BIOINFORMATICS

Vol. 18 no. 10 2002
Pages 1332–1339

Bayesian automatic relevance determination algorithms for classifying gene expression data

Yi Li[1,†], *Colin Campbell*[1,*] *and Michael Tipping*[2]

[1]Department of Engineering Mathematics, University of Bristol, Bristol, BS8 1TR, UK and [2]Microsoft Research, 7 J J Thomson Avenue, Cambridge, CB3 0FD, UK

Received on November 14, 2001; revised on April 22, 2002; accepted on April 26, 2002

ABSTRACT

Motivation: We investigate two new Bayesian classification algorithms incorporating feature selection. These algorithms are applied to the classification of gene expression data derived from cDNA microarrays.

Results: We demonstrate the effectiveness of the algorithms on three gene expression datasets for cancer, showing they compare well with alternative kernel-based techniques. By automatically incorporating feature selection, accurate classifiers can be constructed utilizing very few features and with minimal hand-tuning. We argue that the feature selection is meaningful and some of the highlighted genes appear to be medically important.

Contact: C.Campbell@bris.ac.uk

INTRODUCTION

The recent development of cDNA microarray technology is creating a wealth of gene expression data. Typically these datasets have a high dimensionality corresponding to the large number of probes used and there are often comparatively few examples. As an example, a recent leukaemia dataset (Golub *et al.*, 1999) has 72 examples with 7129 features each. Viewed as a machine learning problem, the high dimensionality and sparsity of datapoints has suggested the use of support vector machines (SVMs). For example, for binary classification, the generalization performance of an SVM does not depend on the dimensionality of the space but on maximizing the margin, γ, or distance between the separating hyperplane and the closest points of each class. Also the high-dimensional input vector $\mathbf{x}_i$ is absorbed in the kernel matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ where i, j are pattern indices. Thus training times follow the reduced dimensionality of the example set size rather than the number of features.

Several papers have reported results on the application of SVMs to classification of gene expression data. For example, Brown *et al.* (2000) considered a dataset from the budding yeast S. Cerevisiae with 2467 features. SVMs outperformed Parzen windows, Fisher's Linear discriminant and two decision tree classifiers. Furey *et al.* (2000) considered three datasets for ovarian cancer (Schummer *et al.*, 1999), colon cancer (Notterman *et al.*, 2001) and a dataset (Golub *et al.*, 1999) for distinguishing acute myeloid leukaemia (AML) and acute lymphoblastic leukaemia (ALL). The authors reported low test errors for these datasets despite the small number of tissue samples available for investigation.

Apart from achieving a low test error it would also be advisable to implement feature selection. By removing redundant features it may be possible to reduce the test error further. Though this reduction in the test error depends on the algorithm and dataset considered, it can be significant. For example, for the leukaemia dataset mentioned in section 3.3, the number of test errors is 3.0 ± 0.1 with all 7129 features against 1.7 ± 0.1 with 128 key features (features ranked by Fisher score, trained using an SVM and with a hundred $36 + 36$ resamplings from a merged data). Feature selection also simplifies the hypothesis and may highlight those genes which are most relevant.

Thus several authors have investigated feature selection in this context. Furey *et al.* (2000) performed feature selection using the Fisher score prior to training. Weston *et al.* (2001) compared use of the Fisher score against feature selection using generalization bounds from statistical learning theory. Guyon *et al.* (2002) further introduced an algorithm, called recursive feature elimination (RFE), in which features are successively eliminated during training of a sequence of SVM classifers. Since SVMs perform well on these datasets and the RFE algorithm appears to be one of the best kernel-based methods for implementing feature selection in this context we have used it as our comparative benchmark in the experiments described below.

Rather than using generalization bounds from statistical learning theory, an alternative approach is to exploit the

[*]To whom correspondence should be addressed.

[†] Present address: Information and Mathematical Sciences, Genome Institute of Singapore, 1 Science Park Road, The Capricorn #05-01, Singapore 117528, Republic of Singapore

© Oxford University Press 2002

Bayesian technique of *automatic relevance determination* (ARD) to perform feature selection (MacKay, 1994; Neal, 1994). An ARD approach has been used previously for constructing a classifier which is sparse in the number of examples i.e. the relevance vector machine (RVM) of Tipping (2000, 2001). For example, for two well-known real-life datasets (Pima Indian diabetes and USPS handwritten digit recognition) the RVM constructed the classifier using 4 examples rather than 109 (diabetes) and 316 examples rather than 2540 (USPS) (Tipping, 2000). In addition, these examples are distinct from those used by an SVM since they tend to represent prototypical examples rather than datapoints close to the decision boundary (the *support vectors* of a SVM).

In this paper we use the same Bayesian approach to outline two algorithms which are sparse in terms of the number of features used. Algorithm 1 is effectively the 'dual' of the standard RVM with sparsity obtained in the feature set rather than the example set (unlike the standard RVM we also allow reintroduction of features during the learning process). In the second algorithm feature selection is performed by isolating the feature dependence in the log-marginal likelihood function. As we will see these algorithms have similar performance to SVMs when applied to gene expression datasets from cDNA microarrays. Theoretically they are interesting because the motivation behind the approach is Bayesian in contrast to the statistical learning theory approach underpinning SVMs. However, they also have the advantage that feature sparsity is naturally incorporated into the algorithm—the optimal number of relevant features is decided automatically. By constrast, for an SVM, an additional feature selection procedure has to be added and a further criterion must be used to indicate when the best feature set has been found. In terms of practical use, these algorithms may highlight the importance of certain genes and create simpler hypotheses for separating classes (such as the genetic subtypes of a cancer). Thus, for colon cancer, we will see that it strongly highlights a feature which has recently been found to have therapeutic significance. In the next section we describe these algorithms in more detail and in Section 3 we will describe performance on three gene expression datasets for cancer.

THE ALGORITHMS

We will consider datasets consisting of m examples with n-dimensional input vectors $\mathbf{x}_\nu$ with corresponding target $y_\nu \in R$ (where ν are pattern indices). For reasons outlined below we will introduce the algorithms for regression before adapting them to classification. For regression we will assume the following *likelihood* for the data (Tipping, 2000):

$$p\left(\boldsymbol{y}|\boldsymbol{w},\sigma^2\right) = \left(2\pi\sigma^2\right)^{-\frac{m}{2}} \exp\left(-\frac{1}{2\sigma^2}\,||\boldsymbol{y}-\boldsymbol{\Phi}\boldsymbol{w}||^2\right) \tag{1}$$

where $\boldsymbol{w} = (w_0, \ldots, w_n)$. The gene expression datasets we will subsequently consider are generally linearly separable and for linear problems we use:

$$\boldsymbol{\Phi} = \begin{pmatrix} 1 & \boldsymbol{x}_1^T \\ & \cdots \\ 1 & \boldsymbol{x}_m^T \end{pmatrix}, \tag{2}$$

with the first column handling the bias w_0 in the hypothesis function:

$$f(z) = \sum_{j=1}^{n} w_j z_j + w_0 \tag{3}$$

where z_j is an assumed input vector (j are feature indices). Pruning the ith column of $\boldsymbol{\Phi}$ implements feature selection. As for the RVM we assume a prior favouring sparse hypotheses (Tipping, 2000), though sparse in features rather than sparse in examples:

$$p\left(\boldsymbol{w}|\boldsymbol{\alpha}\right) = (2\pi)^{-\frac{n+1}{2}} \prod_{i=0}^{n} \alpha_i^{1/2} \exp\left(-\frac{\alpha_i w_i^2}{2}\right) \tag{4}$$

where we have introduced 'hyperparameters' $\alpha_0 \ldots \alpha_n$ which control the 'strength' of the prior. We then integrate out the weights between (1) and (4) to obtain the *marginal likelihood*:

$$p(\boldsymbol{y}|\boldsymbol{\alpha},\sigma^2) = (2\pi)^{-\frac{m}{2}} \left|\boldsymbol{B}^{-1} + \boldsymbol{\Phi}\boldsymbol{A}^{-1}\boldsymbol{\Phi}^T\right|^{-\frac{1}{2}} . \quad \exp\left[-\frac{1}{2}\boldsymbol{y}^T\left(\boldsymbol{B}^{-1} + \boldsymbol{\Phi}\boldsymbol{A}^{-1}\boldsymbol{\Phi}^T\right)^{-1}\boldsymbol{y}\right] \tag{5}$$

where $\boldsymbol{A} = diag(\alpha_0, \alpha_1, \ldots, \alpha_n)$, $\boldsymbol{B} = \sigma^{-2}\boldsymbol{I}_m$ and we have additionally used the Woodbury–Sherman–Morrison matrix identity:

$$\left(\boldsymbol{B}^{-1} + \boldsymbol{\Phi}\boldsymbol{A}^{-1}\boldsymbol{\Phi}^T\right)^{-1} = \boldsymbol{B} - \boldsymbol{B}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T\boldsymbol{B} \tag{6}$$

with $\boldsymbol{\Sigma} = \left(\boldsymbol{\Phi}^T\boldsymbol{B}\boldsymbol{\Phi} + \boldsymbol{A}\right)^{-1}$. The posterior over the weights is:

$$p\left(\boldsymbol{w}|\boldsymbol{y},\boldsymbol{\alpha},\sigma^2\right) = (2\pi)^{-\frac{(n+1)}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} . \quad \exp\left[-\frac{1}{2}\left(\boldsymbol{w}-\boldsymbol{\mu}\right)^T\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{w}-\boldsymbol{\mu}\right)\right] \tag{7}$$

where the mean is given by $\boldsymbol{\mu} = \boldsymbol{\Sigma}\boldsymbol{\Phi}^T\boldsymbol{B}\boldsymbol{y}$. The algorithms we now detail are based upon exploiting the *type-II maximum likelihood* principle. The objective is to maximize the marginal likelihood (5) with respect to the hyperparameters in order to obtain point estimates of their values. These can then be substituted back into (7) to give an updated posterior distribution for the weights, which is typically summarized by its mean.

Algorithm 1

We can obtain an iterative update formula for the $\boldsymbol{\alpha}$ by taking the natural log of (5) and differentiating this expression with respect to $\boldsymbol{\alpha}$. Using the following formulae for this differentiation (with $\boldsymbol{M}$ an arbitrary matrix):

$$\frac{\partial \boldsymbol{M}^{-1}}{\partial \boldsymbol{\alpha}} = -\boldsymbol{M}^{-1}\left(\frac{\partial \boldsymbol{M}^{-1}}{\partial \boldsymbol{\alpha}}\right)\boldsymbol{M}^{-1} \tag{8}$$

$$\frac{\partial \ln |\boldsymbol{M}|}{\partial \alpha} = Tr\left(\boldsymbol{M}^{-1}\frac{\partial \boldsymbol{M}}{\partial \boldsymbol{\alpha}}\right) \tag{9}$$

we derive the following update formula for the $\boldsymbol{\alpha}$ at the optimum:

$$\alpha_i = \gamma_i/\mu_i^2 \tag{10}$$

where $\gamma_i = 1 - \alpha_i \Sigma_{ii}$ and:

$$(\sigma^2)^{new} = \frac{||\boldsymbol{y} - \boldsymbol{\Phi\mu}||^2}{\left(m - \sum_i \gamma_i\right)} \tag{11}$$

These formulae suggest an algorithmic approach to regression (Tipping, 2000) in which we iteratively update $\boldsymbol{\alpha}$ and σ and intermediate quantities such as $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. In practice, during re-estimation, many of the α_i approach infinity. For those α_i, the corresponding individual weight posteriors $p(w_i|\boldsymbol{y}, \boldsymbol{\alpha}, \sigma^2)$ become infinitely peaked at zero, implying that the corresponding ith column in $\boldsymbol{\Phi}$ can be 'pruned'. During execution of the algorithm with forward selection such features were removed and in the simulations described below we used $\alpha_i > 10^{12}$ as the pruning criterion. The re-introduction of pruned features was also allowed if the gradient of the marginal likelihood, (5), with respect to α_i is less than a bound. In the simulations below we used $\alpha_i^{-1} - \Sigma_{ii} - \mu_i^2 < -10^{-5}$. The numerical values of the lower bound for pruning and upper bound for re-introduction of features were found using experiments on artificial datasets. However, for these artificial datasets and the microarray datasets described below, we found very little change in performance on varying these bounds provided the lower bound for pruning is a very large number and the bound on re-introduction correspondingly small. The algorithm is run until a termination criterion is reached (which we took to be that the largest change to a finite α_i value was below a tolerance).

The above formulation is for regression and we must now consider its adaptation to classification. For classification where $y_\nu \in \{0, 1\}$, the algorithm is slightly different. The linear model is generalized by applying the logistic sigmoid function $g(f) = 1/(1 + e^{-f})$ to $f(\cdot)$. The likelihood of the dataset is then written as (Bishop, 1995):

$$p(\boldsymbol{y}|\boldsymbol{w}) = \prod_{\nu=1}^{m} g(f(\boldsymbol{x}_\nu))^{y_\nu}[1 - g(f(\boldsymbol{x}_\nu))]^{1-y_\nu}. \tag{12}$$

As a result of this modification, the weight posterior is no longer analytically obtainable, but for given values of $\boldsymbol{\alpha}$ an effective approximation can be obtained using a Gaussian centred at the maximum of $p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{\alpha})$. Finding this maximum, and hence the mean $\boldsymbol{\mu}$ of the approximating Gaussian, is equivalent to a standard optimization of a regularised logistic model and different methods can be applied to solve it (Tipping, 2001).

The covariance $\boldsymbol{\Sigma} = (-\nabla\nabla \log p(\boldsymbol{y}, \boldsymbol{w}|\boldsymbol{\alpha}))^{-1}$ of the approximating Gaussian is now equal to $(\boldsymbol{\Phi}^T\boldsymbol{B\Phi} + \boldsymbol{A})^{-1}$, where $\boldsymbol{B}$ is an $m \times m$ diagonal matrix with $B_{\nu\nu} = g(f(\boldsymbol{x}_\nu))[1 - g(f(\boldsymbol{x}_\nu))]$. The hyperparameters $\{\alpha_i\}$ are still updated as in (10). Noting the similar covariance matrices between regression and classification models, we can readily adapt the regression algorithm to the task of classification as follows (Tipping, 2001). Let $y_\nu \in \{-1, +1\}$, $\boldsymbol{A} = diag(\alpha_0, \alpha_1, \cdots, \alpha_n)$, and $\boldsymbol{B} = diag(1/\sigma_1^2, \cdots, 1/\sigma_m^2)$. We update the hyperparameters $\{\alpha_i\}$ using (10), but for $\{\sigma_\nu^2\}$, we instead use the formula (Tipping, 2001):

$$(\sigma_\nu^2)^{new} = 1/[g(f(\boldsymbol{x}_\nu))(1 - g(f(\boldsymbol{x}_\nu)))] \tag{13}$$

to determine $\mathbf{B}$. Since $g(f(\boldsymbol{x}_\nu))(1 - g(f(\boldsymbol{x}_\nu)))$ is at most 1/4, we can set σ_ν^2 to a constant for simplicity (for algorithm 2 in the next subsection we set σ_ν to 1 when applying regression likelihood to the task of classification). To predict the labels of test instances, we threshold $f(\cdot)$ in (3) at zero. In the experimental section we will use (13) in the simulations. However, as we note in that section, a small further reduction in the test error can be gained if we do not use (13) but only use (10) and optimize the σ value using further data.

Algorithm 2

An alternative strategy has recently been suggested by Faul and Tipping (2001). In constrast to the derivation of algorithm 1, we proceed by writing down the *log-marginal-likelihood*:

$$\begin{aligned} L(\alpha) &= \ln\left[p\left(\boldsymbol{y}|\boldsymbol{\alpha}, \sigma^2\right)\right] \\ &= -\frac{1}{2}\left(m\ln(2\pi) + \ln|\boldsymbol{C}| + \boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y}\right) \end{aligned} \tag{14}$$

where:

$$\boldsymbol{C} = \boldsymbol{B}^{-1} + \boldsymbol{\Phi A}^{-1}\boldsymbol{\Phi}^T \tag{15}$$

Table 1. The numbers of errors on the test set and feature set size for the colon cancer dataset with 50 training patterns and 12 test patterns. The SVM was trained with decreasing feature set size with a leave-one-out bound to determine the stopping point. 100 partitionings of the data were used

	Algorithm 1	Algorithm 2 (using $\sigma = 1$)	SVM (RFE)	SVM (Fisher score)
Number of errors on test data	2.04 ± 0.14	2.90 ± 0.13	2.84 ± 0.14	2.68 ± 0.15
Size of feature set	15.13 ± 0.31	8.55 ± 0.13	4.25 ± 0.12	14.41 ± 5.35

Table 2. The numbers of errors on the test set and feature set size for the ovarian cancer dataset with 40 training patterns and 14 test patterns. The SVM was trained with decreasing feature set size with a leave-one-out bound to determine the stopping point. 100 partitionings of the data were used

	Algorithm 1	Algorithm 2 (using $\sigma = 1$)	SVM (RFE)	SVM (Fisher score)
Number of errors on test data	1.95 ± 0.11	1.80 ± 0.13	1.82 ± 0.13	1.89 ± 0.13
Size of feature set	9.62 ± 0.30	6.5 ± 0.13	2.66 ± 0.07	17.24 ± 3.14

with $\boldsymbol{B} = \sigma^{-2}\boldsymbol{I}_m$. Here, and in the derivation below, we will assume σ is a constant.

In order to isolate the influence of an individual feature i, we decompose $\boldsymbol{C}$ as:

$$\boldsymbol{C} = \boldsymbol{C}_{-i} + \alpha_i^{-1}\boldsymbol{\Phi}_i\boldsymbol{\Phi}_i^T \tag{16}$$

where $\boldsymbol{\Phi}_i$ is the ith column of $\boldsymbol{\Phi}$. Using the Woodbury–Sherman–Morrison formula for the inverse of $\boldsymbol{C}$:

$$\boldsymbol{C}^{-1} = \boldsymbol{C}_{-i}^{-1} + \frac{\boldsymbol{C}_{-i}^{-1}\boldsymbol{\Phi}_i\boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}}{\alpha_i + \boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}\boldsymbol{\Phi}_i} \tag{17}$$

we can write:

$$L(\alpha) = L(\alpha_{-i}) + R(\alpha_i) \tag{18}$$

where:

$$R(\alpha_i) = \frac{1}{2}\left[\ln(\alpha_i) - \ln\left(\alpha_i + \boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}\boldsymbol{\Phi}_i\right) + \frac{\left(\boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}\boldsymbol{y}\right)^2}{\left(\alpha_i + \boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}\boldsymbol{\Phi}_i\right)}\right] \tag{19}$$

Conveniently, all dependencies on α_i (and its corresponding feature) are isolated in $R(\alpha_i)$. Taking the derivative of $L(\alpha)$ with respect to α_i we then get:

$$\frac{\partial L(\alpha)}{\partial \alpha_i} = \frac{1}{2}\left[\frac{S_i + \alpha_i^{-1}S_i^2 - Q_i^2}{(\alpha_i + S_i)^2}\right] \tag{20}$$

where for convenience we define $Q_i = \boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}\boldsymbol{y}$ and $S_i = \boldsymbol{\Phi}_i^T\boldsymbol{C}_{-i}^{-1}\boldsymbol{\Phi}_i$. As shown in Faul and Tipping (2001), if $Q_i^2 > S_i$ then analysis of the second derivative indicates that $R(\alpha_i)$ is maximized at $\alpha_i = S_i^2/(Q_i^2 - S_i)$. If $Q_i^2 \leq S_i$ then considering the asymptotic behaviour of the first derivative demonstrates that $R(\alpha_i)$ is maximized at $\alpha_i = \infty$, equivalent to the removal of feature i.

These properties suggest a sequential algorithm for maximizing the marginal likelihood where at each iteration we compute (efficiently) optimal new $\boldsymbol{\alpha}$ values individually for every feature. Note, importantly, that we can do this for features that are currently excluded (i.e. where $\alpha_i = \infty$). It is then straightforward to compute the change in $L(\boldsymbol{\alpha})$ resulting from the individual optimization of each α_i, and then select that change which most increases $L(\boldsymbol{\alpha})$. At each iteration we may therefore:

- introduce a feature when $\alpha_i = \infty$ but $Q_i^2 > S_i$,
- exclude a feature when $\alpha_i < \infty$ but $Q_i^2 \leq S_i$,
- update a current feature (re-estimate α_i) when $\alpha_i < \infty$ and $Q_i^2 > S_i$.

When the algorithm has terminated (according to a similar criterion as previously stated) we then find $\boldsymbol{\Sigma} = \left(\boldsymbol{\Phi}^T\boldsymbol{B}\boldsymbol{\Phi} + \boldsymbol{A}\right)^{-1}$ and hence determine the mean values for the weights via $\boldsymbol{\mu} = \boldsymbol{\Sigma}\boldsymbol{\Phi}^T\boldsymbol{B}\boldsymbol{y}$.

In the experiments below we set $\sigma = 1$, though, as for algorithm 1, a small gain in test error reduction can be achieved if σ is optimized using further data. In the simulations presented below we also found use of the regression likelihood (1) with thresholding gave marginally lower test errors than use of the classification likelihood (12). Consequently we used the regression likelihood in our experiments (as an illustration, in reference to Tables 1 and 2 the test errors were 3.26 ± 0.13 and 1.81 ± 0.13 respectively using the classification likelihood with algorithm 2).

EXPERIMENTAL RESULTS

We now report the test performance of these algorithms on gene expression datasets for colon cancer, ovarian cancer and the leukaemia dataset. For the first dataset (colon cancer) the data had been preprocessed before presentation to the classifier. In order to compare with the RFE algorithm of Guyon *et al.* (2002) we used exactly the same data and pre-processing steps: the log of all values was taken, sample and feature vectors were then normalized and the values passed through a $tanh$-function to diminish the effect of outliers. For ovarian cancer the data supplied by the experimentalists did not require further pre-processing by the authors and for leukaemia we used the standard presentation of the data (Proceedings of CAMDA, 2000).

We will also compare the test errors with those obtained using an SVM with Fisher score feature ranking or recursive feature elimination (RFE). Both these SVM algorithms must use a stopping criterion to halt the feature selection process at an appropriate point. A good procedure would be to use leave-one-out (LOO) cross validation each time a feature is eliminated. However, this process becomes too computationally intensive for most datasets encountered. Instead we have used the efficient leave-one-out bound of Joachims (2000) each time a feature is eliminated. This comparison of the algorithms will not make implicit use of the test data and so the reported performance on the test data is representative. However, for completeness, we will also give the lowest achieveable test error (which will make implicit use of the test data). For an SVM this will be the lowest averaged test error across all feature set sizes and for the Bayesian algorithms, where the feature set is automatically determined, we give the lowest averaged test error across a range of σ values.

An important point to note is that current gene expression datasets are characterized by few datapoints in a very high-dimensional space. Thus separability and low test errors may not be a surprise. Instead the problem may be that the hypotheses generated are not unique: many hypotheses could fit the data well, each using a distinct set of features. To investigate this issue we will randomly partition the data into two disjoint subsets of equal size and train the algorithm on both sets. After training we find the number of features common to both hypotheses. Since it is straightforward to determine the probability of randomly having common features, we can therefore quantify the extent to which feature selection is meaningful. Let n be the maximum number of possible features, n_1 the number of features in hypothesis 1, n_2 the number of features in hypothesis 2 and let N_c represent the number of features common to both hypotheses. Further let $n_a = \max(n_1, n_2)$ and $n_b = \min(n_1, n_2)$. If n_1 and n_2 features are drawn from n features uniformly at random, then the conditional probability of having n_c features in common is:

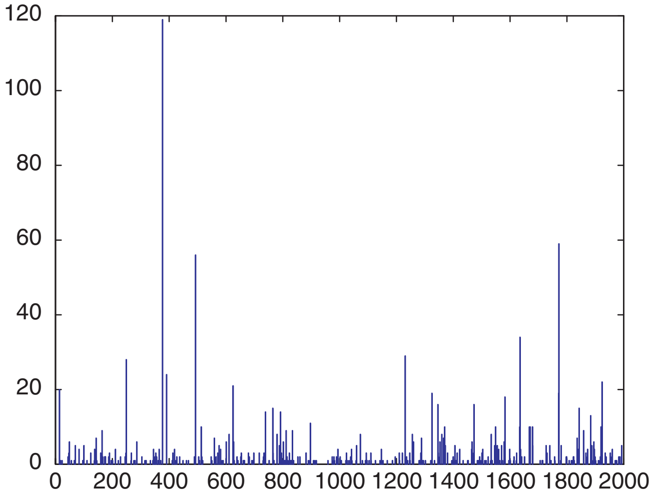

Fig. 1. For 100 partitionings of the colon cancer dataset the y-axis shows the number of occurrences of the feature (the maximum would be 200) and the x-axis shows the feature index.

$$Pr(N_c = n_c | n, n_1, n_2) = \frac{\begin{pmatrix} n_a \\ n_c \end{pmatrix} \begin{pmatrix} n - n_a \\ n_b - n_c \end{pmatrix}}{\begin{pmatrix} n \\ n_b \end{pmatrix}} \qquad (21)$$

where $n_b \geq n_c \geq max(0, n_a + n_b - n)$. For large n and small n_1, n_2, the probability that $N_c = 1$ or $N_c = 2$ is generally very small for the datasets considered here.

Colon cancer dataset

For the colon cancer (Alon *et al.*, 1999) dataset the task is to distinguish cancer from normal tissue using microarray data with 2000 features per example. The data was derived from 22 normal and 40 cancer tissues. In Table 1 we report average performance over 100 random partitions into 50 training and 12 test examples. The above Bayesian algorithms and the SVMs have similar performance. For the minimal achieveable test errors, the SVM with Fisher score feature ranking gave an average 1.63 ± 0.10 errors (using four features), the RFE algorithm gave 1.76 ± 0.10 errors (with 77 features). For the Bayesian algorithms the lowest errors were 1.42 ± 0.128 for algorithm 1 (at $\sigma = 1.1$) and 2.13 ± 0.12 for algorithm 2 (at $\sigma = 2.2$).

To investigate the degree of uniqueness of the hypotheses we performed 100 random partitionings of the data into disjoint subsets of 31 examples each and trained the algorithm on each subset separately. For algorithm 1 the occurrence rate for particular features is shown in Figure 1. Several features were particularly prominant: feature

377 (accession no. Z50753) corresponding to *m*RNA for uroguanylin precursor, 493 (R87126) corresponding to myosin heavy chain, nonmuscle, and 1772 (H08393) corresponding to collagen alpha 2(XI) chain (*Homo sapiens*). Feature 377 was a feature in common for 27 out of 100 partitionings of the data and hence it is a *very* significant feature according to our earlier discussion (the lowest conditional probability found was 0.947 for finding *no* features in common for the given individual experiment). Feature 493 is found in common three times for these 100 partitionings, and features 1772 and 625 (X12671) are found in common once each. For algorithm 2 a similar distribution appeared with feature 377 appearing in common in eight of 100 partitions and feature 493 occurring on three occasions in common (for interest, using the likelihood (12) feature 377 appears on six occasions and feature 1772 on one occasion).

Guanylin and uroguanylin are markedly reduced in early colon tumours (Notterman *et al.*, 2001) with very low expression in adenocarcinoma of the colon and also in its benign precursor, the adenoma (D. Notterman, personal communication). Treatment with uroguanylin has recently been found to have possible therapeutic significance (Shailubhai *et al.*, 2000) with a significant reduction in the number of pre-cancerous colon polyps (adenomas), shrinkage in the remainder and observed apoptosis of adenocarcinoma cells. It is remarkable that therapeutically significant targets can be highlighted so clearly using these methods.

Ovarian cancer dataset

As a second experiment we also evaluated our algorithm on a new ovarian cancer dataset (for an earlier study see Schummer *et al.* (1999)). This consisted of 30 examples derived from ovarian tumours and 24 normal examples, each example having 1536 features. The results are presented in Table 2. Again the Bayesian algorithms and SVMs have similar performance. Implicitly using the test set, the lowest averaged test error performance was 1.08 ± 0.10 errors (using Fisher score ranking of features with 80 features), 1.70 ± 0.11 errors (RFE, using six features), 1.56 ± 0.11 errors (algorithm 1, at $\sigma = 1.1$) and 1.31 ± 0.11 errors (algorithm 2, at $\sigma = 1.9$).

Splitting the data into two disjoint datasets of 27 examples each we found the occurrence rate for features given in Figure 2 for algorithm 1. Five features were found in common for 100 partitionings of the data: feature 9 was in common on 34 occasions, feature 1526 on 20 occasions and features 1483, 510, 93 on one occasion each. For algorithm 2 we found 1526 in common on four occasions, 1483 in common on four occasions and feature 9 in common once (for the classification likelihood (12) feature 1491 appears on 16 occasions and 1526 on four occasions).

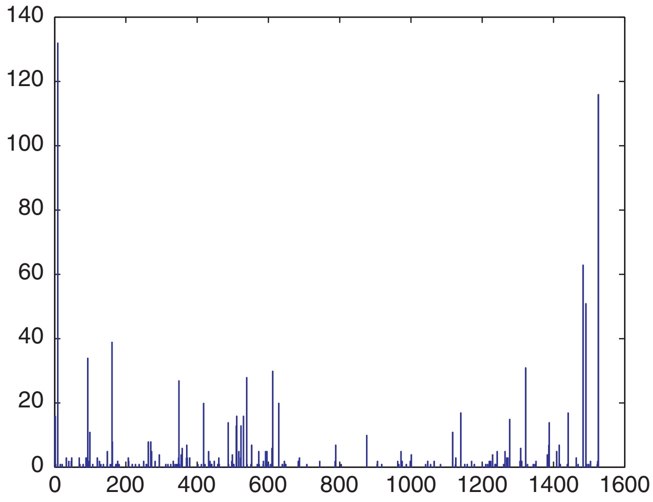

Fig. 2. For 100 partitionings of the ovarian cancer dataset the y-axis shows the number of occurrences of the feature (the maximum would be 200) and the x-axis shows the feature index.

Feature 1526 actually has no match to a known sequence. However, it has also been ranked high in alternative investigations of this dataset and therefore appears an interesting target for further investigation (M. Schummer, personal communication). Of the other features, feature 1491 (HE4) is a significant feature (Schummer, personal communication) and has been identified recently in serial analyses of gene expression data for ovarian cancer (Hough *et al.*, 2000). Several features are not so interesting as markers. For example, feature 9 corresponds to Dihydropyrimidase rel. prot-3 and it is not a good marker candidate because of its occurence in other tissues.

Leukaemia dataset

In our final study we considered distinguishing acute myeloid leukaemia (AML) from acute lymphoblastic leukaemia (ALL). This dataset (Golub *et al.*, 1999) has been extensively studied and we have chosen to use the standard split consisting of a training set of 38 examples (17 ALL and 11 AML) and a test set of 34 (20 ALL and 14 AML). The authors who gathered the data (Golub *et al.*, 1999) investigated the use of a weighted voting scheme. This correctly learnt 36 of the 38 training examples and on the test set it gave 29 from 34 correct, declining to predict on 5. They also tried a self-organizing map which gave two clusters: one with 24 ALL and 1 AML and the other with 10 AML and 3 ALL. Furey *et al.* (2000) investigated the use of a SVM with different settings of kernel parameter and soft margin. It correctly learnt all the training data and the test results varied according to the different configurations achieving zero training error. There were between four and two test errors except for

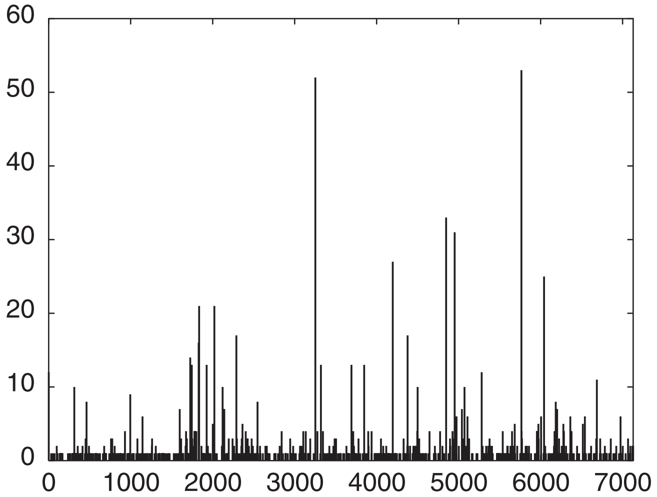

Fig. 3. For 100 partitionings of the leukemia cancer dataset the y-axis shows the number of occurrences of the feature (the maximum would be 200) and the x-axis shows the feature index.

one choice with 29 correct and the five declined by the weighted voting scheme classified incorrectly. For an SVM (Fisher score ranking of features and LOO stopping criterion) we found a set of eight features was used but there were six errors on the test set. For the RFE algorithm 65 features were used and there were three errors. For this particular dataset algorithm 1 gave three test errors and algorithm 2 gave one error. Making use of the test set we can obtain minimal test errors similar to those reported elsewhere (Proceedings of CAMDA, 2000) namely zero errors (SVM/Fisher score feature ranking), one error (SVM/RFE), two errors (algorithm 1) and one error (algorithm 2).

For 36 + 36 splits of the data, with 100 resamplings, the average number of features used was 6.98 ± 0.25 for algorithm 1 and 7.09 ± 0.10 for algorithm 2 (at $\sigma = 1$). However, in contrast to the colon and ovarian datasets, few features appeared in common at significant levels. For algorithm 1 features 3252 appeared twice in common, feature 1834 once in common and for algorithm 2 features 985, 1779 and 3252 appeared once and 4847 twice in common. From theorem (21) the lowest conditional probability of finding no features in common on random grounds was found to be 0.983 per experiment for algorithm 1, for example.

DISCUSSION

The two algorithms give similar performance and we have stated both variants since they can give differing emphasis to various features. Both algorithms have similar test error rates to SVMs. Though the datasets considered here were linearly separable it is also possible to use kernel substitution to handle non-linearly separable datasets (see Tipping (2001)).

Simpler methods (e.g. scores such as the Fisher score or Pearson correlation coefficient) can be used to highlight differential expression between genes belonging to samples from different classes. Compared to feature selection by the above algorithms, these scoring methods would typically give a much larger number of differentially expressed genes since, for discrimination, there is no need to use redundant features if the decision function can be accurately formulated using a few strong features. Of course, this smaller set of important discriminating features could be potentially interesting. Indeed, using the methods proposed here some of the genes highlighted certainly appear to be medically relevant, most notably the uroguanylin precursor for colon cancer and HE4 for ovarian cancer. In addition, scores can indicate the significance of individual genes but do not use mutual information between features, whereas the decision boundary created by a classifier has a more holistic dependence on the features.

However, we believe the main use of the above algorithms would be in tasks such as diagnosis and the assignment of new samples to particular categories. For example, for the leukaemia dataset the task was to assign the input to one of two classes. In addition, cDNA microarray experiments have indicated genetic sub-categories for a number of cancers (e.g. lymphoma (Alizadeh *et al.*, 2000) and lung cancer (Sorlie *et al.*, 2001)). In these cases, use of a smaller number of important features simplifies the experimental process. At the same time the proposed algorithms incorporate Bayesian principles to ensure good generalization for the classification of new examples.

ACKNOWLEDGMENTS

We would like to thank Dan Notterman (Princeton University) for comments on results for the colon cancer dataset and Michel Schummer (University of Washington, Seattle) for comments on the results and access to his group's ovarian cancer dataset.

REFERENCES

Alizadeh,A.A. *et al.* (2000) Different types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, **403**, 503–511.

Alon,U., Barkai,N., Notterman,D., Gish,K., Ybarra,S., Mack,D. and Levine,A. (1999) Broad patterns of gene expression revealed by clustering analysis of tumour and normal colon cancer tissues probed by oligonucleotide arrays. *Cell Biol.*, **96**, 6745–6750.

Bishop,C.M. (1995) *Neural Networks for Pattern Recognition*, Chapter 10, Oxford University Press.

Brown,M., Grundy,W., Lin,D., Cristianini,N., Sugnet,C., Furey,T., Ares,M. Jr. and Haussler,D. (2000) Knowledge-based analysis of microarray gene expression data using support vector machines. *Proc. Natl Acad. Sci. USA*, **97**, 262–267.

Faul,A. and Tipping,M. (2001) Analysis of sparse Bayesian learning. *Advances in Neural Information Processing Systems*, **15**, (to appear).

Furey,T., Cristianini,N., Duffy,N., Bednarski,D., Schummer,M. and Haussler,D. (2000) Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, **16**, 906–914.

Golub,T., Slonim,D., Tamayo,P., Huard,C., Gaasenbeek,M., Mesirov,J., Coller,H., Loh,M., Downing,J., Caligiuri,M. *et al.* (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531–537.

Guyon,I., Weston,J., Barnhill,S. and Vapnik,V. (2002) Gene selection for cancer classification using support vector machines. *Machine Learning*, **46**, 389–422.

Hough,C.D., Sherman-Baust,C.A., Pizer,E.S., Montz,F.J., Im,D.D., Rosenhein,N.B., Cho Riggins,G.J. and Morin,P.J. (2000) Large-scale serial analysis of gene expression in ovarian cancer. *Cancer Research*, **15**, 6281–6287.

Joachims,T. (2000) Estimating the generalization performance of an SVM efficiently. In *Proceedings of the Seventeenth International Conference on Machine Learning*. Morgan Kaufmann, pp. 431–438.

MacKay,D.J.C. (1994) Bayesian methods for back-propagation networks. In Domany,E., van Hemmen,J.L. and Schulten,K. (eds), *Models of Neural Networks III*, Chapter 6, Springer, New York.

Neal,R. (1994) *Bayesian Learning for Neural Networks*, PhD thesis, University of Toronto, Canada.

Notterman,D., Alon,U., Sierk,A. and Levine,A. (2001) Transcriptional gene expression profiles of colorectal adenoma, adenocarcinoma, and normal tissue examined by oligonucleotide arrays. *Cancer Res.*, **61**, 3124–3130.

Proceedings of CAMDA (December 2000) Critical Assessment of Microarray Data Analysis Techniques, Duke University.

Schummer,M., Ng,W., Bumgarner,R., Nelson,P., Schummer,B., Bednarski,D., Hassell,L., Baldwin,R., Karlan,B. and Hood,L. (1999) Comparative hybridization of an array of 21 500 ovarian cDNAs for the discovery of genes overexpressed in ovarian carcinomas. *Gene*, **238**, 375–385.

Shailubhai,K., Yu,H.H., Karunanandaa,K., Wang,J.Y., Eber,S.L., Wang,Y., Joo,N.S., Kim,H.D., Miedema,B.W. and Abbas,S.Z. *et al.* (2000) Uroguanylin treatment suppresses polyp formation in the Apc$^{Min/+}$ mouse and induces apoptosis in human colon adenocarcinoma cells via cyclic GMP. *Cancer Res.*, **60**, 5151–5163.

Sorlie,T. *et al.* (2001) Gene Expression Patterns of Breast Carcinomas distinguish Tumor Subclasses with Clinical Implications. *Proc. Natl Acad. Sci. USA*, **98**, 10869–10874.

Tipping,M.E. (2000) The relevance vector machine. *Advances in Neural Information Processing Systems*, **12**, 652–658.

Tipping,M.E. (2001) Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, **1**, 211–244.

Weston,J., Mukherjee,S., Chapelle,O., Pontil,M., Poggio,T. and Vapnik,V. (2001) Feature selection for SVMs. *Advances in Neural Information Processing Systems*, **13**, 668–674.