vector of ℓ ones and $\boldsymbol{y}$ is a column vector with elements $y_i = \ell/\ell_j \forall i$: $\boldsymbol{x}_i \in \mathcal{X}_j$. This illustrates the similarities between the kernel Fisher discriminant and the least-squares support vector machine (LS-SVM) [21]. The kernel Fisher discriminant (KFD) classifier has been shown experimentally to demonstrate near state-of-the-art performance on a range of artificial and real-world benchmark data sets [1] and so is worthy of consideration for small to medium scale applications.

4. Efficient leave-one-out cross-validation

The system of linear equations (5) can be written more concisely in the form

$$\boldsymbol{p} = [\boldsymbol{R} + \boldsymbol{Z}^{\mathrm{T}} \boldsymbol{Z}]^{-1} \boldsymbol{Z}^{\mathrm{T}} \boldsymbol{y}, \tag{6}$$

where $\boldsymbol{Z} = [\boldsymbol{K}\ \boldsymbol{1}]$, $\boldsymbol{R} = \mathrm{diag}([\mu \boldsymbol{1}\ 0])$ and $\boldsymbol{p} = (\boldsymbol{\alpha}, b)$ (n.b. this is very similar to the set of *normal equations* to be solved in multi-variate linear regression [22]). At each step of the leave-one-out cross-validation procedure, a kernel Fisher discriminant classifier is constructed excluding a single training pattern from the data. The vector of model parameters, $\boldsymbol{p}_{(i)} = (\boldsymbol{\alpha}_{(i)}, b_{(i)})$ at the ith iteration is then given by the solution of a modified system of linear equations,

$$\boldsymbol{p}_{(i)} = \left[\boldsymbol{R} + \boldsymbol{Z}_{(i)}^{\mathrm{T}} \boldsymbol{Z}_{(i)}\right]^{-1} \boldsymbol{Z}_{(i)}^{\mathrm{T}} \boldsymbol{y},$$

where $\boldsymbol{Z}_{(i)}$ is the sub-matrix formed by omitting the ith row of $\boldsymbol{Z}$. Normally, the most computationally expensive step is the inversion of the matrix $\boldsymbol{C}_{(i)} = [\boldsymbol{R} + \boldsymbol{Z}_{(i)}^{\mathrm{T}} \boldsymbol{Z}_{(i)}]$, with a complexity of $\mathcal{O}(\ell^3)$ operations. Fortunately, $\boldsymbol{C}_{(i)}$ can be

written as a rank one modification of a matrix $\boldsymbol{C}$,

$$\boldsymbol{C}_{(i)} = [\boldsymbol{R}_{(i)} + \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z} - \boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}] = \left[\boldsymbol{C} - \boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}\right], \tag{7}$$

where $\boldsymbol{z}_i$ is the ith row of $\boldsymbol{Z}$. The following matrix inversion lemma then allows $\boldsymbol{C}_{(i)}^{-1}$ to be found in only $\mathcal{O}(\ell^2)$ operations, given that $\boldsymbol{C}^{-1}$ is already known:

Lemma 3 (Matrix inversion formula [23–28]). *Given an invertible matrix $\boldsymbol{A}$ and column vectors $\boldsymbol{u}$ and $\boldsymbol{v}$, then assuming $1 - \boldsymbol{v}^{\mathrm{T}}\boldsymbol{A}^{-1}\boldsymbol{u} \neq 0$,*

$$(\boldsymbol{A} + \boldsymbol{u}\boldsymbol{v}^{\mathrm{T}})^{-1} = \boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1}\boldsymbol{u}\boldsymbol{v}^{\mathrm{T}}\boldsymbol{A}^{-1}}{1 + \boldsymbol{v}^{\mathrm{T}}\boldsymbol{A}^{-1}\boldsymbol{u}}. \tag{8}$$

This is known as the Bartlett–Sherman–Woodbury–Morrison formula.

Applying the Bartlett–Sherman–Woodbury–Morrison formula to the matrix inversion problem given in Eq. (7), we have that

$$\boldsymbol{C}_{(i)}^{-1} = [\boldsymbol{C} - \boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}]^{-1} = \boldsymbol{C}^{-1} + \frac{\boldsymbol{C}^{-1}\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}}{1 - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{z}_i}.$$

The computational complexity of the leave-one-out cross-validation process is thus reduced to only $\mathcal{O}(\ell^3)$ operations, as l matrix inversions are required, at a computational complexity of $\mathcal{O}(\ell^2)$. This is the same as that of the basic training algorithm for the kernel Fisher discriminant classifier, and by extension the k-fold cross-validation procedure for these models.

4.1. A practical model selection criterion

For model selection purposes, we are not principally concerned with the values of the model parameters themselves, but only statistics such as the leave-one-out error rate

$$E = \frac{1}{\ell}\,\mathrm{card}\{i\colon\, y_i(\boldsymbol{w}_{(i)} \cdot \boldsymbol{\phi}(\boldsymbol{x}_i) + b_{(i)}) \leqslant 0\} \tag{9}$$

or equivalently

$$E = \frac{1}{\ell}\,\mathrm{card}\{i\colon\, \mathrm{sign}(y_i)\{r_{(i)}\}_i \leqslant -1\},$$

where $\{r_{(i)}\}_i = y_i - \boldsymbol{w}_{(i)} \cdot \boldsymbol{\phi}(\boldsymbol{x}_i) + b_{(i)}$ is the residual error for the ith training pattern during the ith iteration of the leave-one-out cross-validation procedure. Alternatively, since the kernel Fisher discriminant minimises a regularised sum of squares loss functional [20], the natural model selection criterion would be a leave-one-out estimate of the sum of squares error, i.e. Allen's PRESS (predicted residual sum of squares) statistic [29],

$$\mathrm{PRESS} = \sum_{i=1}^{\ell} \{\boldsymbol{r}_{(i)}\}_i^2. \tag{10}$$

Fortunately, it is possible to compute these residuals without explicitly evaluating the model parameters in each trial. It is relatively straightforward to show that

$$\{\boldsymbol{r}_{(i)}\}_i = \frac{r_i}{1 - h_{ii}},$$

(see Appendix A) where $r_i = y_i - f(\boldsymbol{x}_i)$ is the residual for the ith training pattern for a kernel Fisher discriminant model trained on the entire data set and $\boldsymbol{H} = \boldsymbol{Z}\boldsymbol{C}^{-1}\boldsymbol{Z}^{\mathrm{T}}$ is the *hat* matrix [22] of which h_{ii} is the ith element of the leading diagonal [30]. Allen's PRESS statistic can therefore be evaluated in closed form without explicit inversion of $\boldsymbol{C}_{(i)} \forall i \in \{1, 2, \ldots, \ell\}$, again with a computational complexity of only $\mathcal{O}(\ell^3)$. Note that this result is well known in the field of linear least-squares regression (e.g. Ref. [30]); again the "kernel trick" enables its use in a non-linear context.

5. Results

In this section we present an experimental comparison of efficient leave-one-out and conventional k-fold cross-validation procedures for model selection in training kernel Fisher discriminant classifiers, in terms of both computational complexity (efficiency) and in terms of the generalisation of the resulting kernel Fisher discriminant networks. The relative efficiency of the proposed approach is determined using a relatively large-scale synthetic learning task. A set of 13 real-world and synthetic benchmark datasets from the UCI repository [31] is used to evaluate the generalisation properties resulting from model selection schemes based on leave-one-out and k-fold cross-validation. An isotropic Gaussian kernel,

$$\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \exp\left\{-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right\},$$

is used in all experiments.

5.1. Computational complexity

The proposed approximate leave-one-out cross-validation method is evaluated over a series of randomly generated synthetic datasets, as shown in Fig. 1. In each case, approximately one-quarter of the data belong to class $\mathscr{C}_1$ and three-quarters to class $\mathscr{C}_2$. The patterns comprising class $\mathscr{C}_1$ are drawn from a bivariate normal distribution with zero mean and unit variance. The patterns forming class $\mathscr{C}_2$ form an annulus; the radii of the data are drawn from a normal distribution with a mean of 5 and unit variance, and the angles uniformly distributed. The data sets vary in size between 10 and 1000 patterns. Fig. 2 shows a graph of run-time as a function of the number of training patterns for fast and naïve leave-one-out and 10-fold cross-validation estimates of the test sum of squares error statistic. Clearly, the fast leave-one-out method is considerably faster and exhibits significantly better scaling properties than the naïve implementation of the leave-one-out estimator. For large

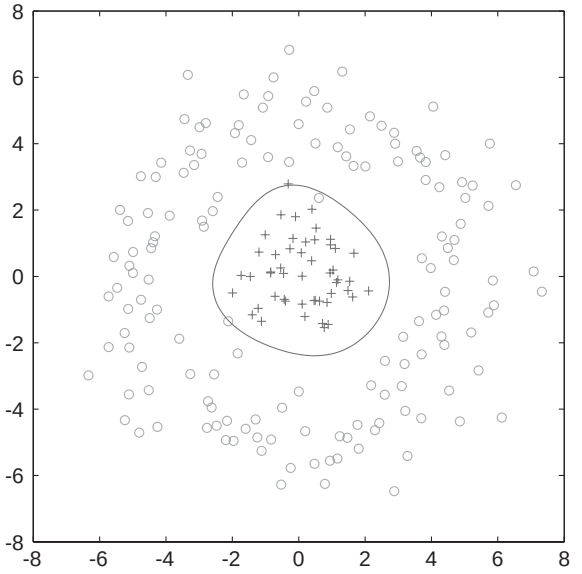

Fig. 1. Decision boundary formed by kernel Fisher discriminant analysis for a synthetic data set.

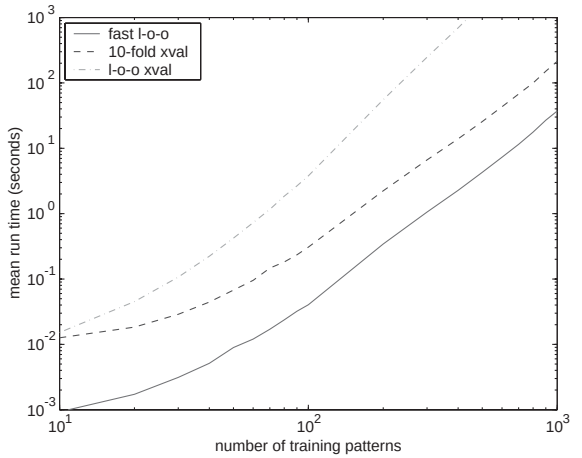

Fig. 2. Graph of run-time as a function of the number of training patterns for leave-one-out cross-validation of kernel Fisher discriminant classifiers via direct and fast approximate methods (mean of 20 trials).

data sets, the run-time for the fast leave-one-out estimator is also approximately seven times faster than 10-fold cross-validation. Inspection of the gradients of the curves displayed on The log–log axes show that the computational complexity of k-fold and the proposed leave-one-out estimator is, as expected, approximately $\mathcal{O}(\ell^3)$.

5.2. Generalisation

In order to verify that the improved efficiency of the leave-one-out cross-validation procedure is not obtained at the expense of generalisation, the proposed model selection procedure is evaluated on a suite of 13 real-world and synthetic benchmark problems from the UCI repository [31]. We adopt the experimental procedure used in the study by Rätsch et al. [32], where 100 different random training and test splits are defined (20 in the case of the large-scale image and splice datasets). Model selection is performed on the first five training splits, taking the median of the estimated values for the optimal regularisation (γ) and kernel (σ) parameters. Generalisation is then measured by the mean error rate over the 100 test splits (20 for image and splice datasets). The benchmarks, including test and training splits are available from http://ida.first.gmd.de/~raetsch/data/benchmarks.htm.

Model selection via minimisation of leave-one-out and 10-fold cross-validation estimates of the sum of squares error (10) are compared directly, to determine whether the higher variance of the leave-one-out estimator results in a consistent reduction in generalisation ability. The results obtained are also compared with those from Mika et al. [33], including kernel Fisher discriminant models where the model selection procedure minimised a 10-fold cross-validation estimate of the test error rate (9). This supports a comparison of continuous and discrete model selection criteria as well as a comparison with a range of other state-of-the art classification algorithms such as AdaBoost [34] and the support vector machine [16].

Table 1 shows the outcome of a comparison of model selection procedures for kernel Fisher discriminant models and a range of state-of-the-art statistical pattern recognition algorithms. The KFD with a leave-one-out model selection procedure (KFD-LOO) outperforms the KFD with 10-fold cross-validation (sum of squares) model selection (KFD-XVAL) on two of the 13 data sets (german and twonorm), demonstrates similar performance on nine, and performs worse on two (breast-cancer and splice). This clearly demonstrates that for a sum of squares selection criteria, the leave-one-out estimator does not significantly degrade performance, despite being known to exhibit a higher variance. The proposed leave-one-out model selection procedure outperforms the 10-fold cross-validation estimate of the test error rate adopted by Mika et al. (KFD) on seven of the 13 data sets (banana, diabetes, german, heart, ringnorm, titanic and waveform) and performs worse on the remaining six. This demonstrates that neither the continuous sum of squares or the discrete error rate statistics result in consistently superior generalisation. The leave-one-out model selection procedure should then be considered superior on the grounds of computational complexity. The superior performance of the leave-one-out KFD method, against the range of state-of-the-art algorithms, should also be noted, providing the lowest error

Table 1
Comparison of kernel Fisher discriminant with leave-one-out and k-fold model selection procedures using a simple least-squares criterion (LOO-KFD and XVAL-KFD, respectively), support vector machine (SVM) [2,8], kernel Fisher discriminant (KFD) [1], radial basis function (RBF) [35], AdaBoost (AB) [34] and regularised AdaBoost (AB_R) [32] classifiers on 13 different benchmark data sets [33]

Data set	LOO-KFD	XVAL-KFD	SVM	KFD	RBF	AB	AB_R
Banana	**10.4±0.04**	**10.4±0.04**	11.5±0.07	*10.8±0.05*	*10.8±0.06*	12.3±0.07	10.9±0.04
Breast cancer	26.3±0.42	26.1±0.43	*26.0±0.47*	**25.8±0.46**	27.6±0.47	30.4±0.47	26.5±0.45
Diabetes	**23.1±0.18**	**23.1±0.17**	23.5±0.17	*23.2±0.16*	24.3±0.19	26.5±0.23	23.8±0.18
German	**23.6±0.20**	*23.7±0.20*	**23.6±0.21**	*23.7±0.22*	24.7±0.24	27.5±0.25	24.3±0.21
Heart	**15.9±0.35**	**15.9±0.33**	*16.0±0.33*	16.1±0.34	17.6±0.33	20.3±0.34	16.5±0.35
Image	4.0±0.06	4.0±0.006	*3.0±0.06*	3.3±0.06	3.3±0.06	**2.7±0.07**	**2.7±0.06**
Ringnorm	**1.4±0.08**	**1.4±0.08**	1.7±0.01	*1.5±0.01*	1.7±0.02	1.9±0.03	1.6±0.01
Solar flare	34.2±1.63	34.2±1.66	**32.4±0.18**	*33.2±0.17*	34.4±0.2	35.7±0.18	34.2±0.22
Splice	10.8±0.07	10.7±0.06	10.9±0.07	10.5±0.06	*10.0±0.1*	10.1±0.05	**9.5±0.07**
Thyroid	4.5±0.20	4.5±0.21	4.8±0.22	**4.2±0.21**	4.5±0.21	*4.4±0.22*	4.6±0.22
Titanic	**22.3±0.12**	**22.3±0.09**	*22.4±0.1*	23.2±0.2	23.3±0.13	22.6±0.12	22.6±0.12
Twonorm	*2.7±0.02*	2.8±0.02	3.0±0.02	**2.6±0.02**	2.9±0.03	3.0±0.03	*2.7±0.02*
Waveform	**9.7±0.04**	**9.7±0.04**	9.9±0.04	9.9±0.04	10.7±0.11	10.8±0.06	*9.8±0.08*

The results for models SVM, KFD, RBF, AB and AB_R are taken from the study by Mika et al. [3,33]. The results for each method are presented in the form of the mean error rate over test data for 100 realisations of each data set (20 in the case of the image and splice data sets), along with the associated standard error. The best results are shown in bold-face and the second best in italics.

rate on seven of the 13 data sets and the second best on a further one.

6. Summary

In this paper we have presented a generalisation of an existing algorithm for leave-one-out cross-validation of multi-variate linear regression models (see e.g. Ref. [22]) to provide an estimate of the leave-one-out error of kernel Fisher discriminant classifiers. The proposed algorithm implements leave-one-out cross-validation of this class of kernel machine at a computational complexity of only $\mathcal{O}(\ell^3)$ operations, instead of the $\mathcal{O}(\ell^4)$ of a naïve approach. Furthermore, profiling information reveals that, providing $\boldsymbol{C}^{-1}$ is cached during training, the time taken to estimate the leave-one-out error rate is considerably *less* than the time taken to train the KFD classifier on the entire data set. As a result leave-one-out cross-validation becomes an attractive model selection criterion in large-scale applications of kernel Fisher discriminant analysis, being approximately seven times faster than conventional 10-fold cross-validation, while achieving a similar level of generalisation.

Acknowledgements

The authors thank Danilo Mandic, Tony Bagnall and Rob Foxall for their helpful comments on previous drafts of this manuscript and Ron Kohavi for interesting correspondence regarding leave-one-out cross-validation. This work was supported by a research grant from the Royal Society (grant number RSRG-22270).

Appendix. Derivation of closed-form expression for predicted residuals

From Eq. (6) we know that the vector of model parameters $p = (\boldsymbol{\alpha}, b)$ is given by

$$\boldsymbol{p} = (\boldsymbol{R} + \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z})^{-1}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{y},$$

where $\boldsymbol{Z} = [\boldsymbol{K}\ \boldsymbol{1}]$. For convenience, let $\boldsymbol{C} = \boldsymbol{R} + \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z}$ and $\boldsymbol{d} = \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{y}$, such that $\boldsymbol{p} = \boldsymbol{C}^{-1}\boldsymbol{d}$. Furthermore, let $\boldsymbol{Z}_{(i)}$ and $\boldsymbol{y}_{(i)}$ represent the data with the ith observation deleted, then

$$\boldsymbol{C}_{(i)} = \boldsymbol{C} - \boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}$$

and

$$\boldsymbol{d}_{(i)} = \boldsymbol{d} - y_i\boldsymbol{z}_i.$$

The Bartlett matrix inversion formula then gives

$$\boldsymbol{C}_{(i)}^{-1} = \boldsymbol{C} + \frac{\boldsymbol{C}^{-1}\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}}{1 - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{z}_i},$$

such that the vector of model parameters during the ith iteration of the leave-one-out cross-validation procedure becomes

$$\boldsymbol{p}_{(i)} = \left(\boldsymbol{C} + \frac{\boldsymbol{C}^{-1}\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}}{1 - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{z}_i}\right)(\boldsymbol{d} - y_i\boldsymbol{z}_i).$$

Let $\boldsymbol{H} = \boldsymbol{Z}\boldsymbol{C}^{-1}\boldsymbol{Z}^{\mathrm{T}}$ represent the *hat* matrix; note that the ith element of the leading diagonal can be written $h_{ii} = \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{z}_i$, so expanding the parentheses we have

$$\boldsymbol{p}_{(i)} = \boldsymbol{C}^{-1}\boldsymbol{d} - \boldsymbol{C}^{-1}y_i\boldsymbol{z}_i + \frac{\boldsymbol{C}^{-1}\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}}{1 - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{z}_i}$$

$$\boldsymbol{d} - \frac{\boldsymbol{C}^{-1}\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}}{1 - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{z}_i}y_i\boldsymbol{z}_i = \boldsymbol{p} + \left(\frac{\boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{p} - y_i}{1 - h_{ii}}\right)\boldsymbol{C}^{-1}\boldsymbol{z}_i.$$

The residual error for the ith training pattern is $r_i = y_i - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{p}$ and so

$$\boldsymbol{p}_{(i)} = \boldsymbol{p} - \frac{r_i}{1-h_{ii}}\boldsymbol{C}^{-1}\boldsymbol{z}_i.$$

Noting that $\boldsymbol{o} = \boldsymbol{Z}\boldsymbol{p}$, the output of the model during the ith iteration of the leave-one-out cross-validation procedure can be written as

$$\boldsymbol{o}_{(i)} = \boldsymbol{Z}\boldsymbol{p}_{(i)} = \boldsymbol{t} - \frac{r_i}{1-h_{ii}}\boldsymbol{h}_i,$$

where $\boldsymbol{h}_i$ is the ith column of $\boldsymbol{H}$. The vector of residuals for the training patterns during the leave-one-out cross-validation procedure can then be written in closed form as

$$\boldsymbol{r}_{(i)} = \boldsymbol{y} - \boldsymbol{o}_{(i)} = \boldsymbol{r} + r_i\frac{1}{1-h_{ii}}\boldsymbol{h}_i.$$

The ith element of $\boldsymbol{r}_{(i)}$ is given by

$$\{\boldsymbol{r}_{(i)}\}_i = \frac{r_i}{1-h_{ii}}.$$

References

[1] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, K.-R. Müller, Fisher discriminant analysis with kernels, in: Y.H. Hu, J. Larsen, E. Wilson, S. Douglas (Eds.), Neural Networks for Signal Processing, Vol. IX, IEEE Press, New York, 1999, pp. 41–48.

[2] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines (and other kernel-based learning methods), Cambridge University Press, Cambridge, UK, 2000.

[3] B. Schölkopf, A.J. Smola, Learning with Kernels—Support Vector Machines, Regularization, Optimization and Beyond, MIT Press, Cambridge, MA, 2002.

[4] C. Saunders, A. Gammermann, V. Vovk, Ridge regression in dual variables, in: J. Shavlik (Ed.), Proceedings of the 15th International Conference on Machine Learning (ICML-1998), Morgan Kaufmann, Los Altos, CA, 1998.

[5] B. Schölkopf, A.J. Smola, K. Müller, Kernel principal component analysis, in: W. Gerstner, A. Germond, M. Hasler, J.-D. Nicoud (Eds.), Proceedings of the International Conference on Artificial Neural Networks (ICANN-1997), Lecture Notes in Computer Science (LNCS), Vol. 1327, Springer, Berlin, 1997, pp. 583–588.

[6] P.L. Lai, C. Fyfe, Kernel and nonlinear canonical correlation analysis, Int. J. Neural Systems 10 (5) (2000) 365–377.

[7] B.E. Boser, I.M. Guyon, V. Vapnik, A training algorithm for optimal margin classifiers, in: D. Haussler (Ed.), Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory, Pittsburgh, PA, July 1992, pp. 144–152.

[8] C. Cortes, V. Vapnik, Support vector networks, Mach. Learning 20 (1995) 273–297.

[9] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugenics 7 (1936) 179–188.

[10] J. Mercer, Functions of positive and negative type and their connection with the theory of integral equations, Philos. Trans. R. Soc. London, A 209 (1909) 415–446.

[11] A.N. Tikhonov, V.Y. Arsenin, Solutions of Ill-posed Problems, Wiley, New York, 1977.

[12] M. Stone, Cross-validatory choice and assessment of statistical predictions, J. R. Stat. Soc. B 36 (1) (1974) 111–147.

[13] V. Vapnik, O. Chapelle, Bounds on error expectation for SVM, in: A.J. Smola, P.L. Bartlett, B. Schölkopf, D. Schuurmans (Eds.), Advances in Large Margin Classifiers, MIT Press, Cambridge, Massachusetts, USA, 2000, pp. 261–280 (Chapter 14).

[14] C. Chapelle, V. Vapnik, O. Bousquet, S. Mukherjee, Choosing multiple parameters for support vector machines, Mach. Learning 46 (1) (2002) 131–159.

[15] A. Luntz, V. Brailovsky, On estimation of characters obtained in statistical procedure of recognition, Techicheskaya Kibernetica 3 (1969) (in Russian).

[16] V. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.

[17] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: Proceedings of the 14th International Conference on Artificial Intelligence (IJCAI), San Mateo, CA, Morgan Kaufmann, Los Altos, CA, 1995, pp. 1137–1143.

[18] N. Aronszajn, Theory of reproducing kernels, Trans. Amer. Math. Soc. 68 (1950) 337–404.

[19] G.S. Kimeldorf, G. Wahba, Some results on Tchebycheffian spline functions, J. Math. Anal. Appl. 33 (1971) 82–95.

[20] J. Xu, X. Zhang, Y. Li, Kernel MSE algorithm: a unified framework for KFD, LS-SVM and KRR, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN-2001), Washington, DC, July 2001, pp. 1486–1491.

[21] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Process. Lett. 9 (3) (1990) 293–300.

[22] S. Weisberg, Applied Linear Regression, 2nd Edition, Wiley, New York, 1985.

[23] J. Sherman, W.J. Morrison, Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix, Ann. Math. Stat. 20 (4) (1949) 621.

[24] J. Sherman, W.J. Morrison, Adjustment of an inverse matrix corresponding to a change in one element of a given matrix, Ann. Math. Stat. 21 (1) (1950) 124–127.

[25] M. Woodbury, Inverting modified matrices, Memorandum Report 42, Princeton University, Princeton, USA, 1950.

[26] M.S. Bartlett, An inverse matrix adjustment arising in discriminant analysis, Ann. Math. Stat. 22 (1) (1951) 107–111.

[27] W.W. Hager, Updating the inverse of a matrix, SIAM Rev. 31 (2) (1989) 221–239.

[28] G.H. Golub, C.F. Van Loan, Matrix Computations, 3rd Edition, The Johns Hopkins University Press, Baltimore, 1996.

[29] D.M. Allen, The relationship between variable selection and prediction, Technometrics 16 (1974) 125–127.

[30] R.D. Cook, S. Weisberg, Residuals and Influence in Regression, Monographs on Statistics and Applied Probability, Chapman & Hall, New York, 1982.

[31] S.D. Bay, The UCI KDD archive, University of California, Department of Information and Computer Science, Irvine, CA, 1999 (http://kdd.ics.uci.edu/).

[32] G. Rätsch, T. Onoda, K.-R. Müller, Soft margins for AdaBoost, Mach. Learning 42 (3) (2001) 287–320.

[33] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, A.J. Smola, K.-R. Müller, Invariant feature extraction and classification in feature spaces, in: S.A. Solla, T.K. Leen, K.-R. Müller (Eds.),

Advances in Neural Information Processing Systems, Vol. 12, MIT Press, Cambridge, MIT, 2000, pp. 526–532.
[34] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: Proceedings of the 13th International Conference on Machine Learning, Morgan Kaufmann, Los Altos, CA, pp. 148–156.
[35] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Oxford, 1995.

About the Author—GAVIN CAWLEY (AMIEE, MIEEE) received a B.Eng and Ph.D. in Electronic Systems Engineering from the University of Essex in 1990 and 1996, respectively. He is currently a lecturer in the School of Information System at the University of East Anglia. His research interests include machine learning and signal processing.

About the Author—NICOLA TALBOT (CMath MIMA) received her B.Sc. in mathematics and Ph.D. in Electronic Systems Engineering from the University of Essex in 1991 and 1996, respectively. She formerly worked at the Institute of Food Research, funded by the Ministry of Agriculture Fisheries and Food. Her research interests include optimisation and Bayesian belief networks.