Biostatistics (2004), **5**, 3, *pp.* 329–340
doi: 10.1093/biostatistics/kxh010

Efficient quadratic regularization for expression arrays

TREVOR HASTIE[*], ROBERT TIBSHIRANI
Departments of Statistics, and Health Research & Policy, Stanford University, Sequoia Hall, CA 94305, USA
hastie@stanford.edu

SUMMARY

Gene expression arrays typically have 50 to 100 samples and 1000 to 20 000 variables (genes). There have been many attempts to adapt statistical models for regression and classification to these data, and in many cases these attempts have challenged the computational resources. In this article we expose a class of techniques based on quadratic regularization of linear models, including regularized (ridge) regression, logistic and multinomial regression, linear and mixture discriminant analysis, the Cox model and neural networks. For all of these models, we show that dramatic computational savings are possible over naive implementations, using standard transformations in numerical linear algebra.

Keywords: Eigengenes; Euclidean methods; Quadratic regularization; SVD.

1. INTRODUCTION

Suppose we have an expression array $\mathbf{X}$ consisting of n samples and p genes. In keeping with statistical practice the dimension of $\mathbf{X}$ is n rows by p columns; hence its transpose $\mathbf{X}^{\mathrm{T}}$ gives the traditional biologists' view of the vertical skinny matrix where the ith column is a microarray sample x_i. Expression arrays have orders of magnitude more genes than samples, hence $p \gg n$. We often have accompanying data that characterize the samples, such as cancer class, biological species, survival time, or other quantitative measurements. We will denote by y_i such a description for sample i. A common statistical task is to build a prediction model that uses the vector of expression values x for a sample as the input to predict the output value y.

In this article we discuss the use of standard statistical models in this context, such as the linear regression model, logistic regression and the Cox model, and linear discriminant analysis, to name a few. These models cannot be used 'out of the box', since the standard fitting algorithms all require $p < n$; in fact the usual rule of thumb is that there be five or ten times as many samples as variables. But here we consider situations with n around 50 or 100, while p typically varies between 1000 and 20 000.

There are several ways to overcome this dilemma. These include

- dramatically reducing the number of genes to bring down p; this can be done by univariate screening of the genes, using, for example, t-tests (Tusher *et al.*, 2001, e.g.);
- use of a constrained method for fitting the model, such as naive Bayes, that does not fit all p parameters freely (Tibshirani *et al.*, 2003);

[*] To whom correspondence should be addressed.

- use of a standard fitting method along with regularization.

In this article we focus on the third of these approaches, and in particular quadratic regularization, which has already been proposed a number of times in this context (Eilers *et al.*, 2001; Ghosh, 2003; West, 2003, for example). We show how all the computations, including cross-validation, can be simply and dramatically reduced for a large class of quadratically regularized linear models.

2. LINEAR REGRESSION AND QUADRATIC REGULARIZATION

Consider the usual linear regression model $y_i = x_i^{\mathrm{T}}\beta + \epsilon_i$ and its associated least-squares fitting criterion

$$\min_{\beta} \sum_{i=1}^{n} (y_i - x_i^{\mathrm{T}}\beta)^2. \tag{2.1}$$

The textbook solution $\hat{\beta} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$ does not work when $p > n$, since in this case the $p \times p$ matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ has rank at most n, and is hence singular and cannot be inverted. A more accurate description is that the 'normal equations' that lead to this expression, $\mathbf{X}^{\mathrm{T}}\mathbf{X}\beta = \mathbf{X}^{\mathrm{T}}\mathbf{y}$, do not have a unique solution for β, and infinitely many solutions are possible. Moreover, they all lead to a perfect fit; perfect on the training data, but unlikely to be of much use for future predictions.

The 'ridge regression' solution to this dilemma (Hoerl and Kennard, 1970) is to modify (2.1) by adding a quadratic penalty

$$\min_{\beta} \sum_{i=1}^{n} (y_i - x_i^{\mathrm{T}}\beta)^2 + \lambda\beta^{\mathrm{T}}\beta \tag{2.2}$$

for some $\lambda > 0$. This gives

$$\hat{\beta} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}, \tag{2.3}$$

and the problem has been fixed since now $\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}$ is invertible. The effect of this penalty is to constrain the size of the coefficients by shrinking them toward zero. More subtle effects are that coefficients of correlated variables (genes, of which there are many) are shrunk toward each other as well as toward zero.

Remarks:

- In (2.2) we have ignored the intercept for notational simplicity. Typically an intercept is included, and hence the model is $f(x) = \beta_0 + x^{\mathrm{T}}\beta$, but we do not penalize β_0 when doing the fitting. In this particular case we can rather work with centered variables (from each of the genes subtract its mean), which implies that the unpenalized estimate $\hat{\beta}_0$ is the mean of the y_i.
- Often in ridge regression, the predictor variables are measured in different units. To make the penalty meaningful, it is typically recommended that the variables be standardized first to have unit sample variance. In the case of expression arrays, the variables (genes) are all measured in the same units, so this standardization is optional.
- The tuning parameter λ controls the amount of shrinkage, and has to be selected by some external means. We demonstrate the use of K-fold cross-validation for this purpose in the examples later on.

It appears that the ridge solution (2.3) is very expensive to compute, since it requires the inversion of a $p \times p$ matrix (which takes $\mathrm{O}(p^3)$ operations). Here we demonstrate a computationally efficient solution to this problem.

Let

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}} \tag{2.4}$$

$$= \mathbf{R}\mathbf{V}^{\mathrm{T}} \tag{2.5}$$

be the singular-value decomposition (Golub and Van Loan, 1983, SVD) of **X**; that is, **V** is $p \times n$ with orthonormal columns, **U** is $n \times n$ orthogonal, and **D** a diagonal matrix with elements $d_1 \geqslant d_2 \geqslant d_n \geqslant 0$. Hence $\mathbf{R} = \mathbf{UD}$ is also $n \times n$, the matrix of so-called *eigengenes* (Alter *et al.*, 2000). Plugging this into (2.3), and after some careful linear algebra, we find that

$$\hat{\beta} = \mathbf{V}(\mathbf{R}^{\mathrm{T}}\mathbf{R} + \lambda\mathbf{I})^{-1}\mathbf{R}^{\mathrm{T}}\mathbf{y}. \tag{2.6}$$

Comparing with (2.3), we see that (2.6) is the ridge-regression coefficient using the much smaller $n \times n$ regression matrix **R**, pre-multiplied by **V**. In other words, we can solve the ridge-regression problem involving p variables, by

- reducing the p variables (genes) to $n \ll p$ variables (eigengenes) via the SVD in $\mathrm{O}(pn^2)$ operations;
- solving the n dimensional ridge regression problem in $\mathrm{O}(n^3)$ operations;
- transforming the solution back to to p dimensions in $\mathrm{O}(np)$ operations.

Thus the computational cost is reduced from $\mathrm{O}(p^3)$ to $\mathrm{O}(pn^2)$ when $p > n$. For our example in Section 4.4 this amounts to 0.4 seconds rather than eight days!

3. Linear predictors and quadratic penalties

There are many other models that involve the variables through a linear predictor. Examples include logistic and multinomial regression, linear and mixture discriminant analysis, the Cox model, linear support-vector machines, and neural networks. We discuss some of these in more detail later in the paper. All these models produce a function $f(x)$ that involves x via one or more linear functions. They are typically used in situations where $p < n$, and are fit by minimizing some loss function $\sum_{i=1}^{n} L(y_i, f(x_i))$ over the data. Here L can be squared error, negative log-likelihood, negative partial log-likelihood, etc. All suffer in a similar fashion when $p \gg n$, and all can be *fixed* by quadratic regularization:

$$\min_{\beta_0,\beta} \sum_{i=1}^{n} L(y_i, \beta_0 + x_i^{\mathrm{T}}\beta) + \lambda\beta^{\mathrm{T}}\beta. \tag{3.1}$$

For the case of more than one set of linear coefficients (multinomial regression, neural networks), we can simply add more quadratic penalty terms.

We now show that the SVD trick used for ridge regression can be used in *exactly the same way* for all these problems: replace the huge gene expression matrix **X** with p columns (variables or genes) by the much smaller matrix **R** with n columns (eigengenes), and fit the same model in the smaller space. All aspects of model evaluation, including cross-validation, can be performed in this reduced space.

3.1 *Reduced space computations*

THEOREM 1 Let $\mathbf{X} = \mathbf{R}\mathbf{V}^{\mathrm{T}}$ as in (2.5), and denote by r_i the ith row of $\mathbf{R}$, a vector of n predictor values for the ith observation. Consider the pair of optimization problems:

$$(\hat{\beta}_0, \hat{\beta}) = \underset{\beta_0, \beta \in \mathbb{R}^p}{\operatorname{argmin}} \sum_{i=1}^{n} L(y_i, \beta_0 + x_i^{\mathrm{T}}\beta) + \lambda\beta^{\mathrm{T}}\beta; \tag{3.2}$$

$$(\hat{\theta}_0, \hat{\theta}) = \underset{\theta_0, \theta \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{i=1}^{n} L(y_i, \theta_0 + r_i^{\mathrm{T}}\theta) + \lambda\theta^{\mathrm{T}}\theta. \tag{3.3}$$

Then $\hat{\beta}_0 = \hat{\theta}_0$, and $\hat{\beta} = \mathbf{V}\hat{\theta}$.

The theorem says that we can simply replace the p-vectors x_i by the n-vectors r_i, and perform our penalized fit as before, except with much fewer predictors. The n-vector solution $\hat{\theta}$ is then transformed back to the p-vector solution via a simple matrix multiplication.

Proof. Let $\mathbf{V}_\perp$ be $p \times (p - n)$ and span the complementary subspace in $\mathbb{R}^p$ to $\mathbf{V}$. Then $\mathbf{Q} = (\mathbf{V} : \mathbf{V}_\perp)$ is a $p \times p$ orthonormal matrix. Let $x_i^* = \mathbf{Q}^{\mathrm{T}} x_i$ and $\beta^* = \mathbf{Q}^{\mathrm{T}}\beta$. Then

- $x_i^{*\mathrm{T}}\beta^* = x_i^{\mathrm{T}}\mathbf{Q}\mathbf{Q}^{\mathrm{T}}\beta = x_i^{\mathrm{T}}\beta$, and
- $\beta^{*\mathrm{T}}\beta^* = \beta^{\mathrm{T}}\mathbf{Q}\mathbf{Q}^{\mathrm{T}}\beta = \beta^{\mathrm{T}}\beta$.

Hence the criterion (3.2) is equivariant under orthogonal transformations. There is a one–one mapping between the location of their minima, so we can focus on β^* rather than β. But from the definition of $\mathbf{V}$ in (2.5), $x_i^{*\mathrm{T}}\beta^* = r_i^{\mathrm{T}}\beta_1^*$, where β_1^* consists of the first n elements of β^*. Hence the loss part of the criterion (3.2) involves β_0 and β_1^*. We can similarly factor the quadratic penalty into two terms $\lambda\beta_1^{*\mathrm{T}}\beta_1^* + \lambda\beta_2^{*\mathrm{T}}\beta_2^*$, and write (3.2) as

$$\left[\sum_{i=1}^{n} L(y_i, \beta_0 + r_i^{\mathrm{T}}\beta_1^*) + \lambda\beta_1^{*\mathrm{T}}\beta_1^*\right] + \left[\lambda\beta_2^{*\mathrm{T}}\beta_2^*\right], \tag{3.4}$$

which we can minimize separately. The second part is minimized at $\beta_2^* = 0$, and the result follows by noting that the first part is identical to the criterion in (3.3) with $\theta_0 = \beta_0$ and $\theta = \beta_1^*$. From the equivariance,

$$\hat{\beta} = \mathbf{Q}\hat{\beta}^* = (\mathbf{V} : \mathbf{V}_\perp)\begin{pmatrix}\hat{\theta}\\ 0\end{pmatrix} = \mathbf{V}\hat{\theta} \tag{3.5}$$

□

3.2 *Eigengene weighting*

Although Theorem 1 appears to be only about computations, there is an interpretative aspect as well. The columns of $\mathbf{R} = \mathbf{U}\mathbf{D}$ are the principal components or eigengenes of $\mathbf{X}$ (if the columns of $\mathbf{X}$ are centered), and as such they have decreasing variances (proportional to the diagonal elements of $\mathbf{D}^2$). Hence the quadratic penalty in (3.3) favors the larger-variance eigengenes. We formalize this in terms of the *standardized* eigengenes, the columns of $\mathbf{U}$.

COROLLARY 2 Let u_i be the ith row of $\mathbf{U}$. The optimization problem

$$(\hat{\omega}_0, \hat{\omega}) = \underset{\omega_0, \omega \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{i=1}^{n} L(y_i, \omega_0 + u_i^{\mathrm{T}} \omega) + \lambda \sum_{j=1}^{n} \frac{\omega_j^2}{d_j^2} \tag{3.6}$$

is equivalent to (3.3).

This makes explicit the fact that the leading eigengenes are penalized less than the trailing ones. If λ is not too small, and some of the trailing d_j are very small, one could reduce the set of eigengenes even further to some number $m < n$ without affecting the results much.

3.3 *Cross-validation*

No matter what the loss function, the models in (3.2) are defined up to the regularization parameter λ. Often λ is selected by k-fold cross-validation. The training data are randomly divided into k groups of roughly equal size n/k. The model is fit to $\frac{k-1}{k}$ and tested on $\frac{1}{k}$ of the data, k separate times, and the results averaged. This is done for a series of values for λ (typically on the log scale), and a preferred value is chosen.

COROLLARY 3 The entire model-selection process via cross-validation can be performed using a single reduced data set $\mathbf{R}$. Hence, when we perform cross-validation, we simply sample from the rows of $\mathbf{R}$.

Proof. Cross-validation relies on predictions $x^{\mathrm{T}}\beta$, which are equivariant under orthogonal rotations. □

Although for each training problem of size $n\frac{k-1}{k}$, an even smaller version of $\mathbf{R}$ could be constructed, the computational benefit in model fitting would be far outweighed by the cost in constructing these k copies $\mathbf{R}_k$.

3.4 *Derivatives*

In many situations, such as when the loss function is based on a log-likelihood, we use the criterion itself and its derivatives as the basis for inference. Examples are profile likelihoods, score tests based on the first derivatives, and (asymptotic) variances of the parameter estimates based on the information matrix (second derivatives). We now see that we can obtain many of these p-dimensional functions from the corresponding n-dimensional versions.

COROLLARY 4 Define $L(\beta) = \sum_{i=1}^{n} L(y_i, \beta_0 + x_i^{\mathrm{T}}\beta)$, $L(\theta) = \sum_{i=1}^{n} L(y_i, \beta_0 + r_i^{\mathrm{T}}\theta)$. Then with $\theta = \mathbf{V}^{\mathrm{T}}\beta$,

$$L(\beta) = L(\theta). \tag{3.7}$$

If L is differentiable, then

$$\frac{\partial L(\beta)}{\partial \beta} = \mathbf{V}\frac{\partial L(\theta)}{\partial \theta}; \tag{3.8}$$

$$\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^{\mathrm{T}}} = \mathbf{V}\frac{\partial^2 L(\theta)}{\partial \theta \partial \theta^{\mathrm{T}}}\mathbf{V}^{\mathrm{T}}, \tag{3.9}$$

with the partial derivatives in the right-hand side evaluated at $\theta = \mathbf{V}^{\mathrm{T}}\beta$.

Notes:

- These equations hold at all values of the parameters, not just at the solutions.
- Obvious (simple) modifications apply if we include the penalty in these derivatives.

Proof. Equation (3.7) follows immediately from the identity $\mathbf{X} = \mathbf{R}\mathbf{V}^{\mathrm{T}}$, and the fact that x_i^{T} and r_i^{T} are the ith rows of $\mathbf{X}$ and $\mathbf{R}$. The derivatives (3.8) and (3.9) are simple applications of the chain rule to (3.7). □

The SVD is a standard linear algebra tool, and requires $\mathrm{O}(pn^2)$ computations with $p > n$. It amounts to a rotation of the observed data in R^p to a new coordinate system, in which the data have nonzero coordinates on only the first n dimensions. The Q-R decomposition (Golub and Van Loan, 1983) would do a similar job.

4. EXAMPLES OF REGULARIZED LINEAR MODELS

In this section we briefly document and comment on a large class of linear models where quadratic regularization can be used in a similar manner, and the same computational trick of using r_i rather than x_i can be used.

4.1 *Logistic regression*

Logistic regression is the traditional linear model used when the response variable is binary. The class conditional probability is represented by

$$\Pr(y = 1|x) = \frac{\mathrm{e}^{\beta_0 + x^{\mathrm{T}}\beta}}{1 + \mathrm{e}^{\beta_0 + x^{\mathrm{T}}\beta}}. \tag{4.1}$$

The parameters are typically fit by maximizing the binomial log-likelihood

$$\sum_{i=1}^{n} \{y_i \log p_i + (1 - y_i) \log(1 - p_i)\}, \tag{4.2}$$

where we have used the shorthand notation $p_i = \Pr(y = 1|x_i)$.

If $p > n - 1$, maximum-likelihood estimation fails for similar reasons as in linear regression, and several authors have proposed maximizing instead the penalized log-likelihood:

$$\sum_{i=1}^{n} y_i \log p_i + (1 - y_i) \log(1 - p_i) - \lambda\beta^{\mathrm{T}}\beta \tag{4.3}$$

(Ghosh, 2003; Eilers *et al.*, 2001; Zhu and Hastie, 2004).

Remarks:

- Sometimes for $p < n$, and generally always when $p \gg n$, the two classes can be separated by an affine boundary. Maximum likelihood estimates for logistic regression are undefined (parameters march off to infinity); the regularization fixes this, and provides a unique solution in either of the above cases.
- In the separable case above, as $\lambda \downarrow 0$, the sequence of solutions $\hat{\beta}(\lambda)$ (suitably normalized) converge to the optimal separating hyperplane; i.e. the same solution as the support-vector machine (Rosset *et al.*, 2003); see below.

Theorem 1 tells us that we can fit instead a regularized logistic regression using the vector of eigengenes r_i as observations, instead of the x_i. Although Eilers *et al.* (2001) use a similar computational device, they expose it only in terms of the specific ML score equations deriving from (4.3).

4.2 *Generalized linear models*

Linear regression by least squares fitting and logistic regression are part of the class of *generalized linear models*. For this class we assume the regression function $\mathrm{E}(y|x) = \mu(x)$, and that $\mu(x)$ is related to the inputs via the monotonic *link* function g: $g(\mu(x)) = f(x) = \beta_0 + x^{\mathrm{T}}\beta$. The log-linear model for responses y_i that are counts is another important member of this class. These would all be fit by regularized maximum likelihood if $p \gg n$.

4.3 *The Cox proportional hazards model*

This model is used when the response is survival time (possibly censored). The hazard function is modeled as $\lambda(t|x) = \lambda_0(t)\mathrm{e}^{x^{\mathrm{T}}\beta}$. Here there is no intercept, since it is absorbed into the baseline hazard $\lambda_0(t)$. A *partial likelihood* (Cox, 1972) is typically used for inference, regularized if $p \gg n$.

4.4 *Multiple logistic regression*

This model generalizes the logistic regression model when there are $K > 2$ classes. It has the form

$$Pr(y = j|x) = \frac{\mathrm{e}^{\beta_{0j}+\beta_j^{\mathrm{T}}x}}{\sum_{\ell=1}^{K}\mathrm{e}^{\beta_{0\ell}+\beta_\ell^{\mathrm{T}}x}}. \tag{4.4}$$

When $p > n$, this model would be fit by maximum penalized log-likelihood, based on the multinomial distribution

$$\max_{\{\beta_{0j},\beta_j\}_{j=1}^{K}} \sum_{i=1}^{n}\log \Pr(y_i|x_i) - \lambda\sum_{j=1}^{K}\beta_j^{\mathrm{T}}\beta_j. \tag{4.5}$$

There is some redundancy in the representation (4.4), since we can add a constant c_m to all the class coefficients for any variable x_m, and the probabilities do not change. Typically in logistic regression, this redundancy is overcome by arbitrarily setting the coefficients for one class to zero (typically class K). Here this is not necessary, because of the regularization penalty; the c_m are chosen automatically to minimize the L_2 norm of the set of coefficients. Since the constant terms β_{0j} are not penalized, this redundancy persists, but we still choose the minimum-norm solution. This model is discussed in more detail in Zhu and Hastie (2004).

Even though there are multiple coefficient vectors β_j, it is easy to see that we can once again fit the multinomial model using the reduced set of eigengenes r_i.

Figure 1 shows the results of fitting (4.4) to a large cancer expression data set (Ramaswamy *et al.*, 2001). There are 144 training tumor samples and 54 test tumor samples, spanning 14 common tumor classes that account for 80% of new cancer diagnoses in the U.S. There are 16 063 genes for each sample. Hence $p = 16\,063$ and $n = 144$, in our terminology.

The deviance plot (center panel) measures the fit of the model in terms of the fitted probabilities, and is smoother than misclassification error rates. We see that a good choice of λ is about 1 for these data; larger than that and the error rates (CV and test) start to increase.

These error rates might seem fairly high (0.27 or 15 misclassified test observations at best). For these data the null error rate is 0.89 (assign all test observations to the dominant class), which is indicative of the difficulty of multi-class classification. When this model is combined with redundant feature elimination (Zhu and Hastie, 2004), the test error rate drops to 0.18 (nine misclassifications).

The multinomial model not only learns the classification, but also provides estimates for the probabilities for each class. These can be used to assign a strength to the classifications. For example,

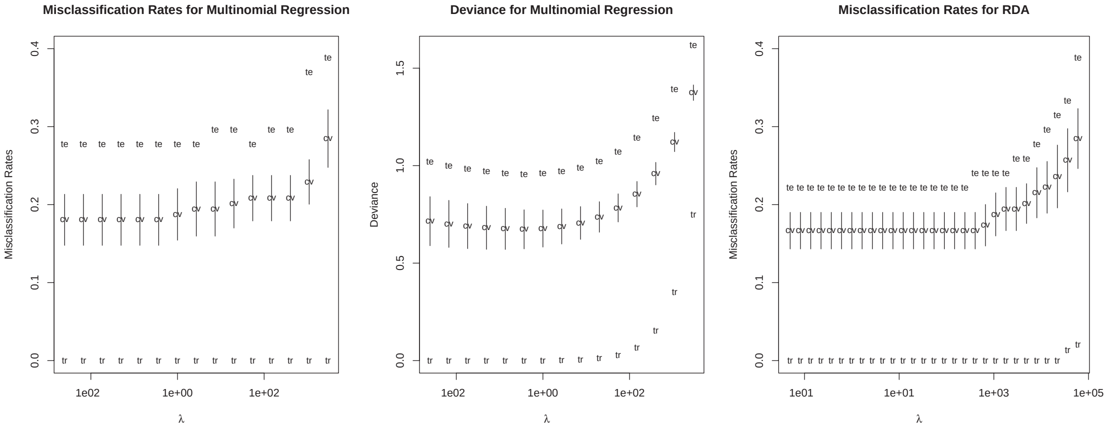

Fig. 1. Misclassification rates and deviance (2× negative log-likelihood) for the 14-class cancer data (left and middle panel). The labels indicate training data (tr), test data (te), and 8-fold cross-validation (cv). The minimum number of test errors was 15. The right panel shows the same for RDA (Section 4.5); the minimum number of test errors for RDA is 12.

one of the misclassified test observations had a probability estimate of 0.46 for the incorrect class, and 0.40 for the correct class; such a close call with 14 classes competing might well be assigned to the *unsure* category. For six of the 15 misclassified test observations, the true class had the second highest probability score.

4.5 *Regularized linear discriminant analysis*

The LDA model is based on an assumption that the input features have a multivariate Gaussian distribution in each of the classes, with different mean vectors μ_k, but a common covariance matrix Σ. It is then easy to show that the log posterior probability for class k is given (up to a factor independent of class) by the *discriminant function*

$$\delta_k(x) = x^{\mathrm{T}}\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^{\mathrm{T}}\Sigma^{-1}\mu_k + \log\pi_k, \tag{4.6}$$

where π_k is the *prior probability* or background relative frequency of class k. Note that $\delta_k(x)$ is linear in x. We then classify to the class with the largest $\delta_k(x)$. In practice, estimates

$$\hat{\pi}_k = \frac{n_k}{n}, \quad \hat{\mu}_k = \frac{1}{n_k}\sum_{y_i=k} x_i, \quad \hat{\Sigma} = \frac{1}{n-k}\sum_{k=1}^{K}\sum_{y_i=k}(x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^{\mathrm{T}} \tag{4.7}$$

are plugged into (4.6) giving the estimated discriminant functions $\hat{\delta}_k(x)$. However, $\hat{\Sigma}$ is $p \times p$ and has rank at most $n - K$, and so its inverse in (4.6) is undefined. *Regularized discriminant analysis* or RDA (Friedman, 1989; Hastie *et al.*, 2001) fixes this by replacing $\hat{\Sigma}$ with $\hat{\Sigma}(\lambda) = \hat{\Sigma} + \lambda\mathbf{I}$, which is nonsingular if $\lambda > 0$.

Now (4.6) and (4.7) do not appear to be covered by (3.1) and Theorem 1. In fact, one can view RDA estimates as an instance of penalized optimal scoring (Hastie *et al.*, 1995, 2001), for which there is an optimization problem of the form (3.1). However, it is simple to show directly that (4.6) and its regularized version are invariant under a coordinate rotation, and that appropriate terms can be dropped.

Hence we can once again use the SVD construction and replace the training x_i by their corresponding r_i, and fit the RDA model in the lower-dimensional space. Again the n-dimensional linear coefficients

$$\hat{\beta}_k^* = (\hat{\Sigma}^* + \lambda\mathbf{I})^{-1}\hat{\mu}_k^* \tag{4.8}$$

are mapped back to p-dimensions via $\hat{\beta}_k = \mathbf{V}\hat{\beta}_k^*$.

In this case further simplification is possible by diagonalizing $\hat{\Sigma}^*$ using the SVD. This allows one to efficiently compute the solutions for a series of values of λ without inverting matrices each time; see Guo *et al.* (2003) for more details.

RDA can also provide class probability estimates

$$\hat{\Pr}(y = k|x; \lambda) = \frac{e^{\delta_k(x;\lambda)}}{\sum_{j=1}^{K} e^{\delta_j(x;\lambda)}}. \tag{4.9}$$

From (4.9) it is clear that the models used by RDA and multinomial regression (4.4) are of the same form; they both have linear discriminant functions, but the method for estimating these differ. This issue is taken up in Hastie *et al.* (2001, Chapter 4). On these data RDA slightly outperformed multinomial regression (see Figure 1; 12 vs 15 test errors).

Regularized mixture discriminant analysis (Hastie and Tibshirani, 1996; Hastie *et al.*, 2001) extends RDA in a flexible way, allowing several centers per class. The same computational tricks work there as well.

4.6 *Neural networks*

Single layer neural networks have hidden units $z_m = \sigma(\beta_{0m} + \beta_m^{\mathrm{T}}x)$ that are linear functions of the inputs, and then another linear/logistic/multilogit model that takes the z_m as inputs. Here there are two layers of linear models, and both can benefit from regularization. Once again, quadratic penalties on the β_m allow us to re-parametrize the first layer in terms of the r_i rather than the x_i. The complicated neural-network analysis in Khan *et al.* (2001) could have been dramatically simplified using this device.

4.7 *Linear support vector machines*

The support vector machine (SVM) (Vapnik, 1996) for two-class classification is a popular method for classification. This model fits an optimal separating hyperplane between the data points in the two classes, with built-in slack variables and regularization to handle the case when the data cannot be linearly separated. The problem is usually posed as an application in convex optimization. With y_i coded as $\{-1, +1\}$, it can be shown (Wahba *et al.*, 2000; Hastie *et al.*, 2001) that the problem

$$\min_{\beta_0,\beta} \sum_{i=1}^{n} (y_i - \beta_0 - x_i^{\mathrm{T}}\beta)_+ + \lambda\beta^{\mathrm{T}}\beta \tag{4.10}$$

is an equivalent formulation of this optimization problem, and is of the form (3.1). In (4.10) we have used the *hinge loss* function for an SVM model, where the '$+$' denotes *positive part*.

Users of SVM technology will recognize that our computational device must amount to some version of the 'kernel' trick, which has been applied in many of the situations listed above. For linear models, the kernel trick amounts to a different re-parametrization of the data, also from p down to n dimensions. Since the solution to (4.10) can be shown to be of the form $\hat{\beta} = \mathbf{X}\hat{\alpha}$, the vector of fitted values (ignoring the intercept) is represented as

$$\hat{\mathbf{f}} = \mathbf{X}\mathbf{X}^{\mathrm{T}}\hat{\alpha} = \mathbf{K}\hat{\alpha}. \tag{4.11}$$

The *gram* matrix $\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$ represents the $n \times n$ inner-products between all pair input vectors in the data. The new input variables are the n kernel basis functions $K(x, x_i) = x^{\mathrm{T}} x_i,\ i = 1, \dots, n$.

From (4.11) it is clear that the parametrization recognizes that $\beta = \mathbf{X}^{\mathrm{T}}\alpha$ is in the row space of $\mathbf{X}$, just a different parametrization of our $\beta = \mathbf{V}\theta$. However, with the parametrization (4.11), the general criterion in (3.1) becomes

$$\min_{\beta_0,\alpha} \sum_{i=1}^{n} L(y_i, \beta_0 + k_i^{\mathrm{T}}\alpha) + \lambda\alpha^{\mathrm{T}}\mathbf{K}\alpha, \tag{4.12}$$

where k_i is the ith row of $\mathbf{K}$. Hence our re-parametrization r_i includes in addition an orthogonalization which diagonalizes the penalty in (4.12), leaving the problem in the same form as the original diagonal penalty problem.

The kernel trick allows for more flexible modeling, and is usually approached in the reverse order. A positive-definite kernel $K(x, x')$ generates a set of n basis functions $K(x, x_i)$, and hence a regression model $f(x) = \beta_0 + \sum_{i=1}^{n} K(x, x_i)\alpha_i$. A popular example of such a kernel is the radial basis function (Gaussian bump function)

$$K(x, x') = \mathrm{e}^{-\gamma||x-x'||^2}. \tag{4.13}$$

The optimization problem is exactly the same as in (4.12). What is often not appreciated is that the roughness penalty on this space is induced by the kernel as well, as is evidenced in (4.12). See Hastie *et al.* (2004) for more details.

4.8 *Euclidean distance methods*

A number of multivariate methods rely on the Euclidean distances between pairs of observations. K-means clustering and nearest-neighbor classification methods are two popular examples. It is easy to see that for such methods, we can also work with the r_i rather than the original x_i, since such methods are rotationally invariant.

- With K-means clustering, we would run the entire algorithm in the reduced space of eigengenes. The subclass means $\bar{r}_m$ could then be transformed back into the original space $\bar{x}_m = \mathbf{V}\bar{r}_m$. The cluster assignments are unchanged.
- With k-nearest-neighbor classification we would drop the query point x into the n-dimensional subspace, $r = \mathbf{V}^{\mathrm{T}}x$, and then classify according to the labels of the closest k r_i.

The same is true for hierarchical clustering, even when the correlation ‘distance’ is used.

5. DISCUSSION

There is one undesirable aspect to quadratically regularized linear models, for example, in the gene expression applications. The solutions $\hat{\beta}(\lambda)$ involve all the genes—no selection is done. An alternative is to use the so-called L_1 penalty $\lambda \sum_{j=1}^{p} |\beta_j|$ (Tibshirani, 1996), which causes many coefficients to be exactly zero. In fact, an L_1 penalty permits at most n nonzero coefficients (Efron *et al.*, 2002; Zhu *et al.*, 2003), which can be a problem if n is small. However, our computational trick to address the first issue only works with a quadratic penalty. Practice has shown that quadratically regularized models can still deliver good predictive performance. We have seen that SVMs are of this form, and they have become quite popular as classifiers. There have been several (ad hoc) approaches in the literature to select genes based on the size of their regularized coefficients (see Zhu and Hastie (2004) and references therein).

The models discussed here are not new; they have been in the statistics folklore for a long time, and many have already been used with expression arrays. The computational shortcuts possible with quadratically regularized linear models have also been discovered many times, often recently under the guise of ‘the kernel trick’ in the kernel learning literature (Schölkopf and Smola, 2001). Here we have shown that for all quadratically regularized models with linear predictors this device is totally transparent, and with a small amount of preprocessing all the models described here are computationally manageable with standard software.

References

Alter, O., Brown, P. and Botstein, D. (2000). Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences, USA* **97**, 10101–10106.

Cox, D. (1972). Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B* **74**, 187–220.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2002). Least angle regression. *Technical Report*. Stanford University.

Eilers, P., Boer, J., van Ommen, G. and Houwelingen, J. (2001). Classification of microarray data with penalized logistic regression. *Proceedings of SPIE Volume 4266, Progress in Biomedical Optics and Imaging,* **2**, 23, 187–198.

Friedman, J. (1989). Regularized discriminant analysis. *Journal of the American Statistical Association* **84**, 165–175.

Ghosh, D. (2003). Penalized discriminant methods for the classification of tumors from gene expression data. *Biometrics* **59**, 992–1000.

Golub, G. and Van Loan, C. (1983). *Matrix Computations*. Johns Hopkins University Press.

Guo, Y., Hastie, T. and Tibshirani, R. (2003). Regularized discriminant analysis and its application to microarrays. *Technical Report*. Statistics Department, Stanford University.

Hastie, T. and Tibshirani, R. (1996). Discriminant analysis by gaussian mixtures. *Journal of the Royal Statistical Society, Series B* **58**, 155–176.

Hastie, T., Buja, A. and Tibshirani, R. (1995). Penalized discriminant analysis. *Annals of Statistics* **23**, 73–102.

Hastie, T., Rosset, S., Tibshirani, R. and Zhu, J. (2004). The entire regularization path for the support vector machine. *Technical Report*. Statistics Department, Stanford University.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning; Data Mining, Inference and Prediction*. New York: Springer.

Hoerl, A. E. and Kennard, R. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **12**, 55–67.

Khan, J., Wei, J., Ringner, M., Saal, L., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C., Peterson, C. and Meltzer, P. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine* **7**, 673–679.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J. *et al.*, (2001). Multiclass cancer diagnosis using tumor gene expression signature. *Proceedings of the National Academy of Sciences, USA* **98**, 15149–15154.

Rosset, S., Zhu, J. and Hastie, T. (2003). Margin maximizing loss functions. *Neural Information Processing Systems*, online.

SCHÖLKOPF, B. AND SMOLA, A. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning)*. Cambridge, MA: MIT Press.

TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B* **58**, 267–288.

TIBSHIRANI, R., HASTIE, T., NARASIMHAN, B. AND CHU, G. (2003). Class prediction by nearest shrunken centroids, with applications to DNA microarrays. *Statistical Science* **18**, 104–117.

TUSHER, V., TIBSHIRANI, R. AND CHU, G. (2001). Significance analysis of microarrays applied to transcriptional responses to ionizing radiation. *Proceedings of the National Academy of Sciences, USA* **98**, 5116–5121.

VAPNIK, V. (1996). *The Nature of Statistical Learning*. Berlin: Springer.

WAHBA, G., LIN, Y. AND ZHANG, H. (2000). Gacv for support vector machines. In Smola, A., Bartlett, P., Schölkopf, B. and Schuurmans, D. (eds), *Advances in Large Margin Classifiers*, Cambridge, MA: MIT Press, pp. 297–311.

WEST, M. (2003). Bayesian factor regression models in the 'large p, small n' paradigm. *Bayesian Statistics* **7**, 723–732.

ZHU, J. AND HASTIE, T. (2004). Classification of gene microarrays by penalized logistic regression. *Biostatistics* **5**, 427–443.

ZHU, J., ROSSET, S., HASTIE, T. AND TIBSHIRANI, R. (2003). L1 norm support vector machines. *Technical Report*. Stanford University.

[*Received November 21, 2003; revised March 19, 2004; accepted for publication April 1, 2004*]