Efficient Model Selection for Kernel Logistic Regression

Gavin C. Cawley and Nicola L. C. Talbot
School of Computing Sciences
University of East Anglia
Norwich, United Kingdom
{gcc,nlct}@cmp.uea.ac.uk

Abstract

Kernel logistic regression models, like their linear counterparts, can be trained using the efficient iteratively re-weighted least-squares (IRWLS) algorithm. This approach suggests an approximate *leave-one-out cross-validation estimator based on an existing method for exact leave-one-out cross-validation of least-squares models. Results compiled over seven benchmark datasets are presented for kernel logistic regression with model selection procedures based on both conventional k-fold and approximate leave-one-out cross-validation criteria.*

1. Introduction

Kernel logistic regression provides a useful addition to the family of kernel learning methods for pattern recognition applications where the misclassification costs are not known *a-priori*, and so estimates of *a-posteriori* probability are more useful than simple binary classifications. The optimal values for the parameters of a kernel logistic regression model are given by the solution of a convex optimisation problem, which can be solved efficiently via the iteratively re-weighted least-squares algorithm. However, in order to achieve optimal generalisation performance, good values for the kernel and regularisation parameters must also be found, a process known as "model selection". This process is normally performed via iterative improvement of some model selection criterion, expected to be strongly correlated with performance on unseen data, for instance methods based on cross-validation [15]. In this paper we propose an efficient approximation of the leave-one-out cross-validation procedure for used as a model selection criterion for kernel logistic regression models that can be computed as a bye-product of the training procedure. This estimator is shown to be comparable with conventional 10-fold cross-validation in terms of final model performance, but at substantially reduced computational expense.

1.1. Kernel Logistic Regression

A non-linear form of logistic regression, known as kernel logistic regression (KLR), can be obtained via the so-called "kernel trick", whereby a conventional logistic regression model is constructed in a high-dimensional feature space, induced by a Mercer kernel. More formally, given labelled training data,

$$\mathcal{D} = \{(\boldsymbol{x}_i, t_i)\}_{i=1}^{\ell}, \quad \boldsymbol{x}_i \in \mathcal{X} \in \mathbb{R}^d, \quad t_i \in [0,\ 1],$$

a feature space, $\mathcal{F}$ ($\boldsymbol{\phi} : \mathcal{X} \to \mathcal{F}$), is defined by a kernel function, $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, that evaluates the inner product between the images of input vectors in the feature space, i.e. $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x}) \cdot \boldsymbol{\phi}(\boldsymbol{x}')$ (see e.g. [14]). The kernel function used here is the isotropic radial basis function (RBF),

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

A conventional logistic regression model is then constructed in the feature space, such that

$$\operatorname{logit}\{y(\boldsymbol{x})\} = \boldsymbol{w} \cdot \boldsymbol{\phi}(\boldsymbol{x}) + b, \quad \operatorname{logit}(p) = \log \frac{p}{1-p}.$$

The optimal model parameters $(\boldsymbol{w},\ b)$ are found by minimising a cost function representing the regularised [16] negative-log likelihood of the data,

$$L = \lambda \|\boldsymbol{w}\|^2 - \sum_{i=1}^{\ell} t_i \log \mu_i + (1 - t_i) \log(1 - \mu_i), \quad (1)$$

where λ is a regularisation parameter controlling the bias-variance trade-off [6]. The representer theorem [9] states that the solution to an optimisation problem of this nature can be written in the form of a linear combination of the training patterns, such that $\operatorname{logit}\{y(\boldsymbol{x})\}$ is given by the familiar kernel expansion,

$$\boldsymbol{w} = \sum_{i=1}^{\ell} \alpha_i \boldsymbol{\phi}(\boldsymbol{x}_i), \implies \operatorname{logit}\{y(\boldsymbol{x})\} = \sum_{i=1}^{\ell} \alpha_i \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}) + b.$$

Furthermore, it is straight-forward to show that L (1) represents a convex optimisation problem, and so there is only a single, global minima. The optimal model parameters $(\boldsymbol{\alpha}, b)$ can be found using Newton's method or equivalently an iteratively re-weighted least-squares procedure, e.g. [10].

1.2. Iteratively Re-weighted Least-Squares

Conventional logistic regression models are typically fitted using the well-known iteratively re-weighted least squares (IRWLS) algorithm [10]. With a slight modification to accommodate the regularisation term, kernel logistic regression models can also be trained using IRWLS. The coefficients of the kernel expansion in each iteration are given by the solution of a weighted least squares problem,

$$\boldsymbol{\alpha} = \left(\boldsymbol{\Phi}^T \boldsymbol{W} \boldsymbol{\Phi} + \boldsymbol{R}\right)^{-1} \boldsymbol{\Phi}^T \boldsymbol{W} \boldsymbol{\eta}. \tag{2}$$

The bias, b, is implemented by introducing an additional basis function with a constant value, such that $\boldsymbol{\Phi} = [\boldsymbol{K}\ \mathbf{1}]$, where $\boldsymbol{K} = [k_{ij} = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j=1}^{\ell}$. The effect of the regularisation term is represented by the matrix $\boldsymbol{R}$,

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{K} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix},$$

note the bias term is not regularised. The *weight* matrix, $\boldsymbol{W}$, and "target" vector $\boldsymbol{\eta}$ are then updated for the subsequent iteration, such that

$$\boldsymbol{W} = \mathrm{diag}(\{w_1, w_2, \ldots, w_\ell\}), \quad w_i = \mu_i(1-\mu_i), \tag{3}$$

and

$$\eta_i = z_i + \frac{t_i - \mu_i}{\mu_i(1-\mu_i)}, \tag{4}$$

where $z_i = \mathrm{logit}\{y(\boldsymbol{x}_i)\}$. Steps 2-4 are repeated until convergence. In practise any column of $\boldsymbol{\Phi}$ that is, at least *numerically*, linearly dependent on the remaining columns can be deleted from the model. In this study, we identify a subset of linearly independent columns using the incomplete Cholesky factorisation algorithm [5].

2. Efficient Model Selection

Model selection, the process of determining the optimal regularisation and kernel parameters, is an important issue in fitting kernel models. In this section, we discuss efficient implementation of the k-fold cross-validation procedure and present a novel approximation to the leave-one-out estimator.

2.1. Efficient k-Fold Cross-Validation

A useful feature of the iteratively re-weighted least-squares training algorithm is that only the *output* of an existing model is required to provide a good initial values for $\boldsymbol{W}$ and $\boldsymbol{\eta}$ for use in the first iteration. In implementing the k-fold cross-validation procedure, a useful reduction in computational expense can be obtained through using the output of the model trained in the first fold to "seed" the training of models in subsequent folds. If cross-validation is being used as the optimisation criterion for an iterative model selection procedure, a model trained in a previous iteration can be used as a seed, and so only a single model needs to be trained from the beginning. Note that unlike the "alpha-seeding" approach used in support vector machines [4], the seed and current models need not be trained on the same data and may have completely different kernel functions as only the *output* of the seed model is used.

2.2. Approximate Leave-One-Out Cross-Validation

Efficient methods for leave-one-out cross-validation of linear least-squares regression models have been available for some time [1, 3, 17], and have more recently been applied to kernel learning methods minimising a regularised sum-of-squares loss function [2]. These methods can also be applied to kernel logistic regression models, if we assume that $\boldsymbol{W}$ and $\boldsymbol{\eta}$ are approximately unchanged by the deletion of a single training pattern during each iteration of the leave-one-out procedure. The "hat" matrix, $\boldsymbol{H}$, for a regularised weighted least-squares regression problem is given by

$$\boldsymbol{H} = [h_{ij}]_{i,j=1}^{\ell} = \boldsymbol{\Phi}\left(\boldsymbol{\Phi}^T \boldsymbol{W} \boldsymbol{\Phi} + \boldsymbol{R}\right)^{-1} \boldsymbol{\Phi}^T \boldsymbol{W}. \tag{5}$$

Using the Sherman-Woodbury-Morrison formula [8, 7] for a general rank-1 update of the inverse of a matrix,

$$\left[\boldsymbol{C} + \boldsymbol{u}\boldsymbol{v}^T\right]^{-1} = \boldsymbol{C}^{-1} + \frac{\boldsymbol{C}^{-1}\boldsymbol{u}\boldsymbol{v}^T\boldsymbol{C}^{-1}}{1 - \boldsymbol{v}^T\boldsymbol{C}^{-1}\boldsymbol{u}},$$

it is straight-forward to show that

$$\left\{\boldsymbol{z}_{(i)}\right\}_i = z_i - \frac{h_{ii}\left(\eta_i - z_i^{\mu}\right)}{1 - h_{ii}}, \tag{6}$$

where $\left\{\boldsymbol{z}_{(i)}\right\}_j$ represents the j^{th} element of $\boldsymbol{z}$ during the i^{th} iteration of the leave-one-out cross-validation procedure, from which the leave-one-out cross-validation estimate of the cross-entropy can be obtained. For a more detailed derivation, see [2].

3. Results

Figure 1 shows a graph of the time taken for model selection procedures based on the minimisation of three different model selection criteria, using the Nelder-Mead simplex algorithm [11], for a synthetically generated dataset

[2]. The approximate leave-one-out cross-validation estimator (ALOO) is significantly less expensive than either 10-fold cross-validation (XVAL) or the conventional leave-one-out estimator (LOO). Note also that the scaling properties of the approximate leave-one-out method are similar to those of the 10-fold cross-validation estimator than those of conventional leave-one-out cross-validation, which is generally accepted as being prohibitively expensive for all but the smallest datasets.

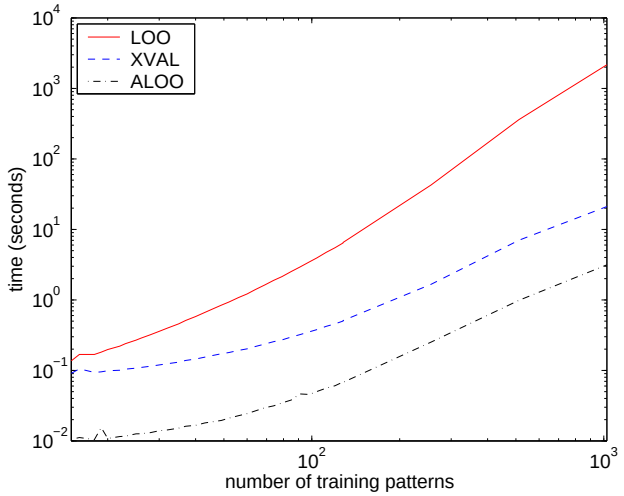


Figure 1. Time taken for model selection using three different model selection criteria.

Figure 2 shows contour plots of the test set, 10-fold cross-validation and approximate leave-one-out estimates of the cross-entropy loss function for the Pima benchmark dataset [13] as a function of the hyper-parameters γ and λ. The general form of all three plots are very similar, indicating that the proposed approximate leave-one-out procedure provides a good estimator of test set error. While the optimal hyper-parameters given by the 10-fold cross-validation and approximate leave-one-out estimators are somewhat different from those optimising the test set error, they all lie along the bottom of a shallow trough in the test set loss and so give similar performance.

The proposed approximate leave-one-out estimator was then used as a criterion for model selection based on a simple Nelder-Mead simplex optimisation algorithm [11]. For comparison, 10-fold and split-sample model selection criteria were also investigated. The split-sample estimator uses the cross-entropy measured over the test set as the selection criterion and so provides an indication of the best achievable performance on the test set. Table 1 shows the cross-entropy measured over the test set for the three model selection criteria for the Pima and Synthetic benchmarks used

Benchmark	TEST	XVAL	ALOO
Breast Cancer	39.46	40.96	40.95
Diabetis	142.98	143.19	143.09
Heart	44.84	47.54	48.30
Pima	146.20	146.56	146.85
Synthetic	228.65	230.99	254.15
Thyroid	2.44	2.83	2.93
Titanic	1029.42	1044.87	1070.47

Table 1. Minimal test set loss according to model selection criteria for a range of benchmark datasets.

Benchmark	XVAL	ALOO
Breast Cancer	203.14	24.34
Diabetis	1902.63	360.30
Heart	148.64	22.15
Pima	228.07	22.99
Synthetic	230.99	14.78
Thyroid	117.06	18.42
Titanic	28.43	2.07

Table 2. Model selection time by model selection criteria for a range of benchmark datasets.

by Ripley [13] and the Diabetis, Heart, Thyroid and Titanic benchmarks used by Rätsch *et al.* [12]. The performances of the approximate leave-one-out and 10-fold cross-validation criteria are generally quite similar, except for the synthetic and titanic benchmarks, where 10-fold cross-validation is clearly superior, although the difference is still relatively small. Table 2 shows the corresponding model selection time for each of these experiments, revealing that the approximate leave-one-out estimator is around an order of magnitude faster.

4. Conclusions

In this paper, we have introduced a novel approximate leave-one-out cross-validation procedure for kernel logistic regression models, based on the interpretation of the training procedure as a weighted least-squares problem. This was shown to provide a highly efficient criterion for automated model selection, achieving generalisation performance comparable to that obtained using conventional 10-fold cross-validation (with seeding) at a significantly lower computational expense.

References

[1] D. M. Allen. The relationship between variable selection and prediction. *Technometrics*, 16:125–127, 1974.

[2] Cawley, G. C. and Talbot, N. L. C. Efficient leave-one-out cross-validation of kernel Fisher discriminant classifiers. *Pattern Recognition*, 36(11):2585–2592, Nov. 2003.

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

[4] D. DeCoste and K. Wagstaff. Alpha seeding for support vector machines. In *Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 345–349, Boston, MA, 2000.

[5] S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, Dec. 2001.

[6] S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58, 1992.

[7] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, third edition edition, 1996.

[8] W. W. Hager. Updating the inverse of a matrix. *SIAM Review*, 31(2):221–239, June 1989.

[9] G. S. Kimeldorf and G. Wahba. Some results on Tchebycheffian spline functions. *J. Math. Anal. Applic.*, 33:82–95, 1971.

[10] I. T. Nabney. Efficient training of RBF networks for classification. In *Proceedings of the Ninth International Conference on Artificial Neural Networks*, volume 1, pages 210–215, Sept. 7–10 1999.

[11] J. A. Nelder and R. Mead. A simplex method for function minimisation. *Computer Journal*, 7:308–313, 1965.

[12] G. Rätsch, T. Onoda, and K.-R. Müller. Soft margins for AdaBoost. *Machine Learning*, 42(3):287–320, 2001.

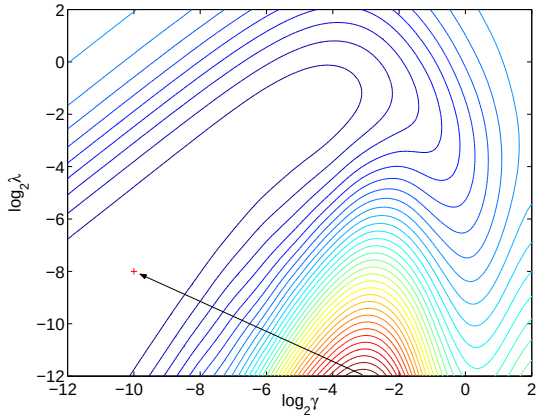
[13] B. D. Ripley. *Pattern recognition and neural networks*. Cambridge University Press, 1996.

[14] B. Schölkopf and A. J. Smola. *Learning with kernels - support vector machines, regularization, optimization and beyond*. MIT Press, Cambridge, MA, 2002.

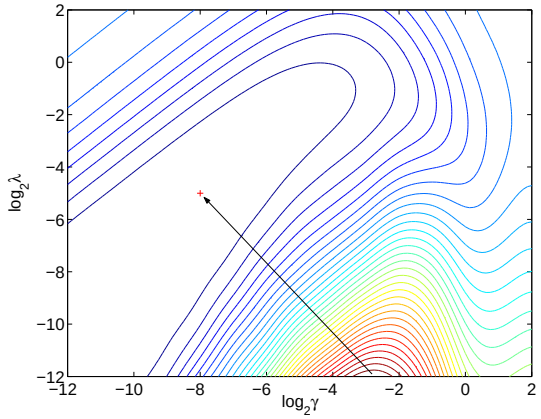
[15] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, B*, 36(1):111–147, 1974.

[16] A. N. Tikhonov and V. Y. Arsenin. *Solutions of ill-posed problems*. John Wiley, New York, 1977.

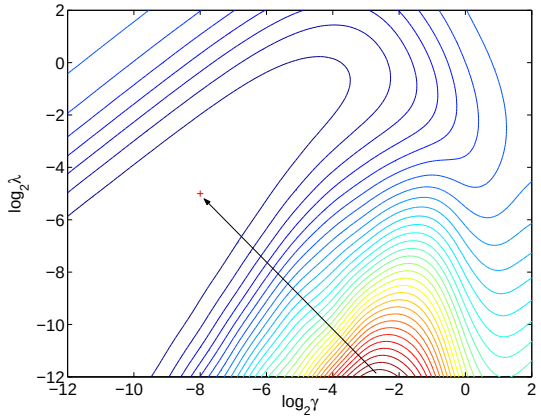
[17] S. Weisberg. *Applied linear regression*. John Wiley and Sons, New York, second edition, 1985.



(a)



(b)



(c)

Figure 2. Plot of (a) test set (b) 10-fold cross-validation and (c) approximate leave-one-out estimates of cross-entropy as a function of the hyper-parameters γ and λ for the Pima dataset. The arrow indicates the direction towards the point of lowest cross-entropy (marked by a cross).