Available online at www.isas.org.in/jisas

JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 68(2) 2014 145-162

Generic Feature Selection with Short Fat Data

B. Clarke[1] and J.-H. Chu[2]

[1]*Department of Statistics, University of Nebraska-Lincoln, Lincoln, NE, 68583, USA*
[2]*Brigham and Women's Hospital, Harvard Medical School, Boston, MA 02115*

Received 08 October 2013; Revised 24 December 2013; Accepted 25 December 2013

SUMMARY

Consider a regression problem in which there are many more explanatory variables than data points, *i.e*., $p >> n$. Essentially, without reducing the number of variables inference is impossible. So, we group the p explanatory variables into blocks by clustering, evaluate statistics on the blocks and then regress the response on these statistics under a penalized error criterion to obtain estimates of the regression coefficients. We examine the performance of this approach for a variety of choices of n, p, classes of statistics, clustering algorithms, penalty terms, and data types. When n is not large, the discrimination over number of statistics is weak, but computations suggest regressing on approximately $[n/K]$ statistics where K is the number of blocks formed by a clustering algorithm. Small deviations from this are observed when the blocks of variables are of very different sizes. Larger deviations are observed when the penalty term is an L^q norm with high enough q.

Keywords: Large p small n, LASSO, Ridge, Bridge, Clustering, Variance-bias tradeoff, Summary statistics.

1. THE BASIC PROBLEM IN THE USUAL SETTINGS

Suppose $Y = Y^n = (Y_1, ..., Y_n)'$ is an $n \times 1$ data vector, $X = (X_1, ..., X_n)'$ is an $n \times p$ design matrix in which each X_i is a vector of p explanatory variables, and $\beta = (\beta_1, ..., \beta_p)'$ is the parameter vector. Suppose all the variables are standardized *i.e*., transformed to have mean zero and variance one so that it will be enough to look at the dependence structure and relative contributions of the X_i's. Let us write the model

$$Y = X\beta + \in \tag{1}$$

in which $\in = (\in_1, ..., \in_n)'$ is the error term and the constant term usually appearing in a regression model has been subsumed by the rescaling. We want $E(\in) = 0$, and $\mathrm{Var}(\in)$ to be diagonal. Regardless of the distribution of $\in$, we have

$$\hat{\beta}_{OLS} = \arg\min_{\beta} \sum_i (y_i - x_i'\beta)^2 = (X'X)^{-1}X'y, \tag{2}$$

as the least squares estimator of β, provided the inverse exists. If $|X'X|$ is small, the inverse is large in the sense that some of its eigenvalues must be large. When $p > n$, X is $n \times p$, *i.e*., short and fat. For Short Fat Data (SFD) $|X'X| = 0$ so its inverse fails to exist.

The central issue here is that the mean function for Y, EY, is in a space of dimension p while only $n < p$ data points are available. That is, the SFD or 'large p, small n' problem would disappear if we had more data. However, even though one can imagine arbitrarily large n's, in practice they do not exist.

Alternatively, we can try to do effective dimension reduction by regressing Y on functions of the X_i's. The idea is that if we evaluate a comparatively small number of suitably chosen functions on each X_i, *i.e*., features, and then do penalized regression on those features we will have retained all the information in the data about the response Y. The question is what kind of statistics

Corresponding author: B. Clarke
E-mail address : bclarke3@unl.edu

to use to achieve optimal dimension reduction. Obviously, good statistics on which to regress should encapsulate the information in the explanatory variables relevant to the response.

In many cases, this is done by careful physical modeling, *i.e*., using domain specific knowlege to restrict the class of models that have to be considered. Recent examples include Stenning *et al*. (2013) for solar image data and McKay (2004) for musical scores. However, feature selection based on modeling is very time-intensive and may require information not available to a researcher. So, here, we address *generic* feature selection, done in the absence of modeling information. We look at five classes of features, but only three are independent of the response and could therefore be used in practice. The other two are for comparison purposes to assess how well the first three seem to perform.

One can readily imagine that when p is large enough relative to n, dimension reduction to a 'reasonable' p' statistics may not give $p' \leq n$. This may be the case when the explanatory variables are known to segregate into a number of disjoint classes and the number of these classes is still greater than n. In these cases, it may be reasonable to use a single statistic within each class, but not to permit statistics to depend on variables from more than one class. Thus, even after reducing to regression on statistics one still has SFD. Not as fat as before, but the new '$X'X$' remains singular.

A second way to correct for $p >> n$ is to change the optimality criterion. Since $X'X$ appears in the solution under squared error, let us add a penalty term to shrink the solution towards non-singularity. One general class is

$$\arg\min_{f\in\mathcal{F}} \sum_{i=1}^{n} L_1(y_i, x_i, f) + \lambda L_2(f), \tag{3}$$

where $\mathcal{F}$ is a class of functions, and L_1 and L_2 are two loss functions. The first, L_1, expresses the sense in which we want the function $f(x)$ to be close to the response y. The second, L_2, ensures that the 'complexity' of f is not so large that we overfit the data. Since we don't want L_2 to swamp the information in the data we use a hyperparameter λ to control the tradeoff between how well f summarizes the data and how complicated f may be. Usually, λ is chosen adaptively and sometimes it is called a decay parameter.

Various instances of (3) are of great interest. The polynomial subclass is

$$\arg\min_{\beta} \sum_{i=1}^{n} (y_i - x_i'\beta)^q + \lambda \sum_{i=1}^{p} |\beta_i|^r \tag{4}$$

where q and r usually are integers. When $(\lambda, q) = (0, 2)$, (4) corresponds to (1) and yields (2). If the x_i's are replaced by functions of the explanatory variables, then we are doing 'feature selection' *i.e*., regression on statistics formed from the explanatory variables. Indeed, an estimator arising from (4) is Bayesian: The argument of the min in (4) can be regarded as the log-likelihood of the product of an n-fold normal density with mean vector $(x'_1\beta, \ldots, x'_n\beta)$ and a prior proportional to $e^{\lambda\Sigma_{i=1}^{p}|\beta_i|^r}$ so that finding the quantity in (4) is equivalent to choosing the mode of a posterior.

The main contribution of this paper is to observe that clustering over variables, summarizing the clusters by statistics, and then feeding the statistics into a shrinkage method may be an effective way to do dimension reduction. More formally, if one must reduce the number of explanatory variables by constructing features in the absence of sufficient modeling information (as is often the case) then one may be led to a two step procedure. The first step is to choose a number of clusters, K, by use of a clustering procedure, or by use of physical modeling when this is possible. In either case, the second step is to summarize each cluster separately by a small number of generic statistics. If one does this then the best number of statistics to use per cluster is roughly $[n/K]$, the smallest integer larger than n/K. Since $n + K \geq K[n/K] \geq n$, the total number of statistics roughly equals the number of data points. These statistics can then be fed into any penalized method such as (3) or (4) to give coefficients and predictors. Essentially this means that even when $p >> n$ one can pragmatically reduce to a $p' \approx n$ meaning that elaborate schemes for permitting $p' > n$ provide little gain.

In this procedure the variability due to the clustering is neglected in practice, although it is built into our simulations which use repeated calculations of the generalized cross-validation error over repeated generation of the data. However, the focus is not only on variability, bias matters too. In essence, the optimal number of statistics $[n/K]$ represents a bias-variance tradeoff: More statistics means less bias but more

variability, fewer statistics means more bias but less variability. Thus, our '[n/K] rule' stems from seeking an optimal variance-bias tradeoff rather than asymptotic optimality because n is small and it is unrealistic to think n will increase without bound.

In the first part of the next section we list several standard families of models and verify that they are of the form of (3) or (4). Equipped with these examples, we discuss the relationship of SFD and generic feature selection. In Section 3, we describe our contribution: It amounts to an investigation of how four modeling factors (penalty, choice of statistics, data type and clustering algorithm) affect regression on generic statistics with SFD. In Section 4, we present the results of this 4 by 4 array that suggest the '[n/K] rule'. In Section 5, we show two limitations of the [n/K] rule and, in Section 6, we give our general conclusions.

2. MODEL CLASSES AND SUMMARY STATISTICS

The four classes we briefly review here are OLS, Ridge, Bridge, and LASSO. There are many other model classes that use penalization such as CART (Breiman *et al*. 1984), SCAD (Fan and Li 2001), elastic net (Zou and Hastie 2005) and so forth. These are usually designed for model identification (CART is the exception) and often have the oracle property to ensure asymptotically good model identification. However, penalized methods with the oracle property do not in general perform as well as other methods do for data summarization and prediction which are the goals here. (SCAD is the one exception to this because its penalty is so small it compares with, say, model averaging methods.) Predictive comparisons among these approaches is not common, but see Austin *et al*. (2013) and Clarke and Severinski (2011) for special cases. At the end of this section we discuss the summary statistics used in our computations.

2.1 Ordinary Least Squares

Recall the ordinary least squares regression problem defined by (1.1) and (1.2). We obtain $\hat{\beta}_{\mathbf{OLS}}$ by minimizing the residual sum of squares $\Sigma_{i=1}^{n}(y_i - x_i^T\beta)^2$ over β. The estimator $\hat{\beta}_{\mathbf{OLS}}$ is unbiased for β but has a large variance when X is nearly collinear. Also, $\hat{\beta}_{\mathbf{OLS}}$ is not unique when X is less than full rank.

With SFD where $n < p$ we need to replace actual inverses with generalized inverses of some sort to get uniqueness. For an $n \times m$ matrix A the procedure begins by trying to solve $AX = y$ when $y \in Range(A)$. One definition of a generalized inverse for A is a matrix B for which $ABA = A$. This reduces to the usual definition of matrix inverse when A is invertible. If $BAB = B$, *i.e*., A is a generalized inverse for B, and both AB and BA are orthogonal projections, then B is unique. This is called the Moore-Penrose generalized inverse.

Using the Moore-Penrose inverse gives unique solutions. Indeed, the central results in the theory of linear models – properties of parameter estimates and fitted values, Chi-squared distributions for sums of squares – continue to hold using Moore-Penrose inverses. The cost is substantially inflated variances.

2.2 Ridge Regression

Hoerl and Kennard (1970) introduced ridge regression, RR, which modifies OLS by introducing a penalty term λ to shrink the β's toward zero. The RR estimator,

$$\hat{\beta}_{\mathbf{Ridge}} = \arg\min_{\beta}\left\{\sum_{i=1}^{n}|\, y_i - x_i^T\beta\,|^2 + \lambda\sum_{j}|\,\beta_j\,|^2\right\}$$

$$= (X^TX + \lambda I_p)^{-1}X^Ty,$$

is biased, but the variance is smaller than that of the OLS estimator. Therefore, one can often achieve better estimation in terms of MSE, and better prediction. It is seen that RR adds λ times the identity matrix to the objective function to force non-singularity.

2.3 LASSO

Tibshirani (1996) introduced the LASSO – Least Absolute Shrinkage and Selection Operator – which uses a factor λ times the absolute value of β as a penalty term. The LASSO estimator is defined to be

$$\hat{\beta}_{\mathbf{LASSO}} = \arg\min_{\beta}\left\{\sum_{i=1}^{n}|\, y_i - x_i^T\beta\,|^2 + \lambda\sum_{j}|\,\beta_j\,|\right\}.$$

The LASSO emerges from a more general treatment called Least Angle Regression, when an extra correlation restriction is enforced on the algorithm, see Efron *et al*. (2004, Exp. 3.1).

The LASSO combines shrinkage and selection on the regression function. The penalty term itself is often recognized as corresponding to putting a prior on β and 'shrinking' the parameter to a point, usually taken to be zero. The selection arises because there are many cases where optimizing in the LASSO leads to setting some of the β_j's to be zero.

2.4 Bridge Regression

Frank and Friedman (1993) defined bridge regression. It is (1.6) with $q = 2$. That is, the penalty on the sum of squares is λ times $\sum|\beta_j|^r$ for some $r \geq 0$. We write

$$\hat{\beta}_{\mathbf{Bridge}} = \arg\min_{\beta}\left\{\sum_{i=1}^{n}|y_i - x_i^T\beta|^2 + \lambda\sum_j|\beta_j|^r\right\}$$

It is seen that RR is $r = 2$, LASSO is $r = 1$, and AIC is nearly equivalent to "$r = 0$" in the normal error case.

2.5 Choice of Summary Statistics

One topic not usually addressed is how to choose functions of the explanatory variables generically so as to achieve more parsimonious models. This is sometimes done in the context of basis expansions *e.g.*, wavelets, but comparisons from one basis to another are infrequent. Usually, the basis is chosen on the grounds of some sort of physical modeling that may or may not be helpful. Indeed, choosing functions based on modeling assumptions is well-established but in cases where model information is scant, unreliable, or subjective one has little choice but to proceed generically. For instance, in SFD problems, there may not be extra information available on the X_i's to narrow the class of statistics it is worth searching but one can reduce the number of explanatory variables by requiring the sample variance of a component of X_i is large enough to be provide meaningful discrimination.

Here, we suppose the individual variables in the X_i's segregate into M classes C_1, C_2, ..., C_M. Let class C_k have p_k variables and let M_k be the $n \times p_k$ matrix of predictors in class C_k. Write M_{ik} for the i-th row of the submatrix M_k, and let $S(M_{ik})$ denote a function of the i-th row of the k-th class of variables in X_i. Our task is to choose functions of the form S in sensible ways to serve as summary statistics of $X_{i,p_{k-1}+1},\ldots,X_{i,p_{k-1}+p_k}$ on which to regress.

If we choose a single S_k for each C_k the regularized risk from (4) is

$$\sum_{i=1}^{n}\left(y_i - \sum_{k=1}^{M}\gamma_k S_k(M_{ik})\right)^q + \lambda\sum_{k=1}^{M}|\gamma_k|^r.$$

Using the classes C_k permits the p_k's to be reduced to a smaller number of statistics.

There are many natural choices for sequences of statistics to study. Percentiles and moments are the obvious ones to use first. Principal components, PC's, provides another way to choose a sequence of statistics generically. Alternatively, as we discuss below, statistics such as partial least squares, PLS's, or sliced inverse regression, SIR's, can be used. These last two are qualitatively different from moments, percentiles and PC's because of their dependence on the outcomes $y_1, \ldots, y_n$.

In some cases it is realistic to assume classes C_k are known. Hawkins *et al*. (2001) have a setting in which the classes can be specified pre-experimentally. However, in general, it is unclear how many statistics one wants to choose for each class C_k whence our '[n/K] rule'.

3. THE SIMULATION SETTING

The justification of the '[n/K] rule' rests on the computational investigation of a large matrix of cases representing the predictive performance of commonly occurring regularized risks. The problem can be visualized as a one way table crossed with another one way table crossed with a two way table. The first one way table is what the researcher cannot control: The actual properties of the data. The second one way table represents the pre-processing the researcher must do: This is how the explanatory variables are clustered into classes. The two-way table represents how the researcher models: The optimality criterion and the choice of statistics to summarize the data. We go through these factors in turn.

3.1 Data Type

The factor 'data type' is not under the control of the researcher. So, we used a large variety of standard data types to see how each technique performed on it. First we considered independent normal data with equal sized blocks. Then we used unequal sized blocks. Then we used correlated normal data with equal and unequal

block sizes. More generally, we turned to ARMA(a, b) data with a, b = 0, 1, 2. For greater realism, we then used non-normal independently generated data. The non-normality was mostly from the heavier tails although the shape of the distributions we used was not always symmetric. Finally, to investigate non-normal dependent data we generated correlated normal data but applied transformations to it so the distribution of the data going into the analysis would no longer be normal.

3.2 Clustering Algorithm

The factor 'clustering method', reflects how the researcher must preprocess the data so it will be amenable to summarization. We chose 6 levels *i.e*., 6 kinds of clustering, to partition the explanatory variables into disjoint classes C_1, ..., C_K; the elements within each C_i are expected to be more highly dependent than elements from different C_j's. For the first 'level' we assume the clusters to be known. The other five levels were K-means, three agglomerative procedures differing in the dissimilarity used, and one divisive.

The K-means algorithm is based on a distance between explanatory p-vectors X_i, here taken to be the Euclidean metric denoted by $\|\cdot\|$. We did this for a range done of K; the clustering with the smallest error is used as 'true'. We did this with the R function **kmeans**(), taking the clustering with the minimum error over 10 tries as the globally optimal clustering.

Three of the hierarchical methods were agglomerative. These clustering algorithms use dissimilarities d_j. Dissimilarities generalize the concept of distance: For entries of vectors we have values $d_j(x_{i,j}, x_{i',j})$ giving $d_{i,i'} = D(x_i, x_{i'}) = \Sigma_{j=1}^{p} d_j(x_{i,j}, x_{i',j})$ on vectors. Agglomerative hierarchical algorithms begin with n singleton clusters and combine two clusters at each step depending on the dissimilarity. The distance between clusters is expressed in terms of dissimilarity and here we use three forms: Single linkage uses $d_{NN}(G, H) = \min_{i\in G, i'\in H} d_{i,i'}$ for clusters G, and H. Complete linkage, or furthest neighbor, is $d_{FN}(G, H) = \max_{i\in G, i'\in H} d_{i,i'}$. Group average $d_{GA}(G, H)$ takes the mean of the $d_{i,i'}$'s over clusters G and H. These were implemented by **agnes**() in R, see Struyf *et al*. (1996). Of these three methods, complete linkage seemed to work better than single or group for our purpose, although the differences are small.

The sixth hierarchical clustering method was divisive. This approach begins by treating the whole data set as a single cluster and recursively divides it at each iteration. This procedure was implemented by **diana**() in R. For details, see Struyf *et al*. (1996).

3.3 The Optimality Criterion

The researcher gets to choose the optimality criterion to be employed. Here we consider 3 levels for this factor, *i.e*., 3 different forms of regularized risk. These are RR, LASSO, and bridge. RR is computed in closed form because the choice of λ gives non-singularity. That is, we can use fits directly from $(X'X + \lambda I_p)^{-1}X'_y$.

The version of LASSO we use here is a modified Least Angle Regression, LARS, see Efron *et al.* (2004). As described in Section 2, LASSO is a quadratic programming problem. However, using LARS one can obtain solutions readily for all values of λ by a variant of forward stepwise regression. As λ varies from 0 to ∞, the LARS procedure can be used to generate the LASSO solutions.

To implement the third level, bridge regression as in Section 2.1.4, we used the Fortran code and description of Fu (1998). Following this treatment, for given $\lambda \geq 0$ and $r \geq 1$ we compute $\hat{\beta}$ and use

$$p(\lambda) = tr(X(X'X + \lambda W^{-})^{-1}X') - n_0$$

as the effective number of parameters, in which W^{-} is the generalized inverse of $W = \mathrm{diag}(2|\hat{\beta}|^{2-r}/r)$ and n_0 is the number of entries $\hat{\beta}_j$ for which $\hat{\beta}_j = 0$ for $r = 1$. Note that all generalized inverse procedures give the same results on diagonal matrices and that n_0 represents the number of zero entries on the diagonal of W. It is seen that $\hat{\beta}$ solves

$$\left(X'X + \frac{\lambda r}{2}\mathrm{diag}(|\beta_j|^{r-2}) \right)\beta = X'y. \qquad (5)$$

3.4 Choice of Statistics

The second factor under the researcher's control is the choice of statistics. We used 5 levels. These levels consist of 2 subclasses with 3 and 2 levels respectively. The first class consisted of modified moments, percentiles, and PC's. The second class consisted of partial least squares (PLS) and sliced inverse regressions (SIR) statistics. Unlike the first subclass, these depend on the values of Y for their construction.

Our statistics based on moments separate positive and negative parts for odd exponents; this is not needed for even exponents. Thus, for a data vector $X = (X_1, ..., X_p)$ the first moment is represented by two statistics which we regress on together: $\bar{X}^+ = \Sigma_{i=1}^p X_i I_{\{X_i \geq 0\}}$ and $\bar{X}^- = \Sigma_{i=1}^p X_i I_{\{X_i < 0\}}$. The second moment is $\Sigma_{i=1}^p X_i^2$. The third is again separated into two positive and negative parts, like the first moment.

The percentiles we use are standard. The first percentile statistic is the median of $(X_1, ..., X_p)$, calculated with linear interpolation. The second set of percentile statistics consists of 3 statistics, the median and the two quartiles. The third set of percentile statis tics adds the 4 percentiles midway between all of the quartiles giving 7 percentiles at $12.5k$ for $k = 1, ..., 7$, and so forth. In jumping from one to three to seven percentiles, and so forth, the idea was to explore whether tail behavior was helpful by forcing the statistics to respond to different regions of the distribution. However, the 33rd and 67th percentiles, the quartiles, the quintiles, and so forth could have used instead, possibly leading to clearer support for the $[n/K]$ rule at the cost of them not being nested.

The PC's of a matrix X arise from writing $X = UDV'$ so that

$$X'X = V D^2 V',$$

where D is diag$(d_1, ..., d_p)$ with $d_j \geq 0$ in decreasing order. This is the usual diagonalization for a symmetric matrix giving non-negative real eigenvalues. Write $V = (V_1, ..., V_p)$. Then, $Z_j = XV_j$ for $j = 1, ..., p$ is a set of directions that can be assumed orthogonal with $Var(Z_j) = d_j^2 / n$. So, the first PC is Z_1 and it is the linear combination of explanatory variables having the largest variance. Likewise, the second PC is Z_2 and has the second largest variance, and so forth. One can regress on the the first Z_j's as a way to ensure the most important contributions to the variability in the data have been modeled. Regression on all the PC's devolves to the original regression.

Partial least squares, PLS, is a different way to construct a sequence of statistics on which to regress. Recall that Y and the X_j's are standardized. Begin by regressing Y on each of the p explanatory variables. This gives p expressions say $\hat{\phi}_{1,j} = \langle x_j, y \rangle x_j$ for $j = 1, ..., p$. The first PLS direction is $Z_1 = \Sigma_{j=1}^p \hat{\phi}_{1,j}$. Next, regress Y on Z_1 to get, say, $\hat{\psi}_1$ and orthogonalize the p explanatory variables with respect to Z_1, *i.e.*, subtract the portion of each explanatory variable X_j that is in the direction of Z_1. Redo the procedure for the orthogonalized explanatory variables, $X_j^{(new)} = X_j^{(old)} - \left[\langle Z_1, X_j^{(old)} \rangle / \langle Z_1, Z_1 \rangle \right] Z_1$ For all j to generate $\hat{\phi}_{u,j}$ and $\hat{\psi}_u$ for $u = 2, 3, ...$ to obtain Z_2..., see Hastie *et al*. (2001) Section 3.4 for further details. Regression on all the PLS directions devolves to the original regression.

Sliced Inverse Regression, SIR, is a technique from Li (1991) motivated by partitioning the range of Y, doing inverse regression on each region, pooling over the results and doing a pincipal components analysis on the weightd covariance matrix. The resulting statistics can be used for regressions.

3.5 Assessing Performance

In general, in these settings, we are concerned primarily with prediction since the true models are inaccessible. This leads us to use cross-validation, CV, as a performance criterion. The natural choice is leave-one-out CV because it is approximately unbiased for prediction error. However, the variance of leave-one-out CV may be high since any two of the training sets have $n - 2$ data points in common. Rather than using fivefold or other forms of CV, we actually used Generalized CV, GCV, for its computational convenience.

Suppose there exists a matrix S so that the fitted values $\hat{y} = (\hat{y}_1, ..., \hat{y}_n)$ for the outcome y can be expressed as $\hat{y} = Sy$. Then, writing $tr(S)$ for the trace of S,

$$GCV = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{y_i - \hat{f}(x_i)}{1 - tr(S)/n} \right)^2, \qquad (6)$$

which can be computed for ridge. GCV is easier to compute than CV because $tr(S)$ needs to be found only once. When a regression method is not linear, *i.e*., there is no matrix S, a GCV can still be given. For LASSO, the form of the GCV is given by Fu (1998). For bridge, consistent with (5) the GCV error is taken as

$$GCV = \frac{RSS}{n(1 - p(\lambda)/n)^2} \qquad (7)$$

in which $RSS = \Sigma_{i=1}^n (y_i - X_i' \hat{\beta})^2$. For our work below we chose $r = 1.5$, 2.5 and 3 and we let λ vary over (.001, 1000).

For the two different classes of statistics – dependent on Y or not – we used two different forms of the GCV. For the first, we used all the data to generate the statistics and then used leave-one-out GCV to find the best number of statistics per cluster. Thus, results for moments, percentiles, and PC's are an assessment of goodness of fit, averaged over 200 iterations. For the second class – PLS and SIR – we left out one data point, found the statistics, and then did GCV, averaging over 200 iterations. So, this criterion is more predictive and a 'purer' form of GCV than for the first class. This seemed appropriate since goodness of fit seems less relevant for statistics that are more complex. Since our focus is on finding the best number of statistics from a class of statistics rather than comparing from one class to another, these different (but very similar) criteria will not affect our conclusions.

4. RESULTS

For each data type we fix a clustering algorithm, an optimality criterion and then look at the GCV for each choice of statistic.

Since the experimenter cannot choose the type of data to be analyzed, we have separated subsections based on 4 data types: Normal with correlation (or independent), independent non-Normal, Normal with serial correlation (ARMA), and dependent non-normal. To present our findings, nested within each of these subsections we have subsubsections, one for each of the three penalties we used: LASSO, Ridge, and Bridge. Within each of the subsubsections we have nested a two way array based on class of statistic and clustering technique. There are 5 classes of statistics (moments, percentiles, PC's, PLS's and SIR's) and six clustering techniques (k-means, agglomerative with 3 link functions, and divisive). Although the clustering must be done before the optimality criterion can be implemented or the statistics calculated, we have put the use of the clustering procedure last in our presentation (when we comment on it at all) because it rarely affected the $[n/K]$ rule in the main cases we studied.

Subject to the slightly different forms of the GCV for moments, percentiles, and PC's versus PLS's and SIR's, it is reasonable to compare different generic choices of statistics within subsubsections because the data type and penalty are common while the clustering technique appears not to matter. Choices of decay parameter are also reasonable to compare but we have not done this; we have defaulted to the automatic selection of decay parameters in the packages we have used. We tested several choices of interval in which to situate the decay parameter but then settled on [.001, 1000].

The common structure among all the results below is a linear regression model with p regressors and n observations. The regression matrix X consists of K blocks, X_1, ..., X_K, and block X_k contains p_k variables and so is $n \times p_k$ with $\Sigma_{k=1}^{K} p_k = p$. Our results are for $n = 10$, $p = 400$, $K = 4$ and $p_k = 100$ for all k, and $N = 200$ iterations unless specified otherwise. We present the computed results below commenting only briey on the patterns they exhibit.

4.1 Normal Data

Here we chose $\in \sim N(0, 1)$, and supposed the blocks of the regression matrix contained variables drawn from $N(0, \Sigma)$, where Σ has 1 on the diagonal and ρ on all the off-diagonal terms. The β_j's were also drawn from a $N(0, 1)$.

4.1.1 LASSO

As noted, LASSO fits are computed by using the LARS package by Hastie and Efron, which uses the LARS algorithm by Efron *et al*. (2003). This package chooses the decay parameter λ by finding the minimum leave-one-out cross validated mean square prediction error.

Moments: Recall we have 2 statistics for each odd order and one for each even order. Table 4.1 shows the GCV error as a function of the correlation and number of moments used in the regression. Here and below an asterisk, *, denotes the minimum in a row. In some cases, we use a dagger, †, to indicate the minimum in a row and an asterisk to indicate the second largest value in that row. This notation means that we believe the apparent minimum is an artefact of the computing rather than accurately approximating the value of the quantity desired. Also, we often omit columns that merely confirm recognizable patterns. For instance, in Table 4.1, we omit the columns for 6-th and higher moments. On the other hand, for comparison purposes, we sometimes include a column at the right labeled 'all' which gives the GCV for the stated penalty using all

the data. In Table 4.1, for instance, LASSO is applied to the 400 variables and not all moments are used.

Table 4.1. Correlated Normal; LASSO; Moments; Known Clusters

#Moments	1	2	3	4	5	all
$\rho = 0$	43.249	41.727	41.205*	41.743	42.062	44.258
$\rho = 0.1$	45.351	44.359*	44.844	44.821	45.132	45.151
$\rho = 0.3$	38.637	34.132*	35.733	35.278	36.317	40.386
$\rho = 0.5$	35.824	33.211	33.162*	33.375	34.406	35.716
$\rho = 0.7$	31.432	26.370*	26.662	27.899	28.441	27.894
$\rho = 0.9$	22.850	16.4458	18.950	19.780	21.597	18.229

It is seen here and in our further examples that within a row, the GCV score is lowest when the number of moments used per block is close to $[n/K]$, the first integer greater than or equal to $n/K = 10/4 = 2.5$, *i.e*. 3, where n is the sample size, here 10, and K is the number of blocks, here 4. So, the total number of statistics is near n.

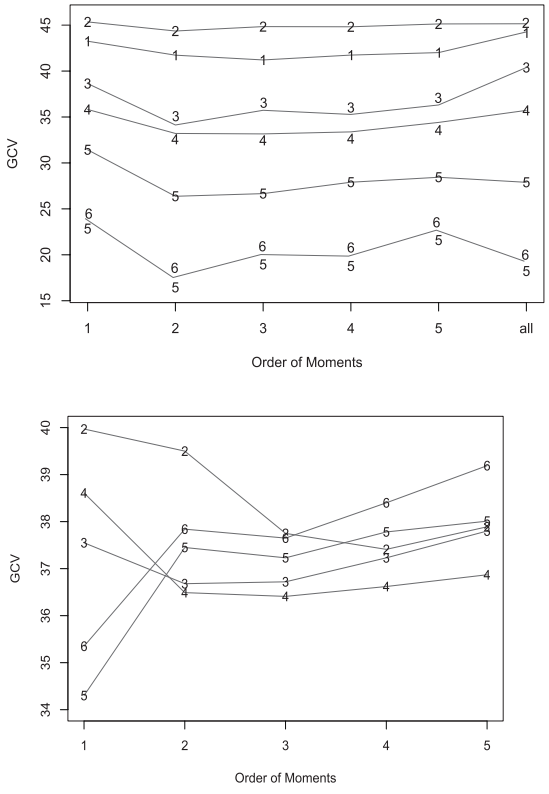

Fig. 1. Above: Graph of the rows in Table 4.1. The numbers on the lines correspond to the row *i.e*., the value of ρ. Below: Graph of Table 4.3 (see below) showing how the number of clusters (indicated by the numbers on the lines) affects the location of the minimum.

The above panel of Fig. 1 shows that for all choices of non-negative correlation, the values in a row form a U-shaped curve as a function of the number of moments. The unique minima occur for 2 or 3 moments, *i.e.*, 3 or 4 statistics tends to give the smallest GCV errors.

A similar case is shown in Table 4.2.

Table 4.2. Same as Table 4.1 but $n = 6$, $p = 300$, $K = 3$

#Moments	1	2	3	4	5
$\rho = 0$	58.351*	59.163	59.619	59.863	60.366

Here, $n/K = 6/3 = 2$ and two first moment statistics are optimal.

Similar results are obtained if we do not use the knowledge that the data come from $K = 4$ independent classes of explanatory variables and we are forced to cluster the data into classes; we chose a range $K = 2, 3, 4, 5, 6$ clusters and used moment based statistics within each block. For clustering to be meaningful, the correlation cannot be zero; we chose $\rho = 0.3$. Results for the K-means and hierarchical clustering are in Tables 4.3 and 4.4. Again, the starred entries tend to be the ones for which the number of statistics is close to $[n/k]$. That is, the asterisks are roughly rising from left to right.

Table 4.3. Normal with $\rho = 0.3$; LASSO, Moments; K-means

#Moment	1	2	3	4	5
$K = 2$	39.970	39.498	37.751	37.408*	37.888
$K = 3$	37.546	36.678*	36.742	37.229	37.802
$K = 4$	38.614	36.492	36.410*	36.623	36.874
$K = 5$	34.299*	37.445	37.235	37.775	38.010
$K = 6$	35.350*	37.841	37.648	38.398	39.185

Table 4.4. Normal with $\rho = 0.3$; LASSO; Moments; Hierarchical

#Moment	1	2	3	4	5
$K = 2$	43.296	41.897	40.317*	40.381	40.586
$K = 3$	38.025	37.675	36.752*	37.085	36.861
$K = 4$	36.684*	37.158	37.512	37.875	38.110
$K = 5$	35.596*	40.193	39.806	40.220	40.196
$K = 6$	36.033*	38.386	38.968	39.523	38.932

The above panel of Fig. 1 shows that for the GCV scores in Table 4.3 the optimal number of statistics (as indicated by the location of the minima in each row of Table 4.3 or its corresponding line in Fig. 1) increases as the number of clusters decreases. That is, within each row, a U-shaped curve of the errors as a function of the number of statistics is seen – even if the left arm of the U is trivial for five or six clusters since the number of statistics cannot be less than one. That is, as we choose more clusters, the optimal number of statistics per cluster decreases, preventing the total number of regressors from increasing much. This observation held for the other clustering algorithms we used, so we omit those results.

Percentiles: Next, we use percentiles in place of moment based statistics. We choose the sequence of percentiles corresponding to probabilities $j/2^k$, where $j = 1, ..., (2^k - 1)$, for $k = 1, 2, 3, 4, 5$. to calculate them effectively we used the R command **quantile**() (Type 7) that uses interpolation to give any quantile for any number of data points. The next 4 tables are parallel Tables 4.1 – 4.4. It is seen that the optimal number of statistics per block remains near [n/K]: In Tables 4.5 and 4.6, 3 statistics are seen to be optimal and $n/K = 2.5$, except in high dependence settings. That is, the results for percentiles are qualitatively the same as for moment based statistics. Like Tables 4.3 and 4.4, Tables 4.7 and 4.8 show the effect of using 2 of the clustering algorithms is minimal. That is, as the number of clusters increases, the optimal number of statistics per cluster decreases, again staying close to [n/K].

Table 4.5. Correlated Normal; LASSO; Percentiles; Known Clusters

#Percentiles	1	3	7	15	31	all
$\rho = 0$	44.216	37.259*	41.459	40.512	41.667	43.225
$\rho = 0.1$	41.394	37.816*	38.794	38.653	39.292	40.083
$\rho = 0.3$	38.321	32.530*	35.388	35.577	35.109	39.441
$\rho = 0.5$	35.929	29.0542*	29.754	29.384	30.469	40.851
$\rho = 0.7$	23.650	20.700	20.665	20.550	20.347*	28.125
$\rho = 0.9$	10.971	8.9469	8.781*	9.1723	8.993	17.504

Table 4.6. Same as Table 4.5 but $n = 6$, $p = 300$, $K = 3$

#Percentiles	1	3	7	15	31
$\rho = 0$	59.622	58.635*	60.468	61.359	62.398

Table 4.7. Normal with $\rho = 0.3$; LASSO, Percentiles; K-means

#Percentiles	1	3	7	15	31
$K = 2$	41.095	36.550	35.588*	35.619	36.045
$K = 3$	39.703	35.121	34.963*	35.230	35.120
$K = 4$	37.624	34.083	34.079*	34.106	34.233
$K = 5$	36.750	33.581*	33.844	33.766	34.680
$K = 6$	36.532	34.103*	34.409	35.198	35.308

Table 4.8. Normal with $\rho = .3$; LASSO, Percentiles; Hierarchical

#Percentiles	1	3	7	15	31
$K = 2$	43.221	40.141	38.005*	39.438	39.735
$K = 3$	41.448	36.665	36.637*	37.494	37.700
$K = 4$	39.410	35.737*	36.140	36.538	37.585
$K = 5$	37.609	35.234*	36.226	35.577	36.914
$K = 6$	36.072	34.620*	36.069	36.340	36.600

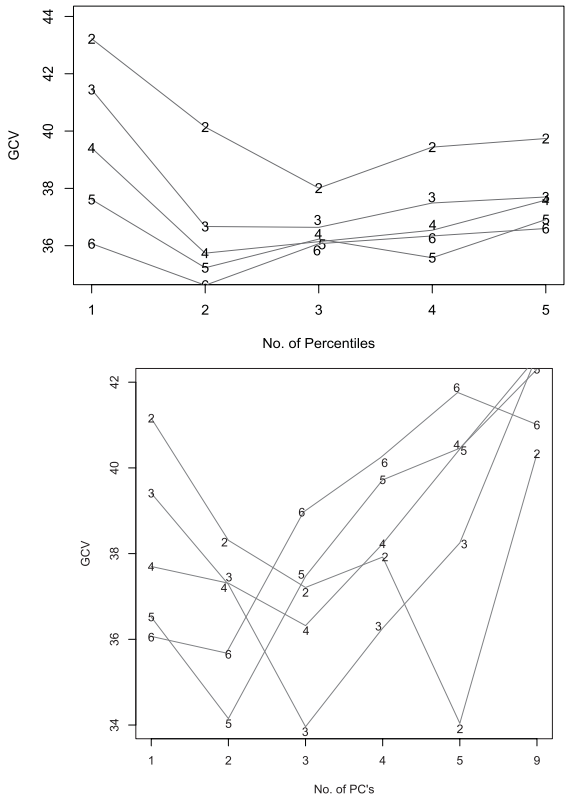

Fig. 2. Above: Graph of the rows in Table 4.8. The numbers on the lines correspond to the row *i.e.*, the value of ρ. Below: Graph of Table 4.11 (see below) showing how the number of clusters (indicated by the numbers on the lines) affects the location of the minimum.

The only difference between Tables 4.7 and 4.8 is the clustering procedure and since they are similar – only differing for $K = 4$ – it is enough to show a plot of Table 4.8. In Fig. 2, the left panel shows that for $K = 4, 5, 6$ the minimum is quite strong at three while for $K = 2, 3$ the minimum is quite strong at seven. Note that since only the binary percentiles were calculated rather than evenly spacing the percentiles over one to 100, the results for percentiles are consistent with the other results even though they do not directly lead to the $[n/K]$ rule.

Principal Components: In this case there are a maximum of $n = 10$ eigenvectors; regressing on all of them is equivalent even though they to using the original explanatory variables.

Tables 4.9 and 4.10 are qualitatively similar to Tables 4.1 and 4.5 and Tables 4.2 and 4.6, respectively. Again, $[n/K] = 3$ for Table 4.9 and $[n/K] = 2$ for Table 4.10, consistent with the starred values. Indeed, it appears in Table 4.9 that PC's may be more stable because three PC's per block is the optimal choice for all the correlations we used. Note that we used nine PC's: We could have used 10 PC's but this would have been degenerate since $n = 10$. Moreover, the minima occurred strictly between one and nine.

Note that we have daggered two entries in Table 4.9: These are for $\rho = 0.7, 0.9$. We suspect, but are unable to establish formally, that the PC algorithm we used broke down. In many computations not shown here we got anomalous results when correlation was high and the number of statistics was at an extreme, either small or large. When we redid Table 4.9 using other values of n (with known clusters), we again found

Table 4.9. Correlated Normal; LASSO; PC's; Known Clusters

#PC	1	2	3	4	5	9
$\rho = 0$	45.095	41.994	41.075*	41.955	42.868	44.774
$\rho = 0.1$	43.515	40.393	39.082*	40.344	41.497	43.148
$\rho = 0.3$	38.203	38.785	35.925*	37.590	39.278	41.891
$\rho = 0.5$	32.728	34.967	32.595*	34.962	37.479	40.697
$\rho = 0.7$	24.966 †	31.705	27.689*	30.678	33.308	38.066
$\rho = 0.9$	12.157 †	24.884	18.424*	23.004	26.008	32.320

Table 4.10. Same as Table 4.9 but $n = 6$, $p = 300$, $K = 3$

#PC's	1	2	3	4	5	all
$\rho = 0$	58.676	55.911*	58.331	61.197	61.303	64.641

that the optimal number of statistics per cluster was $[n/K]$; this is seen in Table 4.10, parallel to Tables 4.2 and 4.6.

As in the last two subsubsections, we examine the effects of clustering. All the algorithms gave qualitatively similar results that matched the earlier cases. For completeness, we show one table using the K-means algorithm. Again, Table 4.11 shows the optimal number of statistics per cluster decreases as the number of clusters increases. It is easy to see that $[n/K]$ is $10/2 = 5$, $10/3 = 3.33$, $10/4 = 2.5$, $10/5 = 2$, and $10/6 = 1.66$ as $K = 2, ..., 6$ in rough agreement with where the asterisks in Table 4.11 appear, validating the optimal number of statistics being approximately $[n/K]$.

Table 4.11. Normal with $\rho = 0.3$; LASSO, PC's; K-means

#PC's	1	2	3	4	5	9
$K = 2$	41.161	38.308	37.210	37.919	34.038*	40.333
$K = 3$	39.408	37.278	33.961*	36.262	38.249	42.620
$K = 4$	37.701	37.307	36.320*	38.227	40.440	42.585
$K = 5$	36.519	34.148*	37.474	39.715	40.462	42.306
$K = 6$	36.047	35.652*	38.938	40.207	41.732	41.902

The rows of Table 4.11 are plotted as lines on the above panel of Fig. 2. For $K = 5, 6$ the minima occurs at two PC's as seen in Table 4.11. When $K = 3, 4$ three PC's achieve the minimum and when $K = 2$, five PC's achieve the minimum.

Partial Least Squares: When we use partial least squares, PLS's, as our statistics the results are similar to the earlier cases, however there is some evidence that the LASSO optimization breaks down as correlation increases. We attribute this to the higher variability of PLS's due to the dependence of PLS's on the Y's as well as the X's.

In Table 4.12, 4 entries are daggered. It seems clear that when ρ is small the algorithm works well and when ρ is large it breaks down. However, on the mid-range, $\rho = 0.3, 0.5$, whether to use a dagger or not is a judgement call. We have used a Principle of Insufficient reason interpolation argument: There is no reason performance of one PLS should be suddenly better than

two PLS's so we presume it isn't. Possibly because of the heightened variability of the PLS's, $[n/K] = [10/4] = [2.5] = 3$, while two statistics is optimal in GCV. On the other hand, $n/K = 2.5$ whereas the smallest nonzero integer difference is one.

Table 4.12. Correlated Normal; LASSO; PLS's; Known Clusters

#PLS	1	2	3	4	5
$\rho = 0$	41.404	40.555*	41.152	41.448	41.691
$\rho = 0.1$	39.904	38.651*	39.106	39.578	39.743
$\rho = 0.3$	38.518 †	39.073*	39.759	39.873	39.943
$\rho = 0.5$	38.721 †	42.230*	43.792	43.966	43.992
$\rho = 0.7$	29.756 †	37.794*	38.619	38.726	38.758
$\rho = 0.9$	15.748 †	40.681*	41.128	41.212	41.211

As before, we investigated performance when the classes of explanatory variable were not known by the procedure. Despite the higher instability seen in Table 4.12, the results when a clustering algorithm was used were qualitatively the same as in the earlier cases: The number of statistics per cluster decreased as the number of clusters increased, at about the same rate. We omit the corresponding tables.

Sliced Inverse Regression: When we used SIR to form summary statistics, the results were somewhat different than for the other choices of statistics. In particular, the results from SIR are much less consistent with the $[n/K]$ rule than the results from PLS are, which in turn are less consistent than for the other three classes of statistics, although not by much. For instance, Table 4.12 shows that the row GCV scores for PLS only have a U-shape for $\rho \leq 0.1$ (although another computational procedure might extend this range to higher values of ρ), and Table 4.13 shows the row GCV scores for SIR

Table 4.13. Correlated Normal; LASSO; SIR; Known Clusters

#SIR	3	4	5	6	7	8
$\rho = 0$	46.556*	46.851	47.326	47.562	47.638	47.790
$\rho = 0.1$	44.323	44.471	45.086	44.158*	44.677	44.826
$\rho = 0.3$	45.137*	45.964	46.353	46.667	46.865	46.507
$\rho = 0.5$	47.854	47.125	46.823	45.691	44.299*	44.796
$\rho = 0.7$	56.178	54.079	49.925	49.890	47.407	46.938*
$\rho = 0.9$	47.390	40.164	34.764	34.042	33.386	32.816*

don't seem to have a pattern at all. Moreover, the U-shape seems to disappear in Table 4.13: The values in a row seem at apart from random fluctuations. Note that columns for one or two statistics have been omited so all the asterisks can be seen and we used a maximum of eight SIR statistics because we reserved one data point for the leave-one-out GCV and noted that using nine SIR statistics would be degenerate.

When the effect of K-means is included, Table 4.14 shows that the optimal number of summary statistics is as high as possible. Results from the use of other clustering algorithms were qualitatively the same. (Columns for one or two statistics have been omited so the asterisks can be seen.) Thus, when clusters are known, SIR does not suggest a convincing optimal number of statistics and, when clusters are unknown, SIR defaults to as many as possible; neither is informative. These phenomena may arise because SIR is such a data dependent procedure and sample size is so small. Indeed, we only have 9 data points since we are doing leave-one-out with $n = 10$. Consequently, we only have 3 slices with 3 data points each. Using a small amount of data in a relatively complicated, data dependent procedure is likely to be unstable or trivial meaning that no dimension reduction via feature selection is possible. Thus, the performance of SIR reflects the small sample size differently when clustering is used or not.

Table 4.14. Normal with $\rho = 0.3$; LASSO, SIR; K-means

#SIR	3	4	5	6	7	8
$K = 2$	46.358	48.121	46.310	45.781	45.726	45.508*
$K = 3$	46.927	46.651	46.388	46.600	46.398	46.125*
$K = 4$	46.461	46.448	46.720	46.315	46.195	45.660*
$K = 5$	46.225	46.686	46.633	46.504	45.882	45.689*
$K = 6$	46.948	47.134	47.366	46.861	46.998	46.320*

4.1.2 Ridge

Unlike LASSO, RR does not do variable selection. That is, where LASSO has a tendency to shrink the coefficients of some terms to zero, ridge shrinks coefficients so they approach zero, but rarely get there.

In this subsection we obtain the GCV errors of ridge regression for the same statistics as in the last subsection. We omit further consideration of the various clustering algorithms since, as seen in the earlier subsections, they do not appear to make a substantial

difference. (Indeed, the effect of clustering on SIR in the last subsection was merely to change the way the data summarization breaks down.)

As with LASSO, the GCV scores tend to decrease as ρ increases. However, within rows, the patterns are harder to discern. Often there is a well-defined, if shallow, U-shape as the number of summary statistics per block increases. Sometimes the U-shape is degenerate in the sense that one arm is missing: The minimum occurs at an extreme rather than on the mid-range. This is strongest for moments, percentiles, and PC's and weaker for PLS and SIR because they depend on the Y's.

We computed the Ridge fits in the closed form, $(X^TX + \lambda I_p)^{-1} X^Ty$ selecting the decay parameter λ by CV over (0.001, 10,000). The optimal λ can be at the boundary (0.001 or 10,000), which suggests that the true optimum will often be effectively zero (no shrinkage) or effectively infinity (shrink everything to zero). We suggest 0.001 and 10,000 are adequate values since the CV did not decrease dramatically when we tested λ's outside that range.

Moments: The results from our simulations using moments with ridge are qualitatively the same as with LASSO. Table 4.15 shows that when $n = 10$ and $K = 4$ choosing 2 or 3 statistics per cluster is optimal so $[n/K]$ statistics is still a approximately optimal.

Table 4.15. Correlated Normal; Ridge; Moments; Known Clusters

#Moments	1	2	3	4	5
$\rho = 0$	46.322	46.069*	49.884	49.915	51.884
$\rho = 0.1$	43.573*	46.319	45.646	48.244	50.077
$\rho = 0.3$	42.212	41.807*	42.788	42.117	44.253
$\rho = 0.5$	31.484*	35.972	34.507	38.824	36.869
$\rho = 0.7$	30.161*	42.460	36.557	45.776	44.018
$\rho = 0.9$	20.935*	35.789	34.991	42.996	42.595

Percentiles: In this case, the results are quite different from what was found with LASSO. In fact, this appears to be a degenerate case because the expression for the ridge fits is $(X^TX+\lambda I_p)^{-1} X^Ty$ meaning that when the X's and errors are generated using a mean zero normal distribution the medians all approach zero. Thus, Table 4.16 shows that one statistic per cluster works best (except when correlation increases). Three statistics per cluster does only slightly worse; we conjecture that using the 33rd and 67th or the first and third quartile *i.e.*, two statistics parallel to the means of the positive and negative values, would give a smaller GCV error than one or three statistics. If this were borne out, the $[n/K]$ rule would be confirmed.

When the correlation is high enough, another tradeoff is seen between number of statistics and the GCV error. Highly correlated data may be easier to predict, hence smaller GCV errors, but accumulating it reflects less information in the sense that the sampling distribution for a statistic based on correlated data will not concentrate as fast with increasing n as in the independent case. So, more statistics are better and the $[n/K]$ rule breaks down.

Table 4.16. Correlated Normal; Ridge; Percentiles; Known Clusters

#Percentiles	1	3	7	15	31
$\rho = 0$	48.088*	49.056	50.805	51.789	51.973
$\rho = 0.1$	44.284*	45.296	46.176	46.090	46.600
$\rho = 0.3$	39.438*	42.475	43.694	43.839	44.004
$\rho = 0.5$	28.846*	30.857	31.551	31.993	32.251
$\rho = 0.7$	29.431	26.393*	27.372	27.574	27.766
$\rho = 0.9$	19.315	16.025	15.443*	16.305	16.325

Principal Components: In contrast to using LASSO with PC's, ridge with PC's achieves the lowest GCV scores when the number of statistics per cluster is maximum, as seen from Table 4.17. This seems to contradict the $[n/K]$ rule.

However, as with PLS and LASSO, within each row, the largest decrease in GCV occurs when passing from 2 to 3 statistics per cluster – and this is in agreement with $[n/K] = [10/4] = [2.5] = 3$. We suggest that with the rounded contours of ridge, which shrinks coefficients but rarely sends them to zero, the largest drop may be more meaningful than the smaller, later reductions which may merely be modeling the noise in the data. (In Table 4.17, † indicates the last big decrease in GCV.)

Table 4.17. Correlated Normal; Ridge; PC's; Known Clusters

#PC	1	2	3	4	5
$\rho = 0$	35.559	23.521	6.046 †	6.107	6.086
$\rho = 0.1$	32.861	21.913	6.822 †	6.694	6.691
$\rho = 0.3$	30.002	20.808	8.981 †	8.828	8.645
$\rho = 0.5$	24.744	15.961	5.370 †	5.077	4.986
$\rho = 0.7$	16.179	10.885	5.351 †	5.130	5.037
$\rho = 0.9$	7.914	5.338	3.712 †	3.617*	3.654

#PC	6	7	8	9	all
$\rho = 0$	6.039	5.975	5.909	5.878	5.874*
$\rho = 0.1$	6.609	6.551	6.581	6.510*	6:542
$\rho = 0.3$	8.428	8.357	8.208	8.203*	8.207
$\rho = 0.5$	4.856	4.819	4.811	4.807	4.780*
$\rho = 0.7$	5.033	5.050	5.021	5.016	4.996*
$\rho = 0.9$	3.624	3.625	3.638	3.661	3.69

Partial Least Squares: Ridge regression with PLS behaves similarly to the earlier cases. Table 4.18 shows that two statistics per cluster is optimal when ρ is small and the largest decrease in GCV is acheived when three statistics per cluster is used as indicated by †; the dependence on the Y's seems to make the GCV scores relatively at for higher correlations. Finding two or three statistics optimal is consistent with $[n/K] = [10/4] = [2.5] = 3$. (Here, †† indicates computational problems.)

Table 4.18. Correlated Normal; Ridge; PLS's; Known Clusters

#PLS	1	2	3	4	5	6	7
$\rho = 0$	53.165	50.940* †	51.816	51.837	51.840	51.840	51.840
$\rho = 0.1$	49.882	48.639* †	49.167	49.158	49.158	49.159	49.159
$\rho = 0.3$	54.792	49.402	48.782 †	48.751	48.750	48.749*	48.750
$\rho = 0.5$	54.056	48.210	46.874 †	46.852*	46.856	46.856	46.856
$\rho = 0.7$	56.630 ††	64.540	61.737 †	61.713	61.711	61.71088*	61.711
$\rho = 0.9$	36.274 ††	99.114	96.324* †	96.719	96.743	96.743	96.743

Sliced Inverse Regression: The rows in Table 4.19 represent $\rho = 0, 0.1, 0.3, 0.5, 0.7, 0.9$. In contrast to LASSO, the results are closer to the $[n/K]$ rule for $\rho \leq 0.3$. Indeed, the optimality of one statistic for $\rho = 0.1$ is only by a very small margin. For higher correlation, $\rho \geq 0.5$, as with LASSO, the optimal number of statistics defaults essentially to the maximum. The high data dependence coupled with the low number of data points per slice means that for high ρ we are forced to use all the SIR's, *i.e.*, there is so much less information all the data is needed. Despite this, the pattern is not strong because the SIR statistics are so variable. Indeed, neither the rows nor columns exhibit strong U-shapes typical of tradeoffs.

Table 4.19. Correlated Normal; Ridge; SIR; Known Clusters

1	2	3	4	5	6	7	8
49.004	48.026*	48.542	48.992	49.356	49.368	49.455	49.757
46.148*	46.163	46.623	46.959	47.303	47.778	47.595	47.949
44.686	43.109*	43.985	45.068	44.772	44.947	44.818	44.763
41.692	40.654	40.394	40.468	40.733	40.467	39.841*	39.917
45.479	45.792	46.391	44.183	43.989	43.227	42.946	41.690*
44.750	44.460	44.593	38.037	34.775	33.042	31.162	30.500*

4.1.3 Bridge

Bridge fits are computed using the brdgrun package, see Fu (1998). The shrinkage parameter γ, defining the penalty, is fixed at 1.5, 2.5 and 3 and λ is chosen by GCV. Since we are looking primarily at the penalty, and computing time is very high for bridge, we have omitted consideration of correlated normal data. We only used $\rho = 0$.

Overall, the results for Bridge do not exhibit a strong pattern because we don't have enough samples. Sometimes there is the expected U-shape, but it is often at (except for PC's) and the minimum occurs at different places. The power γ does not seem to have a consistent effect either, except possibly for moments.

Moments: Table 4.20 shows that as the exponent in the penalty term increases, the optimal number of statistics per cluster seems to decrease from 5 with $\gamma = 1.5$ to 2 when $\gamma = 2.5$. Under ridge, which corresponds to $\gamma = 2$, we saw 3 statistics were optimal when $\rho = 0$. Thus, the cost of statistics to regress on increases with

Table 4.20. Independent Normal; Bridge; Moments; Known Clusters

#Moments	1	2	3	4	5
$\gamma = 1.5$	42.273	41.472	41.048*	43.003	43.187
$\gamma = 2.5$	42.883	42.760*	43.302	45.177	48.135
$\gamma = 3$	44.202*	44.280	45.287	47.348	49.922

the exponent making fewer desirable. This table suggests the $[n = K]$ rule holds only for values of γ close enough to two.

Percentiles: Table 4.21 shows that 1 percentile is optimal for all 3 choices of γ. Thus, percentiles behave differently from moments; however the behavior here is similar to what we saw under ridge for percentiles in Table 4.16. Under LASSO, which corresponds to $\gamma = 1, 3$ percentiles were optimal in the $\rho = 0$ case. This makes sense because LASSO puts a smaller penalty on the size of the β_j's possibly making it worthwhile to regress on more statistics. Again, the $[n/K]$ rule does not appear to hold.

Table 4.21. Independent Normal; Bridge; Percentiles; Known Clusters

#Percentiles	1	3	7	15	31
$\gamma = 1.5$	37.912*	38.018	38.893	39.717	39.658
$\gamma = 2.5$	38.027*	39.009	40.194	41.338	43.393
$\gamma = 3$	38.865*	40.148	41.585	43.210	44.786

Principal Components: Here, Bridge replicates what we found under ridge but differs from the corresponding case under LASSO which found 3 statistics optimal. Table 4.22 shows stability for the various values of γ. However, although not shown, the values also reveal that the last large decrease occurs from 3 to 4 statistics per cluster suggesting four statistics whereas $[n/K] = [10/4] = 3$ meaning the $[n/K]$ rule is weakly validated.

Table 4.22. Independent Normal; Bridge; PC's; Known Clusters

#PC	6	7	8	9
$\gamma = 1.5$	17.204	16.110	15.578*	16.018
$\gamma = 2.5$	16.758	15.385	14.938*	15.239
$\gamma = 3$	16.324	14.868	14.421*	14.761

Partial Least Squares: It is tempting to suggest Table 4.23 implies that the number of PLS's to be used should increase with rather than decrease as in Table 4.20. However, the flatness of the values in the rows means the U-shape is weak. This is different from the corresponding cases with ridge and LASSO where stronger patterns were seen. However, it is possible that the values of γ here lead to a higher sensitivity to the greater data dependence of the PLS's than in the earlier cases. (Note that in Table 4.23 the columns correspond to the number of PLS's used and the rows correspond to $\gamma = 1.5, 2.5, 3$.) Overall, there may be more instability with PLS in the Bridge case even as the size of the penalty leads to more statistics being optimal. That is, the $[n/K]$ rule may apply for moderate values of γ only.

Table 4.23. Independent Normal; Bridge; PLS's; Known Clusters

1	2	3	4	5	6	7	8
45.104	44.661*	44.794	45.011	45.013	45.014	45.014	45.014
46.081	47.1416	45.573*	45.637	45.632	45.633	45.633	45.633
46.128	50.595	46.039	45.814	45.793	45.792	45.792*	45.792*

Sliced Inverse Regression: Table 4.24 gives results for SIR that are fairly consistent with the $[n/K]$ rule; this was not the case for ridge or LASSO. It seems that the higher sensitivity to the data of SIR does not matter here – the exact opposite from PLS above. On the other hand, this may be similar to what was observed in Table 4.19.

Table 4.24. Independent Normal; Bridge; SIR; Known Clusters

#SIR	1	2	3	4	5
$\gamma = 1.5$	40.711	39.857*	41.101	41.534	41.275
$\gamma = 2.5$	40.893	39.736*	41.433	42.209	42.346
$\gamma = 3$	41.034	39.773*	41.510	42.324	42.569

4.2 Independent Non-normal Data

The results for the cases we tried using independent non-normal data were qualitatively the same as for independent normal data. To see an example of this, consider some different design matrices. Table 4.25 shows results using LASSO on principle components, with X drawn from the distributions listed.

Table 4.25. Independent non-normal design matrix; LASSO; PC's; Known Clusters

#PC	1	2	3	4	5	all
Normal(0, 1)	43.661	42.034	40.362*	41.114	41.485	43.598
Double exp(1)	45.197	42.595	41.638*	42.578	43.021	44.620
Uniform(–5, 5)	44.239	39.591	38.671*	39.232	40.318	44.739
Exp(1)	41.745	39.509	37.717*	38.339	39.515	41.065

The results for these independent non-normal cases all have a U-shape in each row, and all the minima occur near where we expect *i.e.*, at $[n/K] = [10/4] = [2.5] = 3$, independent of the distribution. This suggests that the independence of the data is more important than its distribution. This makes sense because often dependence affects the informativity of data more than the shape of the distribution does.

4.3 Serially Dependent Normal Data

Although somewhat atypical, we investigate how the optimality criteria and statistic selection perform on ARMA($p.q$) data. Our motivation is that ARMA data is one proxy for real data whose structure and properties cannot be safely assumed to be of any form. We chose values of p and q to be small, 0, 1 and 2, with $n = 10$. We present results for the LASSO criterion with PC's because it was the easiest to compute. However, other statistics that we tried gave results not too different from before. Results for ridge, too, were similar. We did not investigate Bridge because the computing demands were too high.

For $AR(1)$ data, Table 4.26 was representative for LASSO, AR data, statistics that do not depend on Y (*i.e.*, moments, percentiles and PC's), and various clustering strategies. It is seen that the optimal number of statistics is three, $[n/K] = [10/4] = 3$.

For MA terms data, the variability increases so that for the settings we considered it is difficult to recognize regularities in behavior as indicated by Table 4.27. Again, $[n=K] = 3$ but, at best, three is only weakly preferred.

Table 4.26. AR(1), LASSO, PC's, Known Clusters

#PC	1	2	3	4	5
$\rho = 0.1$	43.229	41.839	39.414*	42.947	41.754
$\rho = 0.3$	43.767	40.952	37.339*	39.781	41.371
$\rho = 0.5$	40.194	39.385	37.971*	38.509	39.985
$\rho = 0.7$	42.341	39.582	35.972*	37.673	38.532
$\rho = 0.9$	39.593	36.738	35.446*	36.180	36.976

Table 4.27. MA(1), LASSO, PC's, Known Clusters

#PC	1	2	3	4	5	all
$\rho = 0.1$	42.405	40.986	39.489	40.102	39.324	38.014*
$\rho = 0.3$	47.006	44.731	43.901*	45.731	46.553	48.678
$\rho = 0.5$	47.004	40.238*	42.018	42.634	44.070	49.080
$\rho = 0.7$	41.904	40.476	39.236	39.173	38.991*	44.209
$\rho = 0.9$	41.820	40.343	39.729*	40.610	41.024	42.244

4.4 Dependent Non-normal Data

As a test of our methods and conclusions we generated dependent non-normal data by transformations of serially correlated normal data. In general, apart from random variation, $[n/K]$ identifies the typically optimal number of statistics to use. Specifically, we used three transformations on the randomly generated X data so that the ARMA properties were disrupted, namely, $y = \arctan(x)$, $y = \sin(x)x$, and $y = \log(x - \min(x))$. For contrast to the last subsection, Table 4.28 presents results for moments rather than for PC's. It is seen that five statistics per cluster is optimal, a little higher than $[n/K]$.

Table 4.28. Transform AR(1) by $y = \arctan(x)$; LASSO; Moments; Known Clusters

#Moment	1	2	3	4	5
$\rho = 0.1$	44.106	40.841	39.243*	40.034	40.137
$\rho = 0.3$	43.184	39.469	37.968	37.925*	38.565
$\rho = 0.5$	46.770	41.611	39.681*	41.174	42.379
$\rho = 0.7$	46.520	43.252	42.266*	42.684	44.748
$\rho = 0.9$	43.462	39.313	37.677*	38.489	39.647

The results for transformed, non-normal correlated data are consistent for all correlation structures and transformations we tested. That is, the optimal number of statistics per cluster was $[n/K]$ or a little larger apart from cases of extreme flatness in the rows. Sometimes the identification was through actual minimality; sometimes it was through looking at the last large decrease – this latter being more typical of ridge.

We did not examine different optimality criteria (Bridge), clustering algorithms, or Y dependent statistics in this setting. However, based on computed results in parallel cases, some included here and many not, we

suggest that the results would not qualitatively differ substantially from the earlier cases presented here.

5. TWO LIMITATIONS

Here we observe two limitations on the main point so far. The first is that a more exact determination of the optimal number of statistics than $[n/K]$ seems hard to justify. Indeed, in the first subsection we present 3 computations in which the number of statistics per block is sensitive to the block sizes but less than might be expected. Second, we address the minimality we have been identifying: Even though the standard deviations of the GCV scores is high the regularities are hard to explain by chance.

5.1 Block Size and Number of Statistics

It is intuitive that summarizing more variables should require more statistics than summarizing fewer variables. However, we find that while one may want an extra statistic or two for relatively large blocks of variables, the benefits may be small.

In Table 5.1 we fixed the number of statistics at a total of 12 and used 400 explanatory variables distributed over 4 classes for $n = 10$ samples. Since we are using LASSO and PC's with known clusters and correlated normal data our results are directly comparable to Table 4.9 where three statistics per cluster were found optimal. We used $N = 2000$ replications but even so the results are suggestive rather than strong.

Table 5.1. Principle Components unequal classes (325, 25, 25, 25)

#PC		(3, 3, 3, 3)	(6, 2, 2, 2)	(9, 1, 1, 1)	all
$\rho = 0$	$N = 2000$	39.958	39.852*	43.219	43.452
$\rho = 0.1$	$N = 2000$	39.722	39.391*	42.823	43.778
$\rho = 0.3$	$N = 2000$	35.267*	35.755	39.305	39.060
$\rho = 0.5$	$N = 2000$	30.013*	30.362	33.675	32.383
$\rho = 0.7$	$N = 2000$	22.489*	22.667	25.405	23.677
$\rho = 0.9$	$N = 2000$	12.713	12.977	15.220	12.196*

In Table 5.1, one class is much larger than the other three, which are equal. Although 325/25 = 13 the choice of (9, 1, 1, 1) as the number of statistics drawn from each block is not seen to be optimal. In fact, (6, 2, 2, 2) and (3, 3, 3, 3) are seen to be optimal in all cases except extreme dependence, $\rho = 0.9$, where (6, 2, 2, 2) and (3, 3, 3, 3) perform only slightly worse than using all the data directly.

Similar results are found if one class is unusually small compared to the other classes, with sizes, say, (125, 125, 125, 25) giving 125/25 = 5 and if the classes consist of two large and two small classes, say (175, 175, 25, 25) with 175/25 = 7. That is, using a slightly larger number of statistics per class, is better, but only slightly so. More data might improve the discrimination over numbers of statistics per cluster, but, in practice, data sets with $p/n \geq 400/10 = 40$ are common.

5.2 Standard Deviations of GCV Scores

To conclude our presentation of results, Table 5.4 is a reprint of Table 4.9 but with the average standard deviation, SD, of each entry shown in parentheses below it. Clearly, the SD's are large, typically over half the means.

Table 5.4. Principle Components

#PC	1	2	3	4	5
$\rho = 0$	45.095 (23.281)	41.994 (24.348)	41.075* (23.127)	41.955 (23.189)	42.868 (23.420)
$\rho = 0.1$	43.515 (23.177)	40.393 (23.475)	39.082* (22.897)	40.344 (23.518)	41.497 (23.552)
$\rho = 0.3$	38.203 (20.228)	38.785 (23.180)	35.925* (21.261)	37.590 (21.831)	39.278 (22.353)
$\rho = 0.5$	32.728 (18.779)	34.967 (22.973)	32.595* (20.692)	34.962 (21.859)	37.479 (22.774)
$\rho = 0.7$	24.966 † (15.041)	31.705 (23.246)	27.689* (19.098)	30.678 (20.296)	33.308 (22.093)
$\rho = 0.9$	12.157 † (8.224)	24.884 (25.333)	18.424* (14.792)	23.004 (18.682)	26.008 (21.090)

The large SD's are the result of n being small. So, the reason that our entry to entry comparisons within a row in the tables in Section 4 is meaningful stems from the fact the findings are consistent. That is, the variability in the individual entries is very large so the validity of the conclusions stems from the fact that the roughly the same number of statistics per cluster, $[n/\mathrm{K}]$, is seen to be optimal over a very wide range of scenarios. If one were to set H_0: '$[n/K]$ is far from the optimal number of statistics to choose' any reasonable hypothesis test (frequentist or Bayes) would be rejected.

6. DISCUSSION

In a large p small n context, we have investigated the roles of clustering, optimality criteria, choice of statistics and data type. As a generality, we found that for each case there was an optimal number of statistics to choose. This was seen by evaluating GCV errors for a range of different numbers of statistics per block of variables, and in some cases, by different allocations of statistics to blocks. The GCV errors often generated a U-shaped curve if they were plotted as a function of number of statistics per block. We found that the penalty term was the most important choice to be made while the clustering algorithm made little difference. The data type mattered somewhat mostly in terms of correlation or other notions of dependence and the choice of statistics mattered somewhat, especially whether the statistics depended on the response or not.

For Ridge Regression and the LASSO, using moments, percentiles, or PC's, we found that choosing $[n/K]$ statistics per block, or maybe one more, generally worked well. That is, the benefits from using different numbers of statistics (still totalling n or a little more) for the clusters seemed to be small. The same was usually true for PLS's and SIR's but often little or discrimination was possible leading to degenerate results. For Bridge regression, the results were not clear at all; we suggest that a simple heuristic will be hard to obtain because the penalty terms are so often large, *e.g.*, $\gamma \geq 2.5$. Indeed, higher penalty terms usually led to a smaller number of statistics as optimal. This is an intuitive result broadly consistent with, say, the Representer theorem that finds at most n terms necessary when there are n data apoints.

Thus, aside from Bridge regression, our results suggest that moments, percentiles, and PC's were generally equally good as summary statistics while PLS's and SIR's were often good but exhibited more variability. Indeed, our results strongly suggest that in $p >> n$ settings, using feature selection gives smaller GCV errors than using all the data 'as is'. In the Bridge case, the $[n/K]$ rule sometimes held, but the pattern was not clear. Otherwise put, typically, generic feature selection followed by a shrinkage method typically gave better predictive performance than using all the data.

Let s denote the number of statistics per block. Then, the U-shaped curves represent the result of a bias curve and of a variance curve. As s increases from 1, the total number of statistics increases, so bias decreases while variance may increase. If we start with a high value of s , then we may have low bias but excessive variance. The optimal s, indicated by asterisks in our tables, here represents an optimal tradeoff between variance and bias in terms of the number of statistics chosen chosen from a class. Unsurprisingly, the optimal variance bias tradeoff is often achieved when the number of statistics is related to the block size, however the improvement is over using the same number of statistics per block was small in the cases we examined.

We have commented in Section 5.2 that the small size of n was insufficient to make discrimination over sets of statistics reliable even though the $[n/K]$ rule was supported. So, this opens the question of how large n should be relative to p for good predictive properties. First, over many simulations choosing $[n/K]$ statistics per cluster worked reasonably well when $n/p = 40$ and better when n/p was smaller where p was the number of variables. The usual heuristic is that one wants 10 data points per parameter for estimation however here the optimal number of parameters is around $K[n/K] \approx n$ given that one has already clustered the data, chosen a class of summary statistics and intends to use a shrinkage criterion. Loosely, the issue for how large n/p should be for good performance comes down to how small the GCV and its standard error is. Thus, a further simulation study could increase n slowly for fixed p until the GCV and its standard error were suficiently small in practice that an indiviudal use on a single data set would be likely to have a GCV that was satisfactorily small, assuming that the model class was large enough that bias was not excessive. Note that while this would be informative, in many data sets n is fixed, cannot be increased, and the goal is to reduce p. Here we have shown that reducing p to n actually gives the best performance in many cases using shrinkage methods.

We comment that we have not done a complete bias-variance decomposition for the GCV since the effect of clustering has been neglected. Obviously, summarizing data by statistics from clusterings may introduce further bias as well as variability. However, one may regard the bias and variance from clustering to have been implicitly represented in the GCV scores. Alternatively, one may regard the apparent 'bias' of using a particular set of summary statistics as having both pure bias and variance conditional on the bias. That is, the pure bias part of the GCV is the minimal bias that could be achieved on average by a set of statistics of the specified form while the variance conditional on the bias may be regarded as the variance of finding those statistics implicit in the use of

clustering. That is, the two stages of data summarization and using the summary statistics in a shrinkage criterion are conceptually disjoint even though here they have been combined in one overall GCV score.

Another feature of our work here is the complete neglect of the interpretability of data summarization. It is our philosophical stance that asking for interpretability in complex short fat data problems restricts the search for good predictors so much that interpretability per se is more harm than help: It prematurely restricts the models so much that poor prediction is nearly assured. Indeed, optimal predictive solutions such as model averaging and kernel methods routinely give 'models' that are uninterpretable. The optimality of such methods over interpretable methods shows that asking for interpretability is often predictively harmful.

Overall, the take home lesson seems to be that, if there are n data points and no sufficient statistic (even in a heuristic sense) is available, then there are n pieces of information that can be regarded as n values of a statistic that may come from any one of a large number of classes. While one might argue that some measurements are more informative than other measurements, on average, no matter how the explanatory variables are summarized it is difficult for much more than or much less than n statistics to be genuinely useful in the sense of giving better results predictively than just using all the data. Moreover, as long as the statistics do not depend on Y and the penalty is not too different from absolute or squared error, it may not matter much which statistics are used for predictive purposes. That is, for predictive purposes, detailed modeling may not be much more useful than generic feature selection. One can argue that generic feature extraction may give unstable solutions, however, interpretable feature selection can also be unstable, and both are just efforts to deal with model uncertainty. In cases where interpretable feature selection is infeasible or unreliable, it is probably better to use generic feature selection and uncertain answers (and admit the uncertainty) than not to get answers at all.

ACKNOWLEDGEMENTS

Both Clarke and Chu gratefully acknowledge support from an NSERC operating grant Clarke used to hold in Canada. In addition, Chu acknowledges support from NIH grant 1K99HL114651.

REFERENCES

Austin, E., Pan, W. and Shen, X. (2013). Penalized regression and risk prediction in genome-wide association studies. *Stat. Anal. Data Mining*, **6(4)**, 315-328.

Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.J. (1984). *Classification and Regression Trees.* Wadsworth, Belmont CA.

Clarke, B. and Severinski, C. (2011). Subordinators, adaptive shrinkage and a prequential comparison of three sparsity methods. Invited comment on Shrink globally, act locally by Polson and Scott. *Proceedings of the IX Valencia Conference on Bayesian Statistics,* Eds. Bernardo, J.M. *et al.,* 523-528. Oxford Univ. Press.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004). Least angle regression. *Ann. Statist*., **32**, 407-499.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Amer. Statist. Assoc.,* **96,** 1348-1360.

Frank, I.E. and Friedman, J.H. (1993). A statistical view of some chemometrics regression tools. *Technometrics,* **35(2)**, 109-135.

Fu, W. (1998). Penalized regressions: The bridge versus the LASSO. *J. Comp. Graph. Stat*., **7(3)**, 397-416.

Hawkins, D., Basak, S. and Shi, X. (2001). QSAR with few compounds and many features. *J. Chem. Inf. Comp. Sci*., **41**, 663-670.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *Elements of Statistical Learning.* Springer-Verlag, New York.

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

McKay, C. (2004). Automatic genre classification of MIDI recordings. M.A. Thesis, Department of Theory, Faculty of Music, McGill University.

Li, Ker-chau (1991). Sliced inverse regression for dimension reduction. *J. Amer. Statist. Assoc*., **86(414)**, 316-327.

Stenning, D., Lee, T., van Dyk, D., Kashyap, V., Sandell, J. and Young, C. (2013). Morphological feature extraction for statistical learning with applications to solar image data. *Stat. Anal. Data Mining*, **6**, 329345.

Struyf, A., Hubert, M. and Rousseeuw, P. (1996). Clustering in an object oriented environment. *J. Stat. Software*, **1**.

Tibshirani, R. (1996). Regression shrinkage and selection via the LASSO. *J. Roy. Statist. Soc., Ser.* B, **58(1)**, 267-288.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *J. Roy. Statist. Soc.*, **76,** 301-320.