BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 19 2006, pages 2348–2355
doi:10.1093/bioinformatics/btl386

Gene expression

Gene selection in cancer classification using sparse logistic regression with Bayesian regularization

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

School of Computing Sciences, University of East Anglia, Norwich NR4 7TJ, UK

Received on March 29, 2006; revised on June 22, 2006; accepted on July 6, 2006
Advance Access publication July 14, 2006
Associate Editor: David Rocke

ABSTRACT

Motivation: Gene selection algorithms for cancer classification, based on the expression of a small number of biomarker genes, have been the subject of considerable research in recent years. Shevade and Keerthi propose a gene selection algorithm based on sparse logistic regression (SLogReg) incorporating a Laplace prior to promote sparsity in the model parameters, and provide a simple but efficient training procedure. The degree of sparsity obtained is determined by the value of a regularization parameter, which must be carefully tuned in order to optimize performance. This normally involves a model selection stage, based on a computationally intensive search for the minimizer of the cross-validation error. In this paper, we demonstrate that a simple Bayesian approach can be taken to eliminate this regularization parameter entirely, by integrating it out analytically using an uninformative Jeffrey's prior. The improved algorithm (BLogReg) is then typically two or three orders of magnitude faster than the original algorithm, as there is no longer a need for a model selection step. The BLogReg algorithm is also free from selection bias in performance estimation, a common pitfall in the application of machine learning algorithms in cancer classification.

Results: The SLogReg, BLogReg and Relevance Vector Machine (RVM) gene selection algorithms are evaluated over the well-studied colon cancer and leukaemia benchmark datasets. The leave-one-out estimates of the probability of test error and cross-entropy of the BLogReg and SLogReg algorithms are very similar, however the BlogReg algorithm is found to be considerably faster than the original SLogReg algorithm. Using nested cross-validation to avoid selection bias, performance estimation for SLogReg on the leukaemia dataset takes almost 48 h, whereas the corresponding result for BLogReg is obtained in only 1 min 24 s, making BLogReg by far the more practical algorithm. BLogReg also demonstrates better estimates of conditional probability than the RVM, which are of great importance in medical applications, with similar computational expense.

Availability: A MATLAB implementation of the sparse logistic regression algorithm with Bayesian regularization (BLogReg) is available from http://theoval.cmp.uea.ac.uk/~gcc/cbl/blogreg/

Contact: gcc@cmp.uea.ac.uk

1 INTRODUCTION

Cancer classification based on microarray gene-expression data, ideally identifying a small number of discriminatory biomarker genes, provides one of the earliest applications of machine learning methods in computational biology. A wide variety of machine learning algorithms have been applied to this problem, including the support vector machine (Guyon *et al*., 2002), sparse logistic regression (SLogReg) (Shevade and Keerthi, 2003), the relevance vector machine (RVM) (Li *et al*., 2002), Gaussian Process models (Chu *et al*., 2005) and simple decision rules (Tan *et al*., 2005). The common aims of such algorithms are 2-fold: primarily to distinguish between patients suffering from subtly different forms of cancer, with the highest possible degree of accuracy, on the basis of their gene expression profiles obtained by broad-spectrum microarray analysis. The second goal is to identify a small sub-set of biomarker genes, for which expression patterns are highly indicative of a particular form of cancer, and are therefore implicated by association. This second goal is concerned with improving our understanding of the underlying causes of the cancer.

In this paper, we propose a substantial improvement to the sparse logistic regression (SLogReg) approach of Shevade and Keerthi (2003). The SLogReg algorithm employs an $L1$-norm regularization term (Tikhonov and Arsenin, 1977), corresponding to a Laplace prior over the model parameters (c.f Williams, 1995), in order to identify a sparse sub-set of the most discriminatory features corresponding to biomarker genes. Both the generalization ability of the classifier and the level of sparsity achieved are critically dependent on the value of a regularization parameter, which must be carefully tuned to optimize performance. This is normally achieved by a computationally intensive search for the minimizer of a cross-validation based estimate of generalization performance. Instead, we adopt a Bayesian approach, in which the regularization parameter is integrated out analytically, using an uninformative Jeffery's prior, in the style of Buntine and Weigend (1991) (see also Lehrach *et al*., 2006). The resulting parameterless classification algorithm (BLogReg) is very much easier to use, is comparable in performance with the original sparse logistic regression algorithms, but is two or three orders of magnitude faster, as there is no longer a need for a model selection stage to optimize the regularization parameter.

The remainder of this paper is structured as follows: The existing sparse logistic regression (SLogReg) algorithm (Shevade and Keerthi, 2003) is reviewed in Section 2. The modified Bayesian logistic regression (BLogReg) algorithm is then introduced in Section 3. Experimental results obtained on the well-studied colon cancer (Alon *et al*., 1999) and leukaemia (Golub *et al*., 1999) benchmark problems are presented in Section 4, demonstrating the competitiveness of the improved algorithm. Finally, the work is summarized and conclusions drawn in Section 5.

*To whom correspondence should be addressed.

2 SPARSE LOGISTIC REGRESSION

We are commonly faced with statistical pattern recognition problems, where we must learn some decision rule distinguishing between objects belonging to one of two classes, based on a set of ℓ training examples,

$$\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{\ell}, \quad \boldsymbol{x}_i \in \mathcal{X} \subset \mathbb{R}^d, \quad y_i \in \{-1, +1\},$$

where $\boldsymbol{x}_i$ represents a vector of measurements describing the i-th example, and y_i indicates the class to which the i-th example belongs, with $y_i = +1$ representing class $\mathcal{C}_1$ and $y_i = -1$ representing class $\mathcal{C}_2$. Logistic regression is a classical approach to this problem, that attempts to estimate the *a-posteriori* probability of class membership based on a linear combination of the input features,

$$p(\mathcal{C}_1 \mid \boldsymbol{x}) = \frac{1}{1 + \exp\{-f(\boldsymbol{x})\}}, \tag{1}$$

where

$$f(\boldsymbol{x}_i) = \sum_{j=1}^{d} \alpha_j x_{ij} + \alpha_0. \tag{2}$$

The parameters of the logistic regression model, $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_d)$, can be found by maximizing the likelihood of the training examples, or equivalently by minimizing the negative log-likelihood. Assuming $\mathcal{D}$ represents an independent and identically distributed (i.i.d.) sample from a Bernoulli distribution, the negative log-likelihood is given by

$$E_{\mathcal{D}} = \sum_{i=1}^{\ell} g\{-y_i f(\boldsymbol{x}_i)\},$$

where

$$g\{\xi\} = \log\{1 + \exp(\xi)\}.$$

Minimizing the negative log-likelihood is relatively straightforward as the first and second derivatives, with respect to individual model parameters, are continuous and easily computed,

$$\frac{\partial E_{\mathcal{D}}}{\partial \alpha_j} = -\sum_{i=1}^{\ell} \frac{\exp\{-y_i f(\boldsymbol{x}_i)\} y_i x_{ij}}{1 + \exp\{-yf(\boldsymbol{x}_i)\}} \tag{3}$$

and

$$\frac{\partial^2 E_{\mathcal{D}}}{\partial \alpha_j^2} = \sum_{i=1}^{\ell} \frac{\exp\{-y_i f(\boldsymbol{x}_i)\} y_i^2 x_{ij}^2}{[1 + \exp\{-yf(\boldsymbol{x}_i)\}]^2}. \tag{4}$$

The resulting model is however fully dense, in the sense that none of the model parameters $\boldsymbol{\alpha}$ are in general exactly zero. Ideally we would prefer a model based on a small selection of the most informative features, with the remaining features being 'pruned' from the model. A sparse model can be introduced by adding a regularization term to the negative log-likelihood (e.g. Williams, 1995), corresponding to a Laplace prior over α, to give a modified training criterion,

$$M = E_{\mathcal{D}} + \lambda E_{\alpha}, \quad \text{where} \quad E_{\alpha} = \sum_{i=1}^{d} |\alpha_i| \tag{5}$$

and λ is a regularization parameter, controlling the bias-variance trade-off and simultaneously the sparsity of the resulting model. Note that the usual bias parameter α_0 is normally left unregularized.

Table 1. Special cases that must be considered in optimizing M with respect to α_i in order to avoid difficulties due to the discontinuity in the first derivative at the origin

Case	α_{i}	$\frac{\partial M}{\partial \alpha_i}\Big\vert_{\alpha_i}$	$\frac{\partial M}{\partial \alpha_i}\Big\vert_{0^-}$	$\frac{\partial M}{\partial \alpha_i}\Big\vert_{0^+}$	L	H
1	0	—	<0	<0	0	$+\infty$
2	0	—	>0	>0	0	$-\infty$
3	<0	>0	—	—	$-\infty$	α_{i}
4	>0	<0	—	—	α_{i}	$+\infty$
5	<0	<0	>0	—	α_{i}	0
6	>0	>0	—	<0	0	α_{i}
7	<0	<0	—	>0	0	$+\infty$
8	>0	>0	<0	—	$-\infty$	0
9	<0	<0	≤ 0	≥ 0	0	0
10	>0	>0	≤ 0	≥ 0	0	0

At a minima of M, the partial derivatives of M with respect to the model parameters will be uniformly zero, giving

$$\left|\frac{\partial E_{\mathcal{D}}}{\partial \alpha_i}\right| = \lambda \quad \text{if} \quad |\alpha_i| > 0 \quad \text{and} \quad \left|\frac{\partial E_{\mathcal{D}}}{\partial \alpha_i}\right| < \lambda \quad \text{if} \quad |\alpha_i| = 0.$$

This implies that if the sensitivity of the negative log-likelihood with respect to a model parameter, α_i, falls below λ, then the value of that parameter will be set exactly to zero and the corresponding input feature can be pruned from the model. The principal shortcomings of this approach lie in the training algorithm no longer involving an optimization problem with continuous derivatives and in the need for lengthy cross-validation trials to determine a good value for the regularization parameter λ. Shevade and Keerthi (2003) provide a solution to the first problem, described in the remainder of this section. This paper proposes a Bayesian solution to the second problem, where the regularization parameter is integrated out analytically.

2.1 An efficient optimization procedure

The training algorithm proposed by Shevade and Keerthi (2003) seeks to minimize the cost function (5) by optimizing one parameter at a time via Newton's method. However, owing to the discontinuity in the first derivative at the origin, care must be exercised when the value of a model parameter passes through zero. This is achieved by bracketing the optimal value for a model parameter, α_i, by upper and lower limits (H and L respectively) such that the interval does not include 0, except perhaps at a boundary. These limits can be computed using the gradient of M with respect to α_i computed at its current value and at zero, from both above and below, as shown in Table 1 and illustrated by Figure 1.

A model parameter must be selected for optimization at each iteration, the parameter with the gradient of the greatest magnitude is a sensible choice. In order to improve the speed of convergence, we begin by optimizing only active parameters (those with non-zero values), and only consider inactive parameters if no active parameter can be found with a non-zero gradient. Iterative optimization procedures do not generally reduce the gradient exactly to zero, and so in practice we only consider parameters for optimization if they have a gradient exceeding a pre-defined tolerance parameter τ. The algorithm terminates when no such parameter can be found.

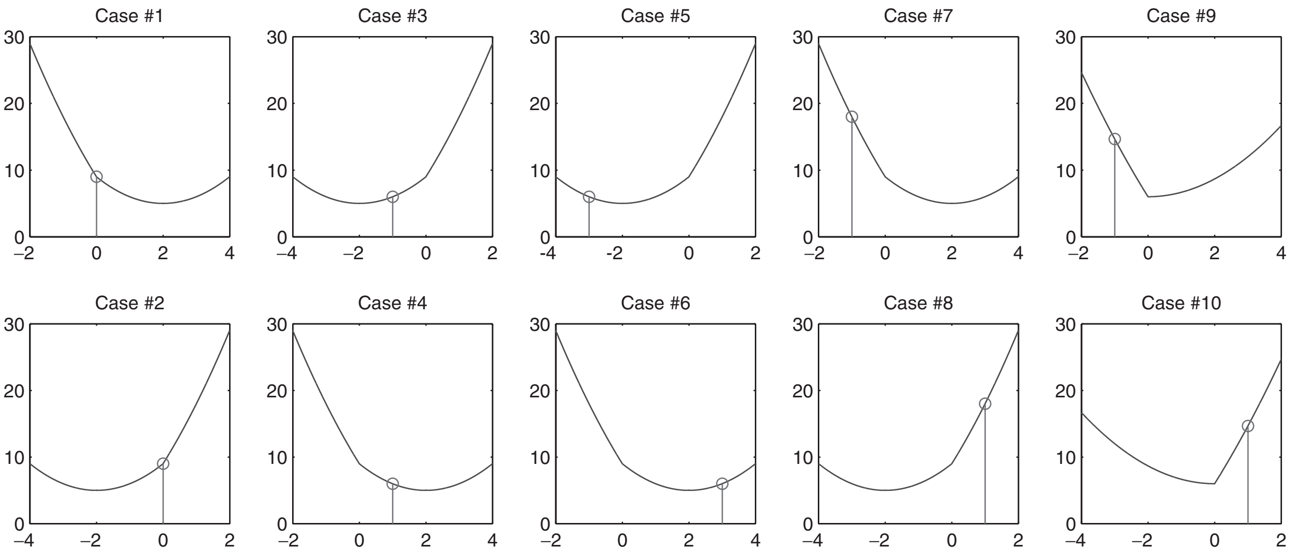

Fig. 1. Graphical depiction of the special cases, listed in full in Table 1, to be considered in optimizing a model parameter in order to deal with the discontinuity in the first derivative at the origin. Note that even and odd numbered cases differ only by reflection.

For a complete description of the training algorithm, see Shevade and Keerthi (2003).

3 BAYESIAN REGULARIZATION

In this section, we demonstrate how the regularization parameter may be eliminated, following the methods of Buntine and Weigend (1991) and Williams (1995), before going on to describe the modification of the training procedure of Shevade and Keerthi (2003) required to accommodate the revised optimization problem.

3.1 Eliminating the regularization parameter λ

Minimization of (5) has a straight-forward Bayesian interpretation; the posterior distribution for $\boldsymbol{\alpha}$, the parameters of the model given by (1 and 2), can be written as

$$p(\boldsymbol{\alpha}|\mathcal{D},\lambda) \propto p(\mathcal{D}|\boldsymbol{\alpha})p(\boldsymbol{\alpha}|\lambda).$$

M is then, up to an additive constant, the negative logarithm of the posterior density. The prior over model parameters, $\boldsymbol{\alpha}$, is then given by a separable Laplace distribution

$$p(\boldsymbol{\alpha}|\lambda) = \left(\frac{\lambda}{2}\right)^N \exp\{-\lambda E_\alpha\} = \prod_{i=1}^{N} \frac{\lambda}{2}\exp\{-\lambda|\alpha_i|\}, \qquad (6)$$

where N is the number of active (non-zero) model parameters. A good value for the regularization parameters λ can be estimated, within a Bayesian framework, by maximizing the evidence (MacKay, 1992a, b, c) or alternatively it may be integrated out analytically (Buntine and Weigend, 1991; Williams, 1995). Here we take the latter approach, where the prior distribution over model parameters is given by marginalizing over λ,

$$p(\boldsymbol{\alpha}) = \int p(\boldsymbol{\alpha}|\lambda)p(\lambda)\,\mathrm{d}\lambda.$$

As λ is a scale parameter, an appropriate ignorance prior is given by the improper Jeffrey's prior, $p(\lambda) \propto 1/\lambda$, corresponding to a uniform prior over $\log \lambda$. Substituting Equation (6) and noting that λ is strictly positive,

$$p(\boldsymbol{\alpha}) = \frac{1}{2^N}\int_0^\infty \lambda^{N-1}\,\exp\{-\lambda E_{\boldsymbol{\alpha}}\}\,\mathrm{d}\lambda.$$

Using the Gamma integral, $\int_0^\infty x^{\nu-1}\mathrm{e}^{-\mu x}\,\mathrm{d}x = \frac{\Gamma(\nu)}{\mu^\nu}$ (Gradshteyn and Ryzhic, 1994, equation 3.384), we obtain

$$p(\boldsymbol{\alpha}) = \frac{1}{2^N}\frac{\Gamma(N)}{E_\alpha^N} \Rightarrow \quad -\log p(\boldsymbol{\alpha}) \propto N\log E_\alpha,$$

giving a revised optimization criterion for sparse logistic regression with Bayesian regularization,

$$Q = E_{\mathcal{D}} + N\log E_\alpha, \qquad (7)$$

in which the regularization parameter has been eliminated, for further details and theoretical justification, see Williams (1995). The use of a Laplace prior and Jeffrey's hyper-prior in sparse supervised learning has also been proposed by Figueiredo (2003), along with an Expectation-Maximization (EM) style training algorithm. Unfortunately this training procedure involves solving a system of ℓ linear equations, analogous to the normal equations to be solved in linear regression, and so is not suitable for large-scale applications, such as the analysis of microarray data. Fortunately the method of Shevade and Keerthi (2003) can easily be adapted to sparse logistic regression with Bayesian regularization.

3.2 Minimizing the Bayesian training criterion

Shevade and Keerthi (2003) demonstrate that the cost function for sparse logistic regression using a Laplace prior can be iteratively minimized in an efficient manner one parameter at a time. Note that the objective function is non-smooth, as the first derivatives exhibit discontinuities at $\alpha_i = 0$, $\forall i \in \{1, 2, \ldots, N\}$, but is otherwise smooth. These properties of the objective function are clearly evident from the first and second derivatives,

$$\frac{\partial}{\partial \alpha_i} \log E_\alpha = \frac{\alpha_i}{|\alpha_i|} \frac{1}{E_\alpha} \quad \text{and} \quad \frac{\partial^2}{\partial \alpha_i^2} \log E_\alpha = -\frac{1}{E_\alpha^2}.$$

The training criterion incorporating a fully Bayesian regularization term can be minimized via a simple modification of the existing training algorithm for sparse logistic regression. Differentiating the original and modified training criteria (5,7), we have that

$$\nabla M = \nabla E_{\mathcal{D}} + \lambda \nabla E_\alpha \quad \text{and} \quad \nabla Q = \nabla E_{\mathcal{D}} + \tilde{\lambda} \nabla E_\alpha,$$

where

$$\frac{1}{\tilde{\lambda}} = \frac{1}{N} \sum_{i=1}^{N} |\alpha_i|. \tag{8}$$

From a gradient descent perspective, Minimizing Q effectively becomes equivalent to minimizing M, assuming that the regularization parameter, λ, is continuously updated according to (8) following every change in the vector of model parameters, $\boldsymbol{\alpha}$ (Williams, 1995). This requires only a very minor modification of the code implementing the sparse logistic regression algorithm, whilst eliminating the only training parameter and hence the need for a model selection procedure in fitting the model.

3.3 Relationship with the evidence framework

It has been observed that the 'integrate-out' approach to deal with the regularization parameter (Buntine and Weigend, 1991) is likely to lead to over-regularized models that under-fit the data, for neural network models with a traditional Gaussian weight-decay prior, and that evidence framework (MacKey, 1992a, b, c) is generally to be preferred (MacKay, 1994). However, it is relatively straight forward to show that, in the case of the Laplace prior, the iterative update formula for the effective regularization parameter (8) is identical to the update formula for the regularization parameter under the evidence framework (Williams, 1995).

4 RESULTS

In this section, we evaluate the performance of the proposed logistic regression method with Bayesian regularization using a Laplace prior against the sparse logistic regression method of Shevade and Keerthi (2003), on which it is based, and the RVM (Tipping, 2001), which represents the most direct competing approach. The performance of all three classifiers are evaluated over two commonly used benchmark datasets: the colon cancer dataset, introduced by Alon *et al*. (1999) and the leukaemia dataset introduced by Golub *et al*. (1999). Sections 4.1–4.3 outline key aspects of the experimental methodology, the experimental results are given in Sections 4.4 and 4.5.

4.1 Performance evaluation

Cross-validation (Stone, 1974) is a commonly used procedure for evaluating the quality of statistical models. In k-fold cross-validations, the available data are partitioned into k disjoint sub-sets of approximately equal size. A set of k classifiers is then constructed; each classifier is trained on a different combination of $k-1$ sub-sets and tested on the remaining sub-set. The average test performance of the k classifiers generally provides a good estimate of the generalization performance of a single classifier trained on the entire dataset. Cross-validation is especially attractive in applications with relatively limited amounts of data as all observations are used as both training and test data. The most extreme form of cross-validation, where each partition contains a single pattern, is known as leave-one-out cross-validation, and has been shown to provide an almost unbiased estimate of the true test error (Luntz and Brailovsky, 1969). Leave-one-out cross-validation is rarely used in performance evaluation owing its high computational expense, but also because is has been observed to exhibit a higher variance than conventional k-fold cross-validation (Kohavi, 1995). However, if the amount of available data are severely limited, leave-one-out cross-validation becomes the more attractive option, and not only because the computational expense involved becomes less of an issue. In these circumstances, k-fold cross-validation can also exhibit a high variance because more data are held out for testing in each fold, and the classifiers may then have too little data to form a stable decision rule (i.e. a small change in the training data may lead to a significant change in the decision rule). The estimator then becomes sensitive to issues such as the partitioning of the data. In microarray analysis, we typically have only a few tens or hundreds of training patterns with a few thousand features. We therefore use leave-one-out cross-validation as it is likely to provide a more reliable indicator of generalization performance than five- or ten-fold cross-validation.

4.2 The relevance vector machine

The RVM is included in the evaluation as it implements a logistic regression model, where the amount of regularization applied and the degree of sparsity obtained are also governed within a Bayesian framework, rather than by an explicit parameter which must be tuned by the user. The algorithm therefore generates a model of the same form as the proposed method, also without the need for a model selection stage to choose good settings for any hyper-parameters. The RVM, however, is based on a separable Gaussian prior over the model parameters, with a distinct regularization parameter for each weight. The regularization parameters are adjusted so as to maximize the marginal likelihood of the model, which tends to force the values of redundant weights strongly towards zero, so that they can be identified and pruned from the model, via a process known as automatic relevance determination (ARD). The implementation used here is based on the fast marginal likelihood approach described by Faul and Tippline (2002, 2003), however rather than re-fitting the Laplace approximation after each update of a regularization, it is only updated when a significant increase in the marginal likelihood is no longer possible via updates of the regularization parameters. This strategy was found to be considerably faster in practice.

4.3 Model selection for sparse logistic regression

The existing sparse logistic regression model of Shevade and Keerthi (2003) includes a regularization parameter, controlling the complexity of the model and the sparsity of the model parameters, which must be chosen by the user or alternatively optimized in an additional model selection stage. In this study, the value of this parameter is found via a (computationally expensive) minimization of the leave-one-out cross-validation estimate of the cross-entropy loss. Again, leave-one-out cross-validation is appropriate as the amount of training data is severely limited. However, we cannot use the same leave-one-out cross-validation estimate for both model

Table 2. Leave-one-out cross-validation estimate of the cross-entropy and error rate for RVM, SLogReg and BLogReg algorithms on the colon benchmark and a bootstrap estimate of the average number of features used.

Algorithm	Cross-entropy	Error rate	# Features
RVM	0.567 ± 0.178	0.177 ± 0.049	5.60 ± 0.040
SLogReg	0.506 ± 0.094	0.177 ± 0.049	15.54 ± 0.103
BLogReg	0.510 ± 0.098	0.177 ± 0.049	11.74 ± 0.033

All averages are given with the associated standard error of the mean.

selection and performance evaluation as this would introduce a (possibly quite strong) selection bias in favour of the existing sparse logistic regression model. A nested leave-one-out cross-validation procedure is therefore used instead. Leave-one-out cross-validation is used for performance evaluation in the 'outer loop' of the procedure, in each iteration of which model selection is performed individually for each classifier based on a separate leave-one-out cross-validation procedure. Obviously this is computationally expensive, but provides an almost unbiased assessment of generalization performance as well as a sensible automatic method of setting the value of the regularization parameter.

4.4 Results on the colon cancer dataset

The colon cancer dataset (Alon *et al*., 1999) describes the expression of 2000 genes in 40 cancer and 22 normal tissue samples, the aim being to construct a classifier capable of distinguishing between cancer and normal tissues. Table 2 shows the leave-one-out cross-validation estimate of the cross-entropy and error rate for the RVM, SLogReg and BlogReg algorithms over the colon dataset. All three classifiers achieve the same error rate of 17.7%. The cross-entropy provides a more refined indicator of the discriminative ability of a classifier, and in this case shows that the SLogReg and BLogReg algorithms clearly outperform the RVM, but that the difference in performance between the SLogReg and BLogReg algorithms is minimal. The higher cross-entropy of the RVM is however offset by the use of a much smaller sub-set of the available features. Figure 2 shows the frequency of selection and the mean weight for each feature comprising the colon dataset for RVM, SLogReg and BlogReg algorithms, averaged over 1000 bootstrap realizations of the data. Note that in each case, a small sub-set of features are selected on a regular basis with significant weights, however the sub-sets of features chosen in each iteration exhibit substantial variation. The BLogReg algorithm is marginally more expensive than the RVM, each fold of the leave-one-out cross-validation procedure taking on average 1.03 and 0.84 s respectively. The SLogReg algorithm is very much more expensive, owing to the need for a model selection stage to choose a good value for the regularization parameter, λ, with each fold of the leave-one-out cross-validation procedure taking ~ 317 s. The choice of algorithm is then dependent on whether sparsity or predictive power (as measured by the cross-entropy) is most important. Given the minimal difference in performance and substantial difference in computational expense there is little reason to prefer the SLogReg over the BLogReg algorithm.

4.5 Results on the leukaemia dataset

The aim of the leukaemia benchmark (Golub *et al*., 1999) is to form a decision rule capable of distinguishing between acute myeloid leukaemia (AML) and acute lymphoblastic leukaemia (ALL). The data describe the expression of 7128 genes in 47 ALL samples and 25 AML samples. The original benchmark partitions the data into training and test sets, however as leave-one-out cross-validation is used in this study, owing to the small size of the dataset, we have neglected this division. Again the leave-one-out cross-validation error rates are very similar, with the original SLogReg model generating four leave-one-out errors and the BLogReg and RVM models both generating five. The RVM and SLogReg models both produce very sparse models, however the BLogReg model still uses on average only 11.59 of the 7128 features (only 0.16%). However, the additional features used by the BLogReg model do provide additional discriminatory power, as the BLogReg model achieves the lowest cross-entropy score. The BlogReg model is slightly faster than the RVM model, in this case each fold of the leave-one-out cross-validation process taking an average of 1.16 s for the BlogReg model and 2.78 s for the RVM model. The BLogReg model is however more than three orders of magnitude faster than the existing SLogReg model, which requires an average of 2392.2 s, owing to the model selection stage required to optimize the value of the regularization parameter (Fig. 3 and Table 3).

4.6 Discussion

It is interesting to compare the BLogReg and RVM approaches from a theoretical perspective as BLogReg, which integrates out the hyper-parameters and subsequently optimizes the parameters implements a strategy that is diametrically opposed to that of the RVM, which integrates over the model parameters and optimizes the hyper-parameters. The proper Bayesian approach to dealing with hyper-parameters seeks to define a suitable hyper-prior and integrate them out analytically (Buntine and Weigend, 1991; Willams, 1995), or via Markov Chain Monte Carlo (MCMC) methods (Williams and Rasmussen, 1996). On the other hand, MacKay (1999) notes that, at least in the case of a Gaussian prior, optimizing the hyper-parameters under the evidence framework is often preferable. In the presence of many ill-defined parameters, the integrate-out approach often leads to over-regularization of the model, also the skewness of the posterior means that the usual Laplace approximation is not representative of the volume of the true posterior under the integrate-out approach. However, in the case of the Laplace prior, the pruning action of the regularizer eliminates any ill-determined parameters, and so the model will not generally be overly regularized. Also the model being sparse and composed solely of well-determined parameters, the posterior is likely to be comparatively compact, and so the Laplace approximation under the Laplace prior can be expected to be relatively accurate. Indeed, the integrate-out and optimization approaches have been shown to be equivalent in the case of the Laplace prior (Williams, 1995). The theoretical justification for the BLogReg algorithm is thus at least as sound as that of the RVM. The BLogReg and RVM have also been evaluated for the task of discriminative detection of regulatory elements (Cawley *et al*. 2006a), where the BLogReg algorithm generally out-performed the RVM, although the difference in performance was relatively slight.

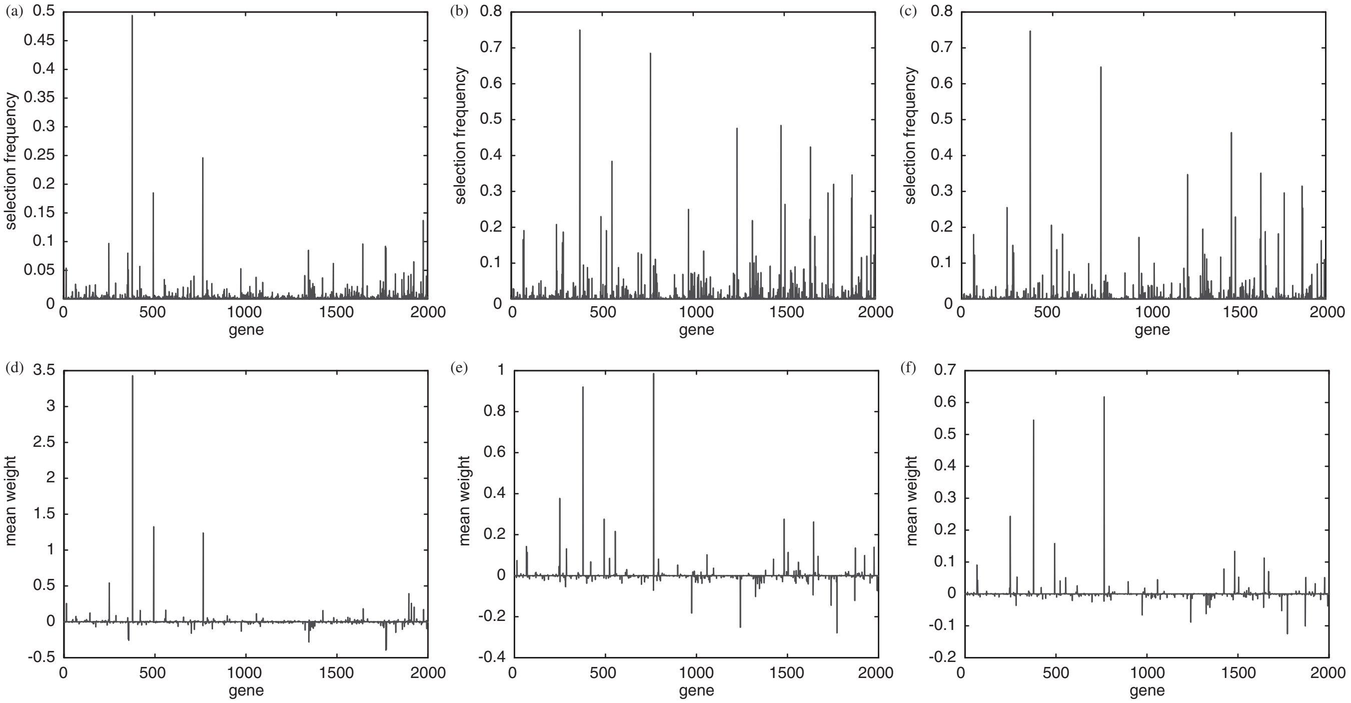

Fig. 2. Frequency of selection for the genes comprising the colon dataset computed over 1000 bootstrap realizations of the data. Note that the usage of features is a little different for the RVM (**a**) and SLogReg (**b**) or BLogReg (**c**) models, however, the usage of features is very similar for the SLogReg and BLogReg models. Also shown are the mean weights associated with each gene, Note that again the feature weights are a little different for the RVM (**d**) and SLogReg (**e**) or BLogReg (**f**) models, but the weights obtained from the SLogReg and BLogReg algorithms are very similar.

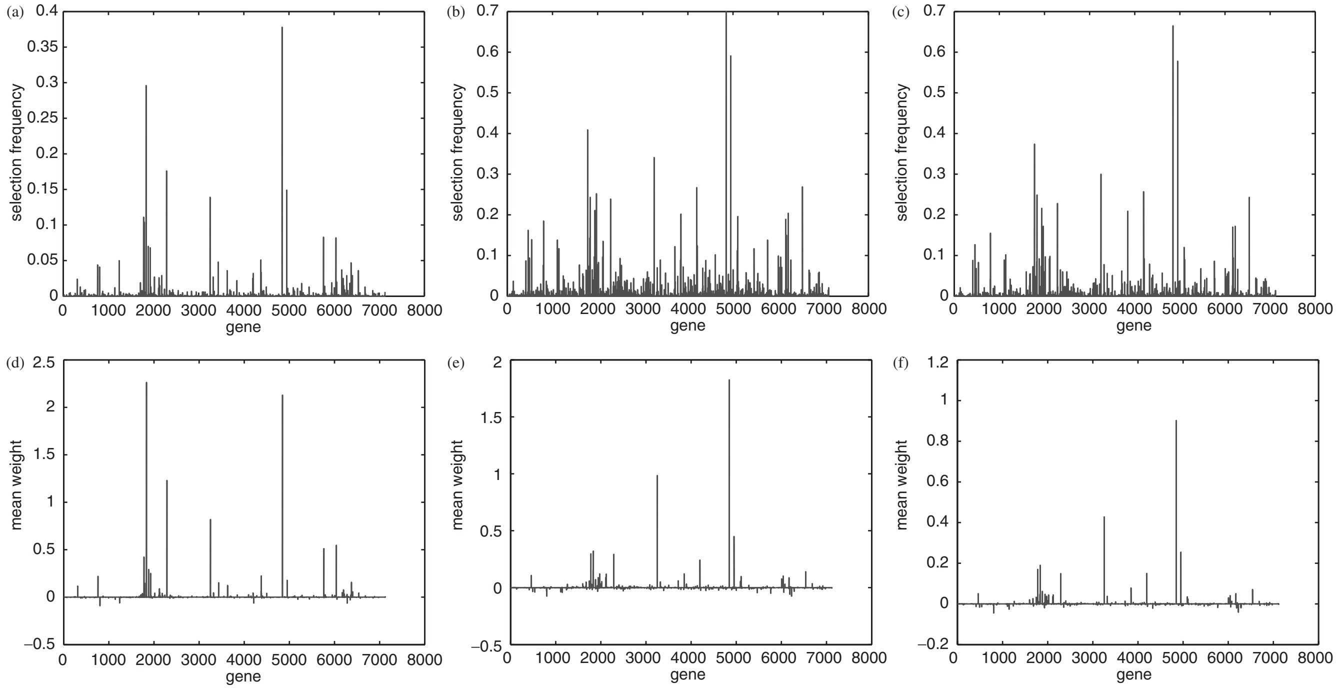

Fig. 3. Frequency of selection for the genes comprising the leukaemia dataset computed over 1000 bootstrap realizations of the data. Note that the usage of features is a little different for the RVM (**a**) and SLogReg (**b**) or BLogReg (**c**) models, however, the usage of features is very similar for the SLogReg and BLogReg models. Also shown are the mean weights associated with each gene, Note that again the feature weights are a little different for the RVM (**d**) and SLogReg (**e**) or BLogReg (**f**) models, but the weights obtained from the SLogReg and BLogReg algorithms are very similar.

Table 3. Leave-one-out cross-validation estimate of the cross-entropy and error rate for RVM, SLogReg and BLogReg algorithms on the leukaemia benchmark and a bootstrap estimate of the average number of features used.

Algorithm	Cross-entropy	Error rate	# Features
RVM	0.878 ± 0.674	0.069 ± 0.030	3.63 ± 0.040
SLogReg	0.359 ± 0.220	0.055 ± 0.027	5.06 ± 0.078
BLogReg	0.259 ± 0.081	0.069 ± 0.030	11.59 ± 0.064

All averages are given with the associated standard error of the mean.

The Jeffreys prior is used here as this is the standard reference prior for a scale parameter (Jeffreys, 1961), expressing ignorance of the true value on a logarithmic scale. The other advantage of the Jeffreys prior is entirely practical, in that it results in an analytic solution for the desired integral, giving rise to the modified training criterion (7). It is likely that good results may be obtained using other reasonable hyper-priors, although the resulting training algorithm is likely to be of a less convenient form.

Selection bias is an important issue in performance evaluation of cancer classification algorithms (Ambroise and McLachlan, 2002). The proposed BLogReg is essentially free from selection bias as it is self-contained, without parameters that must be optimized during model selection. The use of a nested leave-one-out cross-validation procedure, in order to optimize the regularization parameter without incurring selection bias difficulties, makes the SLogReg algorithm essentially impractical. In the case of the leukaemia dataset, unbiased performance estimation for the SLogReg algorithm took almost two days on a modern PC, but <2 min using the BLogReg algorithm. Given the minimal difference in generalization performance between the original and improved sparse logistic regression models, there is little practical reason not to prefer the BLogReg variant. More recently, multinomial variants of the BLogReg and SLogReg algorithms were evaluated over a suite of nine benchmark datasets (Cawley *et al*. 2006b). Again the differences in generalization performance were generally minimal, with the BLogReg algorithm being generally two or three orders of magnitude faster.

It should be noted that in this study, the RVM produces models with significantly fewer input features, than the other models. However, this is achieved at the expense of the accuracy of the conditional probability generated by the model (as measured by the leave-one-out cross-entropy statistic). Qi *et al*. (2004) also show that the RVM selects too few input features, and therefore under-fits the data in cancer-classification using gene-expression data. Accurate estimation of conditional probabilities is essential in statistical decision making (Berger, 1985), especially in medical applications where false-positive and false-negative costs may be different, or where we may wish to reject an uncertain diagnosis in favour of performing additional tests. Unless the focus is on identifying the smallest possible set of biomarker genes, rather than predictive performance, the BLogReg algorithm is likely to be the better option.

5 CONCLUSIONS

In this paper we demonstrate that the regularization parameter arising in the sparse logistic regression algorithm (SLogReg) of Shevade and Keerthi (2003) can be eliminated, via Bayesian marginalization, without a significant effect on predictive performance. Results on the well-studied colon cancer and leukaemia benchmarks clearly demonstrate that the proposed algorithm for sparse logistic regression with Bayesian regularization (BLogReg) is competitive with the original SLogReg and RVM algorithms in terms of performance and sparsity. However, as the need for a cross-validation based model selection process is obviated, the improved algorithm is two to three orders of magnitude faster than its predecessor. The computational expense of eliminating selection bias for the existing SLogReg algorithm is shown to be prohibitive; in this study estimation of the error rate for the SLogReg algorithm took just under five and a half hours, whereas a comparable estimate for the BLogReg algorithm took just over one minute, clearly a very significant improvement. The absence of a model selection stage also automatically eliminates any risk of selection bias in the estimation of the test error rate. Further work will investigate the use of Bayesian logistic regression using, e.g. radial basis functions as an alternative to the RVM in a more general non-linear pattern recognition setting.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their helpful and constructive comments. This work was supported by a grant from the Biotechnology and Biological Sciences Research Council (BBSRC), grant number 83/EGM16128.

Conflict of Interest: none declared.

REFERENCES

Alon,U. *et al*. (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl Acad. Sci. USA*, **96**, 6745–6750.

Ambroise,C. and McLachlan,G.J. (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc. Natl Acad. Sci. USA*, **99**, 6562–6566.

Berger,J.O. (1985) *Statistical Decision Theory and Bayesian Analysis*, *Springer Series in Statistics*, 2nd edn, Springer.

Buntine,W.L. and Weigend,A.S. (1991) Bayesian back-propagation. *Complex Syst*., **5**, 603–643.

Cawley,G.C. *et al*. (2006a) Discriminative detection of regulatory elements. *Bioinformatics*, (submitted).

Cawley,G.C., Talbot,N.L.C. and Girolami,M. (2006b) Sparse multinomial logistic regression via Bayesian regularisation using a Laplace prior. In *Neural Information Processing Systems*. MIT Press, Vancouver, BC, Canada, (submitted).

Chu,W. *et al*. (2005) Biomarker discovery in microarray gene expression data with Gaussian processes. *Bioinformatics*, **21**, 3385–3393.

Faul,A.C. and Tipping,M.E. (2002) Analysis of sparse Bayesian learning. In Dietterich,T.G., Becker,S. and Ghahramani,Z. (eds), *Advances in Neural Information Processing Systems*. MIT Press, Vol. **14,** pp. 383–389.

Faul,A.C. and Tipping,M.E. (2003) Fast marginal likelihood maximisation for sparse Bayesian models. In Bishop,C.M. and Frey,B.J. (eds), *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*. Key West, FL, USA.

Figueiredo,M. (2003) Adaptive sparseness for supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell*., **25**, 1150–1159.

Golub,T.R. *et al*. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531–537.

Gradshteyn,I.S. and Ryzhic,I.M. (1994) *Table of Integrals, Series and Products*. 5th edn. Academic Press.

Guyon,I. *et al*. (2002) Gene selection for cancer classification using support vector machines. *Mach. Learn*., **46**, 389–422.

Jeffreys,H. (1961) *Theory of Probability*. *Oxford Classic Texts in the Physical Sciences*. 3rd edn, Oxford University Press.

Kohavi,R. (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence (IJCAI),* San Mateo, CA, Morgan Kaufmann, pp. 1137–1143.

Lehrach,W.P. *et al*. (2006) A regularized discriminative model for the prediction of peptide-peptide interactions. *Bioinformatics*, **22**, 532–540.

Li,Y. *et al*. (2002) Bayesian automatic relevance determination algorithms for classifying gene expression data. *Bioinformatics*, **18**, 1332–1339.

Luntz,A. and Brailovsky,V. (1969) On estimation of characters obtained in statistical procedure of reognition (in Russian). *Technicheskaya Kibernetica*, **3**.

MacKay,D.J.C. (1992a) Bayesian interpolation. *Neural Comput*., **4**, 415–447.

MacKay,D.J.C. (1992b) The evidence framework applied to classification networks. *Neural Comput*., **4**, 720–736.

MacKay,D.J.C. (1992c) A practical Bayesian framework for backprop networks. *Neural Comput*., **4**, 448–472.

MacKay,D.J.C. (1994) Hyperparameters: optimise or integrate out?. In Heidbreder,G. (ed.), *Maximum Entropy and Bayesian Methods*. Kluwer.

MacKay,D.J.C. (1999) Comparison of approximate methods for handling hyperparameters. *Neural Netw*., **11**, 1035–1068.

Qi,Y., Minka,T., Picard,R.W. and Ghahramani,Z. (2004) Predictive automatic relevance determination by expectation propagation. In *Proceedings of the International Conference on Machine Learning (ICML-2004),* Banff, Alberta, Canada, pp. 85–92.

Shevade,S.K. and Keerthi,S.S. (2003) A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics*, **19**, 2246–2253.

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

Tan,A.C. *et al*. (2005) Simple decision rules for classifying human cancers from gene expression profiles. *Bioinformatics*, **21**, 3896–3904.

Tikhonov,A.N. and Arsenin,V.Y. (1977) *Solutions of Ill-Posed Problems*. John Wiley, New York.

Tipping,M.E. (2001) Sparse Bayesian learning and the relevance vector machine. *J. Mach. Learn. Res*., **1**, 211–244.

Williams,C.K.I. and Rasmussen,C.E. (1996) Gaussian processes for regression. *Neural Information Processing Systems 8*. Morgan Kaufmann, pp. 514–520.

Williams,P.M. (1995) Bayesian regularization and pruning using a Laplace prior. *Neural Comput*., **7**, 117–143.