

Regularization and Estimation in Regression with Cluster Variables

Qingzhao Yu¹, Bin Li²

¹School of Public Health, Louisiana State University Health Sciences Center, New Orleans, LA, USA

²Department of Experimental Statistics, Louisiana State University, Baton Rouge, LA, USA

Email: gyu@lsuhsc.edu, bli@lsu.edu

Received 14 October 2014; revised 5 November 2014; accepted 15 November 2014

Copyright © 2014 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Clustering Lasso, a new regularization method for linear regressions is proposed in the paper. The Clustering Lasso can select variable while keeping the correlation structures among variables. In addition, Clustering Lasso encourages selection of clusters of variables, so that variables having the same mechanism of predicting the response variable will be selected together in the regression model. A real microarray data example and simulation studies show that Clustering Lasso outperforms Lasso in terms of prediction performance, particularly when there is collinearity among variables and/or when the number of predictors is larger than the number of observations. The Clustering Lasso paths can be obtained using any established algorithm for Lasso solution. An algorithm is proposed to construct variable correlation structures and to compute Clustering Lasso paths efficiently.

Keywords

Clustered Variables, Lasso, Principal Component Analysis

1. Introduction

We are often interested in finding important variables that are significantly related to the response variable and can be used to predict quantities of interest in regressions and classification problems. Important variables are often shown in clusters where variables in the same cluster are highly correlated and have similar pattern relating to the response variable. For example, a major application of microarray technology is to discover important genes and pathways that are related to clinical outcomes such as the diagnosis of a certain cancer. Typically, only a small proportion of genes from a huge bank have significant influence on the clinical outcome of interest. In addition, expression data frequently have cluster structures: the genes within a cluster often share

the same pathway and are therefore similarly related to the outcome. When regression is adapted in this setting, we often face the challenge from multi-collinearity of covariates. An ideal variable selection procedure should be able to find all genes of important clusters rather than just some representative genes from the clusters. Typically, two characteristics, pointed out by [1], evaluate the quality of a fitted model: accuracy of prediction on new data and interpretation of the model. For the latter, the sparse model with fewer selected covariates is preferred for interpretation due to its simplicity. However, when multiple variables share the same mechanism for explaining the response, all the involved variables should have an equal chance of being selected, and should exhibit the same relationship to the outcome in the fitted model, for scientific reasoning.

It is well known that the ordinary least square estimate (OLS) in linear regression often performs poorly when some of the predictors are highly correlated. OLS would generate unstable results where the estimates have inflated variances. Regularizations have been proposed to improve OLS. For example, ridge regression [2] penalizes the model complexity by the l_2 penalty of the coefficients. This method was proposed to solve the collinearity problem by adding a constant to the diagonal terms of $X'X$, where X is the observation or design matrix. Ridge regression stabilizes the estimates through the bias-variance trade-off. It can often improve the predictions but cannot select variables. [3] proposed the Lasso method by imposing an l_1 -penalty on the regression coefficients. Lasso is a promising method, as it can improve prediction and produce sparse models simultaneously. However, when high correlations among predictors are present, the predictive performance of Lasso is dominated by ridge regression [3]. Moreover, when there is a cluster of variables, in which each variable associates with the response variable similarly, Lasso tends to arbitrarily select one variable from the cluster instead of identifying the cluster [1]; see also Section 2 for more discussion. Elastic Net, proposed by [1], combines both l_1 and l_2 penalties of the coefficients as the regularization criterion. The method is promising in that it encourages cluster effects and shows improved predictive performance over Lasso. Elastic Net can automatically choose cluster variables and estimate parameters at the same time. Many other methods can be used to choose clustered variables, such as principal component analysis (PCA). [4] defined “eigen-arrays” and “eigen-genes” in this way. But PCA can not choose sparse models. [5] proposed sparse principal component analysis (SPCR), which formulated PCA as a regression-type optimization problem, and then obtained sparse loadings by imposing the Elastic Net constraint. SPCR can successfully yield exact zero loadings in principal components. However, for each principal component, a regularization parameter has to be selected, which results in an overwhelming computational burden when the number of parameters is large. Other penalized regression methods have been proposed for group effect [6]-[13]. However, these methods either pre-suppose a grouping structure or assume that each predictor in a group shares an identical regression coefficient.

In practice, we often have some prior knowledge about the structure of variables and would like to make use of a priori information in analysis. For example, in gene analysis, we know the pathways and genes involved in these pathways. Therefore, we would like to group the involved variables in the same pathway together. Another example is in spatial analysis, we would like to keep a certain correlation structure among the spatial error terms. For example, sometimes we would like to fit a different coefficient for a certain variable at different regions (e.g., if the variable has different effect at different regions) but keep a correlation structure among the coefficients at neighborhood regions. The conditional autoregressive model (CAR, [14]) is one of the methods that can be used to keep such correlation structure.

In this paper, we propose a method that encourages cluster variables to be selected together and can incorporate available prior information on coefficient structures in variable selection. When there is no prior information on coefficient structure, we propose a data augmentation algorithm to find the structure. Moreover, the method uses the Lasso regularization to choose sparse models. The proposed method can be solved by any efficient Lasso algorithm such as least angle regression (LARS, [15]) and the coordinate-wise descent algorithm (CDA, [16]). We call our method the Clustering Lasso (CL).

The rest of the paper is organized as follows. In Section 2, we review the Lasso method and discuss its limitation in identifying clustered variables. Then we propose the Clustering Lasso in a Bayesian setting. Its counterparts in the Frequentist setting and computational strategies are discussed in Section 3. Sections 4 and 5 demonstrate the predictive and explanatory performance of CL through real examples and simulations. Finally, conclusions and future work are discussed in Section 6.

2. Clustering Lasso in Bayesian Setting

Consider linear regression settings with the response vector $y = (y_1, \dots, y_n)'$ and $n \times p$ dimensional input

matrix \mathbf{X} . The \mathbf{y} and columns of \mathbf{X} are centered and standardized to have the same l_2 norm. The Lasso estimates $\tilde{\boldsymbol{\beta}}_{\text{lasso}}$ are calculated by minimizing

$$\frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^p |\beta_j| \tag{1}$$

The solution of Lasso can be obtained through LARS or CDA. Compared with ordinary linear regressions, Lasso shows superior predictive performance and more stable estimates. Moreover, Lasso can often select variables and estimate coefficients simultaneously.

Group effect has been defined by [1] in the linear regression setting. Let x_i be the i th predictor. The estimates of coefficients have the group effect if $x_i = x_j$ would result in the estimated coefficients $\hat{\beta}_i = \hat{\beta}_j$. [1] further proved that if the solution for estimation is to minimize the objective function of the form:

$$\frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda J(\boldsymbol{\beta}) \tag{2}$$

and the penalty term, $J(\cdot)$, is strictly convex, then the estimates from Equation (2) enjoy the group effect property. In Lasso, $J(\cdot)$ is l_1 norm of $\boldsymbol{\beta}$, which is not strictly convex. Zou and Hastie proved that in this case Lasso estimates do not have the group effect. This is also understandable through the Lasso solution path from LARS. In LARS, suppose a variable x_i is selected in the model. Its coefficient solution path will move in a direction to reduce the correlation between x_i and the current residual, $\mathbf{y} - \mathbf{X}\tilde{\boldsymbol{\beta}}$, until another variable, say x_k , has the same correlation to the current residual as does x_i . At this point, variable x_k is added into the model. If x_j is highly correlated with x_i , when the correlation between x_i and the residual decreases, so does that between x_j and the residual. Therefore, if x_i has been included in the model, Lasso is less likely to select the highly correlated variable x_j in the model. Consequently, Lasso cannot select clustered variables.

In a Bayesian setting, if \mathbf{x}_i is the i th row of \mathbf{X} , [3] showed that the Lasso solution is identical to the posterior mode of the coefficients when the prior distributions of the coefficients are set as independent double exponential distributions, where

$$y_i | \mathbf{x}_i, \boldsymbol{\beta}, \sigma \sim N(\mathbf{x}_i' \boldsymbol{\beta}, \sigma^2), \text{ for } i = 1, \dots, n,$$

$$\pi(\boldsymbol{\beta} | \sigma^2) = \prod_{j=1}^p \frac{\lambda}{2\sigma^2} e^{-\lambda |\beta_j| / \sigma^2}.$$

In Lasso, the penalty term of model complexity is $\sum_{j=1}^p |\beta_j|$. Because each coefficient is penalized equally, each one can be shrunk to zero independently. When the variables are clustered, an ideal solution path should be that the clustered variables are selected together. Therefore, we would like to penalize the coefficients with a restriction that keeps the correlation structure among the variables. With the penalization, if the coefficient of one variable is nonzero, those variables in the same cluster are less likely to be zero. For this purpose, we assume a correlation structure, specified as the structural correlation matrix \mathbf{R} , of the coefficients $\boldsymbol{\beta}$.

For simplicity, assume that the variance of the random error, σ^2 ($\sigma > 0$), and the structural correlation matrix \mathbf{R} are known. Then the likelihood and prior distributions can be set as:

$$y_i | \mathbf{x}_i, \boldsymbol{\beta}, \sigma \sim N(\mathbf{x}_i' \boldsymbol{\beta}, \sigma^2), \text{ for } i = 1, \dots, n,$$

$$\boldsymbol{\beta} = \mathbf{R}^{\frac{1}{2}} \boldsymbol{\beta}^*, \text{ and}$$

$$\pi(\boldsymbol{\beta}^* | \sigma^2) = \prod_{j=1}^p \frac{\lambda}{2\sigma^2} e^{-\lambda |\beta_j^*| / \sigma^2}.$$

Therefore, the posterior distribution of $\boldsymbol{\beta}$ has the form

$$\pi(\boldsymbol{\beta} | \mathbf{y}, \mathbf{R}) \propto l(\mathbf{y} | \boldsymbol{\beta}) \cdot \pi\left(\mathbf{R}^{-\frac{1}{2}} \boldsymbol{\beta}\right) \left| \mathbf{R}^{-\frac{1}{2}} \right|$$

$$\propto \exp\left(-\sum_{i=1}^n (y_i - \mathbf{x}_i' \boldsymbol{\beta})^2 / 2\sigma^2\right) \exp\left(-\lambda \left\| \mathbf{R}^{-\frac{1}{2}} \boldsymbol{\beta} \right\|_1 / \sigma^2\right), \tag{3}$$

with a vector $V' = (V_1, \dots, V_p)$, $\|V\|_1 = \sum_{i=1}^p |V_i|$. The posterior mode of β in the distribution (3) is the solution to

$$\operatorname{argmin}_{\beta} \frac{1}{2} (y - X\beta)' (y - X\beta) + \lambda \left\| R^{-\frac{1}{2}} \beta \right\| \tag{4}$$

Relating Equation (4) to the Bayesian Lasso solution to (1), we naturally infer the Clustering Lasso in Frequentist setting.

3. Clustering Lasso

3.1. Clustering Lasso and Its Grouping Effect

In Frequentist setting, we modify the penalization function in Lasso to retain a presumed correlation structure among coefficients. Let $\beta^* = R^{-\frac{1}{2}} \beta$ and β_j^* be the j th element of β^* . The Clustering Lasso estimate is defined as the solution to

$$\frac{1}{2} (y - X\beta)' (y - X\beta) + \lambda \sum_{j=1}^p |\beta_j^*| \tag{5}$$

where $\lambda \geq 0$ is the regularization parameter. Note that instead of restricting $\sum |\beta_j|$, we restrict $\sum |\beta_j^*|$.

Therefore, β 's are not penalized independently and clustered variables could be chosen. Let $r_j^{-\frac{1}{2}}$, with

dimension $1 \times p$, be the j th row of $R^{-\frac{1}{2}}$ and let $R_j^{-1} = \left(r_j^{-\frac{1}{2}} \right)' r_j^{-\frac{1}{2}}$, a $p \times p$ matrix. The penalty term used in

expression (5) can also be written as $\lambda \sum_{j=1}^p (\beta' R_j^{-1} \beta)^{1/2}$, which is intermediate between the l_1 penalty and the l_2 penalty. When R is an identity matrix, the Clustering Lasso is identical to the ordinary Lasso method. Otherwise, the penalty function is strictly convex. Using Lemma 2 developed by [1], the solution to Expression (5) has the group effect. Therefore, Clustering Lasso can select variables by clusters.

Figure 1 illustrates the Clustering Lasso penalty contours with two predictors. The right figure shows the penalty contour when the two predictors are correlated and the left one shows the contour when the two predictors are independent, which is identical to the Lasso method. The sums of the squared errors have elliptical contours, centered and minimized at the full least squares estimate. The constraint region of Lasso is the diamond region $|\beta_1| + |\beta_2| \leq c$, while that for the Clustering Lasso is the parallelogram region defined by $\left| r_1^{-\frac{1}{2}} \beta \right| + \left| r_2^{-\frac{1}{2}} \beta \right| \leq c$. The optimal estimates are realized at the place where the elliptical contours first hit the constraint regions. The sides of the parallelogram are decided by the structural correlation matrix R .

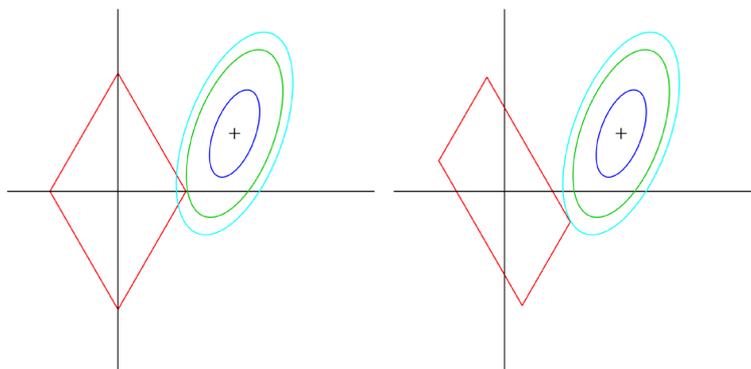


Figure 1. Estimation picture for the Clustering Lasso when two predictors are independent (left, as lasso) and when two predictors are clustered (right).

3.2. Computation

The Clustering Lasso is an extension of the Lasso method. Let $X^* = \mathbf{X}\mathbf{R}^{\frac{1}{2}}$. So the solution to Expression (5) is $\tilde{\beta} = \mathbf{R}^{\frac{1}{2}}\tilde{\beta}^*$, where $\tilde{\beta}^*$ is

$$\operatorname{argmin}_{\beta^*} \frac{1}{2}(\mathbf{y} - \mathbf{X}^* \beta^*)'(\mathbf{y} - \mathbf{X}^* \beta^*) + \lambda \sum_{j=1}^p |\beta_j^*| \quad (6)$$

Therefore, all the established algorithms for Lasso solution, such as the least angle regression (LARS, [15]), could be used for Clustering Lasso.

3.3. The Clustering Lasso Algorithm

We can incorporate prior knowledge of clustering into a structural correlation matrix. For example, Kyoto Encyclopedia and Genes and Genomes (KEGG) and many other biological databases can be referred to in gene analysis to construct the structural correlation matrix. It is required that the structural correlation matrix be symmetric. When no prior information is readily adaptable, a natural method is to use the modified correlation matrix of the observed data, meaning that the coefficients should have a correlation structure that is similar to how the covariates are correlated. There are several well-established potential choices such as partial correlation matrix [17]. In this paper, we propose to use a modified correlation matrix so that if two variables x_i and x_j are not significantly correlated, R_{ij} , the i th row and j th column element of \mathbf{R} , is set to be zero. As the solution for β is $\tilde{\beta} = \mathbf{R}^{\frac{1}{2}}\tilde{\beta}^*$, zero elements in $\mathbf{R}^{\frac{1}{2}}$ are desired so that when β_i^* s are shrunk to zero, which is possible by the Lasso property, some β_j s could also be shrunk to exact zero.

In detail, we develop Algorithm 1—the Clustering Lasso algorithm. Let p_{val} , p_2 , and m , in $[0,1]$ be three prespecified numbers, and $CORR$ be a $p \times p$ matrix.

Algorithm 1 *Clustering Lasso*

1. For $i = 1, \dots, p-1$
for $j = 2, \dots, p$
do correlation test between x_i and x_j , let

$$CORR[i, j] = \begin{cases} \operatorname{cor}(x_i, x_j) & \text{if } p \text{ value} < p_{\text{val}} \text{ and } |\operatorname{cor}(x_i, x_j)| \geq m \\ 0 & \text{o.w.} \end{cases}$$

2. Do eigen decomposition on $CORR$ so that $CORR = \mathbf{U}\mathbf{Q}\mathbf{U}'$ and let $Q_{ii} = 0$ if $\frac{Q_{ii}}{\sum_{j=1}^p Q_{jj}} < p_2$ for

$i = 1, \dots, p$.

3. Let $\mathbf{T} = \mathbf{U}\mathbf{Q}^{\frac{1}{2}}\mathbf{U}'$ and $\mathbf{X}^* = \mathbf{X}\mathbf{T}$.
4. Do Lasso on $(\mathbf{y}, \mathbf{X}^*)$ and get the coefficient solution $\tilde{\beta}^*$.
5. $\tilde{\beta} = \mathbf{T}\tilde{\beta}^*$ is the solution to Clustering Lasso.

Note that only when some elements of \mathbf{T} are set to be zero, could β s be shrunk to exact zero when β^* 's are shrunk to zero by Lasso. A special case is when \mathbf{R} is a block diagonal matrix. To choose sparse models, we need to identify clusters of covariates, where variables in the same cluster are assumed to be correlated while those from different clusters are independent. For this purpose, there are two shrinkage steps in Algorithm 1. Step (1) shrinks the correlation coefficients to zero if there is no significant correlation between the pair of covariates at the significance level p_{val} or if the magnitude of correlation is smaller than a pre-set value m . When two covariates are not correlated, there is little chance that the two variables relate to the response variable with the same underlying pathway. Therefore, the coefficients of the two variables can be estimated independently. Step (2) shrinks some eigen-values of $CORR$ to zero if the corresponding eigenvector explains less than p_2 times the total variance of $CORR$. The two shrinkage steps cannot guarantee that some elements of \mathbf{T} be zero. Subjective intervention can help for this purpose. One resolution is to cluster the covariates first

and then calculate the correlation matrices for each cluster, which in turn used to build the diagonal blocks of \mathbf{R} . In addition to building a diagonal block matrix, another resolution is to adapt shrinkage methods in the eigen decomposition process of \mathbf{R} , so that some loadings of the eigenvectors might degenerate to 0. ScotLASS [18] and sparse principal component analysis (Zou *et al.*, 2006) can serve this purpose. However, these methods require extra computations for each principal component, which brings in high computational costs. The nonzero elements of the j th row of \mathbf{T} imply that the corresponding covariates belong to the j th cluster. Ideally, their values should be proportional to the contributions of each covariate to the cluster in explaining the outcome. As pointed out by a referee of the paper, clusters in the proposed method are identified by rows of $\mathbf{R}^{1/2}$, where $\mathbf{R}^{1/2}$ is defined as $\mathbf{U}\mathbf{Q}^{1/2}\mathbf{U}'$ with \mathbf{Q} being the diagonal matrix of eigenvalues and \mathbf{U} columns of eigenvectors of \mathbf{R} . As in principal component analysis, the nonzero elements of $\mathbf{R}^{1/2}$ are difficult to interpret in practice. The referee recommends setting the elements of $\mathbf{R}^{1/2}$ to be 0 or 1 based on the absence or presence of non-zero elements, respectively. In the paper, we set the estimate of β to be zero, if its estimated value is very close to zero, *i.e.* if $|\beta_i| < 0.005$.

3.4. Choice of Tuning Parameters

Four parameters, $(p_{\text{val}}, m, \lambda, p_2)$, are to be chosen for Algorithm 1. p_{val} is the significance level used to decide whether the correlations between a pair of covariates should be considered to restrict the estimation of their coefficients. We usually select the significance level at 0.05, the traditional significance level. When the data set is large, we can reduce the significance level. Since the correlation would be always significant when little correlation exists and the number of observations is large, we set another restriction on the magnitude of correlation- m , above which we would like to use the correlation as a restriction to the coefficient parameters. m is chosen subjectively by researchers. Algorithm 1 Step (2) is similar to the principal component analysis except that the eigen decomposition is based on the correlation matrix modified by Step (1). p_2 specifies the minimum proportion of variance explained by the eigen vector, below that, the eigen vector will not be used for further analysis. p_2 is set at a small value, typically $0.01/p$, where p is the total number of covariates.

The last parameter to be tuned is λ . In Lasso, the conventional tuning parameter is the fraction (s) of the l_1 -norm. There are well-established methods for choosing s . Tenfold cross-validation (CV) on training data is the method we used in this paper. The training dataset is divided into ten folds randomly. One fold of the data is used as validation data, on which the prediction error is calculated based on the model fitted from the other nine folds of data. s is tested on a fine grid on $[0,1]$. It takes the value that minimizes the averaged prediction error from CV. We can also use ten-fold CV to tune m and p_2 . We found that only a few representative values for $m \times p_2$ need to be cross validated to obtain good results, which are $\{0, 0.5\} \times \left\{0, 0.05, \frac{0.01}{p}\right\}$.

4. Microarray Data Example

We used the proposed method on an Affymetrix gene expression dataset. The data were collected by Singh *et al.* [19] and consists of 12,600 genes, from 52 prostate cancer tumor samples and 50 normal prostate tissue samples. The goal is to construct a diagnostic rule based on the 12,600 gene expressions to predict the occurrence of prostate cancer. Support vector machine (SVM, [20]), Ridge, Lasso, Elastic Net, Weighted Fusion (w.fusion) and Clustering Lasso were all applied to this dataset. We tried four types of Clustering Lasso methods:

1. CL1: $p_{\text{val}} = 0.05$, $m = 0$, and $p_2 = 0$;
2. CL2: $p_{\text{val}} = 0.05$, $m = 0$, and $p_2 = 0.05$;
3. CL3: $p_{\text{val}} = 0.05$, $m = 0$, and $p_2 = \frac{0.01}{\text{\# of covariates}}$;
4. CL4: $p_{\text{val}} = 0.05$, $m = 0.5$, and $p_2 = 0.05$.

To apply these methods, we first coded the presence of prostate cancer as a 0-1 (no and yes) response y . The classification function is I (fitted value > 0.5), where $I(\cdot)$ is the indicator function. For comparison, we randomly select 52 samples as training data, based on which the diagnostic rules are constructed, and the rules are in turn tested on the remaining 50 samples.

The dataset was split 20 times. For each repetition, a 1000-gene set was preselected based on the training data

to make the computation manageable. The genes are those “most significantly” related to the response, tested by individual t -statistics. **Figure 2** shows the boxplots of the misclassification rates on the test data sets from different classifiers. The misclassification rates are summarized in **Table 1**. Overall, the misclassification rate from Clustering Lasso is competitive with Elastic Net and Ridge, and is better than Lasso, Weighted Fusion, and SVM. For the computational time, Clustering Lasso is comparable to the Lasso method and is much more efficient than Elastic Net and Weighted Fusion. Within the four Clustering Lasso methods, the ones with more restrictions on eigenvalues and the magnitudes of correlations perform a little bit worse.

Table 2 shows the average number of genes selected from the 20 repetitions based on different methods. The analyses were based on 1000 genes and 52 observations. We see that Lasso selected fewer than 52 genes. Elastic Net eliminated few genes—the average number of selected genes was close to 1000. Cluster Lasso identified about 25% genes as important. However, we do not know whether the chosen genes are, in fact, important or not. The efficiency of variable selection is further assessed by simulation studies.

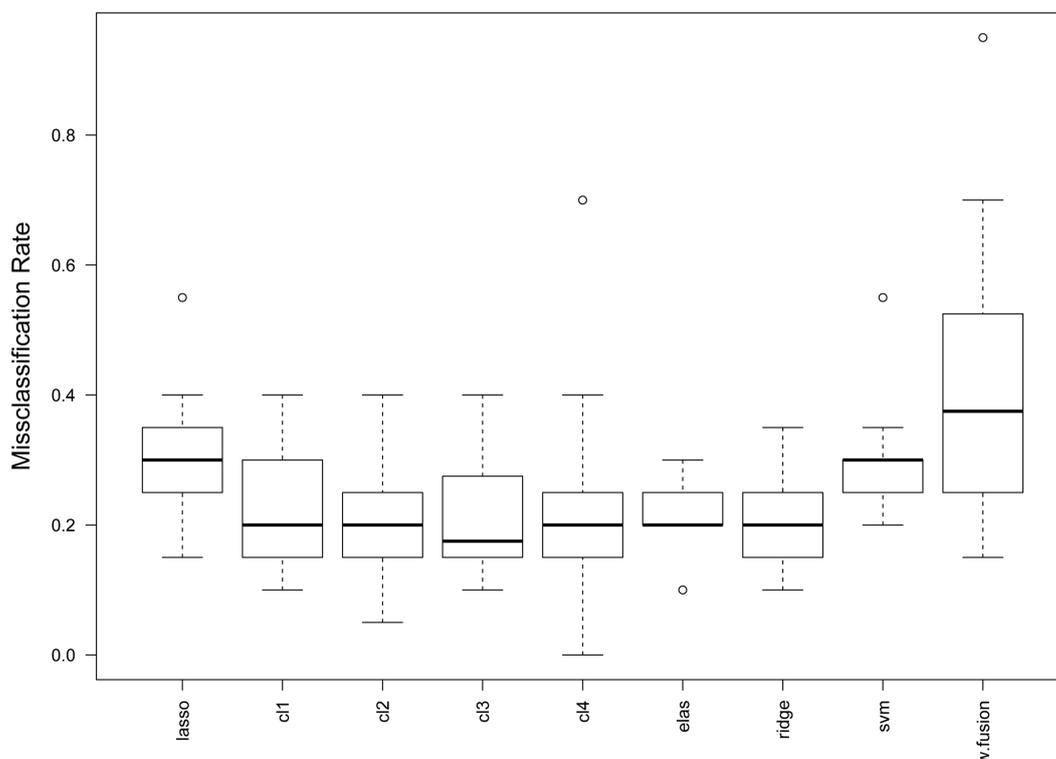


Figure 2. Misclassification rates on singh data. ELAS stands for Elastic Net.

Table 1. Summary of Misclassification Rates on Singh data.

Methods	SVM	Ridge	Elastic Net	Lasso	CL1	CL2	CL3	CL4	W. fusion
Mean	5.75	4.25	4.3	6.05	4.45	4.2	4.2	4.55	8.2
Median	6	4	4	6	4	4	3.5	4	7.5
SD	1.48	1.33	0.98	1.79	1.64	1.74	1.64	2.86	4.03

Table 2. Average number of genes selected by each method.

Methods	Elastic Net	Lasso	CL1	CL2	CL3	CL4	W. fusion
# of genes	999.25	42.25	278.35	221.50	287.05	160.40	856.65

5. Simulation Studies

We applied Clustering Lasso on some simulations to test its prediction accuracy in regressions when compared with Ridge, Lasso, Elastic Net, and Weighted Fusion. The first three simulations are adapted from the Elastic Net paper [1]. To begin, datasets are simulated from the true model:

$$y = X\beta + \sigma\epsilon, \quad \epsilon \sim N(0,1)$$

For each scenario, we simulated 100 data sets, each consisting of a training data set and an independent test data set. Here are the details of the four scenarios.

1. In example one, we simulated 40 observations as training data and 200 observations as test data. We let $\beta = (0.85, 0.85, 0.85, 0.85, 0.85, 0.85, 0.85, 0.85)$ and $\sigma = 3$. The pairwise correlation between x_i and x_j was set to be $\text{corr}(i, j) = 0.5^{|i-j|}$.

2. In Example two, we simulated 200 training data and 400 testing data. There are 40 predictors such that

$$\beta = \left(\underbrace{0, \dots, 0}_{10}, \underbrace{2, \dots, 2}_{10}, \underbrace{0, \dots, 0}_{10}, \underbrace{2, \dots, 2}_{10} \right), \quad \sigma = 15, \text{ and } \text{corr}(i, j) = 0.5$$

3. Example 3 has the group setting that $\beta = \left(\underbrace{3, \dots, 3}_{15}, \underbrace{0, \dots, 0}_{25} \right)$ and $\sigma = 15$, where the predictors are generated as

$$\begin{aligned} x_i &= z_1 + \epsilon_i^x, \quad z_1 \sim N(0,1), \quad i = 1, \dots, 5; \\ x_i &= z_2 + \epsilon_i^x, \quad z_2 \sim N(0,1), \quad i = 6, \dots, 10; \\ x_i &= z_3 + \epsilon_i^x, \quad z_3 \sim N(0,1), \quad i = 11, \dots, 15; \\ x_i &\stackrel{\text{iid}}{\sim} N(0,1), \quad i = 16, \dots, 40; \\ \epsilon_i^x &\stackrel{\text{iid}}{\sim} N(0,0.01), \quad i = 1, \dots, 15. \end{aligned}$$

As explained by [1], three groups are equally important groups, and each group contains five covariates. We created 100 observations as training data and 400 as testing data.

The fourth simulation is a modification of the third example to emphasize the group effects. The true model has the form $y = z_1 + 0.5z_2 + \epsilon$ where $\epsilon_i \sim N(0,1)$. The predictors we observed are

$$\begin{aligned} x_i &= z_1 + \epsilon_i^x, \quad z_1 \sim N(0,1), \quad i = 1, \dots, 5; \\ x_i &= z_2 + \epsilon_i^x, \quad z_2 \sim N(0,1), \quad i = 6, \dots, 10; \\ x_i &= 0.6z_2 + \epsilon_i^x, \quad i = 11, \dots, 15; \\ x_i &= z_3 + \epsilon_i^x, \quad z_3 \sim N(0,1), \quad i = 16, \dots, 20; \\ \epsilon_i^x &\stackrel{\text{iid}}{\sim} N(0,0.5), \quad i = 1, \dots, 20. \end{aligned}$$

The latent variables, z_1 and z_2 , directly relate to the response variable, where z_1 is more important than z_2 . A nuisance variable z_3 , does not related to y . x_i s relate to z s at different levels. In terms of gene analysis, we can think of z_1 , z_2 and z_3 as underlying pathways, some of which are related to the disease measured by y . We observed the gene expression levels, x_i , and would like to identify the related pathways.

We used all four Clustering Lasso methods. In all examples, the results from the four Clustering Lasso methods are close to each other. The prediction results from Lasso, CL2, CL4, Elastic Net, Ridge, and Weighted Fusion are summarized in Table 3 and Figure 3. In Figure 3, relative MSE was defined as the MSE of the corresponding method divided by the minimum MSEs from all the methods. We see that Clustering Lasso always performs better than the Lasso method, and it is close to or better than Ridge, Weighted Fusion and Elastic Nets, even under collinearity and group effect situations.

Table 4 shows the results of variable selection. The two numbers in each cell are the proportion of times an important factor is chosen and the proportion of times a false factor is chosen, respectively. We see that compared with Elastic Net, Weighted Fusion and Lasso, Clustering Lasso is superior at selecting important factors. However, like Weighted Fusion, it is more likely to over select variables than Elastic Net. In Example 2,

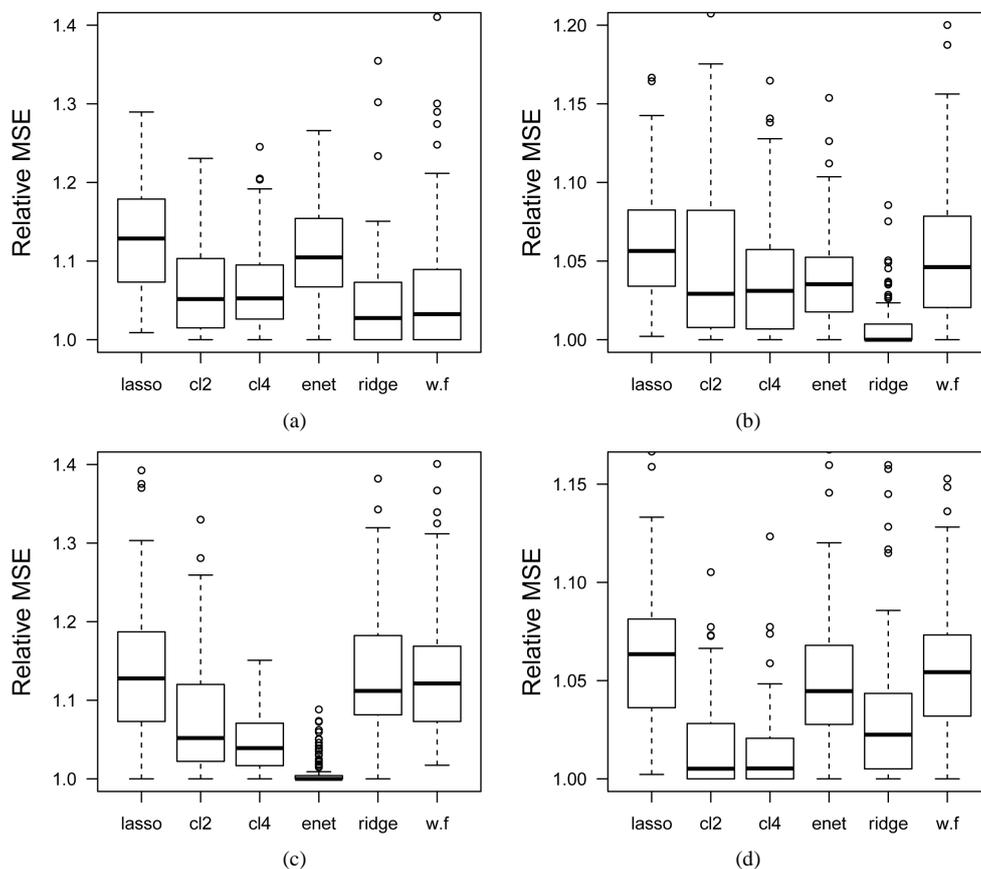


Figure 3. Comparing the simulation results from the four examples. (a)-(d): Example 1-4.

Table 3. Mean (standard deviation) of MSE for the simulated examples based on the 100 iterations.

Methods	Example 1	Example 2	Example 3	Example 4
Lasso	11.50 (1.81)	256.40 (19.05)	279.75 (30.08)	1.151 (0.09)
Elastic Net	11.23 (1.60)	251.01 (18.85)	248.42 (24.45)	1.138 (0.10)
Ridge regression	10.55 (1.46)	243.47 (15.91)	278.09 (25.81)	1.125 (0.10)
Clustering Lasso 2	10.68 (1.46)	253.75 (19.11)	265.33 (28.82)	1.097 (0.08)
Clustering Lasso 4	10.70 (1.35)	250.82 (17.78)	257.33 (23.46)	1.094 (0.08)
Weighted Fusion	10.68 (1.69)	257.07 (24.22)	287.85 (61.14)	1.141 (0.09)

Table 4. Variable selection results for the simulated examples based on the 100 iterations. In each cell, the first number is the proportion of times a true factor is chosen and the second number is the proportion of times a false factor is chosen.

Methods	Example 1	Example 2	Example 3	Example 4
Lasso	0.840, -	0.811, 0.389	0.235, 0.736	0.544, 0.186
Elastic Net	0.870, -	0.838, 0.488	0.958, 0.134	0.585, 0.124
Clustering Lasso 2	0.995, -	1.00, 0.998	1.00, 0.873	0.991, 0.630
Clustering Lasso 4	0.985, -	1.00, 0.997	1.00, 0.493	0.995, 0.460
Weighted Fusion	0.990, -	0.992, 0.975	1.00, 0.997	0.792, 0.574

since all variables are highly correlated, Clustering Lasso cannot identify the most important variables. In comparison, Clustering Lasso performs very well in Examples 3 and 4, when clusters of variables play an important role in real model.

Finally, to show how Clustering Lasso chooses covariates in groups and the behavior of the coefficients for the selected variables, we illustrate the differences between Lasso and Clustering Lasso by a modified example from [1]. Let z_1, z_2 and z_3 be three independent variables with the uniform $(0,20)$ distribution. The response variable is generated as $y \sim N(z_1 + 0.2z_2, 1)$. With the random error terms $\epsilon_i \stackrel{\text{ind}}{\sim} N(0, 1/16)$, the nine observed predictors are

$$\begin{aligned} x_1 &= z_1 + \epsilon_1, & x_2 &= -z_1 + \epsilon_2, & x_3 &= z_1 + \epsilon_3, \\ x_4 &= z_2 + \epsilon_4, & x_5 &= -z_2 + \epsilon_5, & x_6 &= z_2 + \epsilon_6, \\ x_7 &= z_3 + \epsilon_7, & x_8 &= -z_3 + \epsilon_8, & x_9 &= z_3 + \epsilon_9. \end{aligned}$$

The variables x_1, x_2 and x_3 are from group 1, with the direct effect z_1 . x_4, x_5 and x_6 are from group 2, with the direct effect z_2 . The effect from z_2 on y is much smaller than from z_1 —the coefficient for z_1 is 1 compared with 0.2 for z_2 . Variables x_7, x_8 and x_9 are from z_3 , which does not relate to the response variable. The within-group correlations are almost 1, while the between group correlations are almost 0. **Figure 4** shows the solution paths for Lasso, Elastic Net and CL2.

We also use this simulation to compare the sensitivity and specificity of the listed methods in finding significant covariates. The simulation is repeated 100 times. **Table 5** summarizes the number of times that the coefficients of x_i are not zero. We find that the proposed Clustering Lasso of all versions can uniformly identify the important covariates while is less likely to select non-significant covariates than Lasso, Elastic Net and Weighted Fusion.

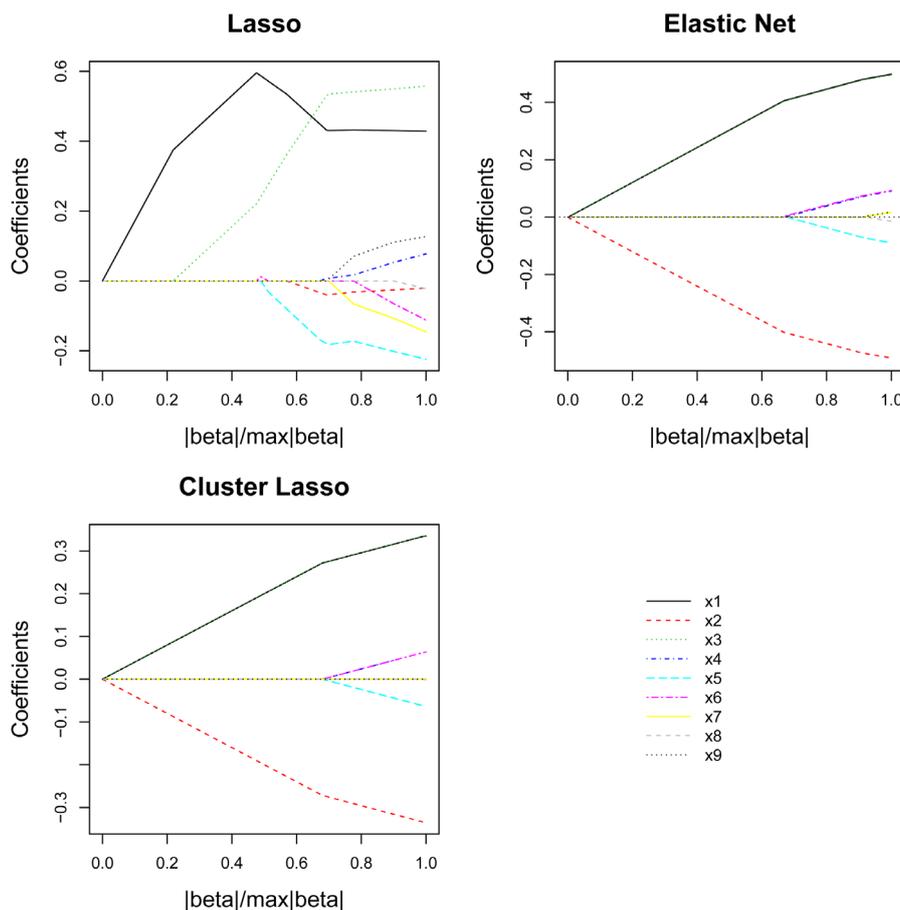


Figure 4. Comparing the solution paths from Lasso, Elastic Net and Clustering Lasso.

Table 5. Number of times the coefficients are not zero based on the 100 repetitions.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
Lasso	86	84	89	69	75	64	73	61	66
Elastic Net	93	93	94	88	91	85	40	42	37
Clustering Lasso 1	100	100	100	100	100	100	58	59	61
Clustering Lasso 2	100	100	100	100	100	100	32	31	30
Clustering Lasso 3	100	100	100	100	100	100	33	35	32
Clustering Lasso 4	100	100	100	100	100	100	33	33	33
Weighted Fusion	100	99	100	100	95	100	85	83	85

6. Conclusions and Future Works

We find that the Clustering Lasso, is a novel predictive model that produces sparse model with good predictive performance, while encouraging group effects. The empirical results from a two-class microarray data classification problem and several simulation studies on regression problems show that Clustering Lasso has very good predictive performance and is superior to the Lasso method.

The method was proposed to encourage group effects so that clustered variables are selected together in a model. Clustering Lasso can automatically select groups of variables. If the structural correlation matrix used for regularization is block diagonal matrix, Clustering Lasso is equivalent to the group Lasso proposed by [7]. However, if the relationships among the variables are complicated, we have to simplify the structural correlation matrix to obtain sparse models. We proposed some shrinkage steps to build the desired structural correlation matrix. Rotating the eigen vectors or adapting techniques such as sparse component analysis can also help for this purpose. As a next step, we will use the Clustering Lasso method in the spatial analysis, so that we can maintain the important spatial correlations while selecting sparse models.

Acknowledgements

We thank Mrs. Patricia Andrews for editing the paper.

References

- [1] Hoerl, A.E. and Kennard, R.W. (1970) Ridge Regression: Application to Nonorthogonal Problems. *Technometrics*, **12**, 69-82. <http://dx.doi.org/10.1080/00401706.1970.10488635>
- [2] Tibshirani, R. (1996) Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society: Series B*, **58**, 267-288.
- [3] Zou, H. and Hastie, T. (2005) Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society: Series B*, **67**, 301-320. <http://dx.doi.org/10.1111/j.1467-9868.2005.00503.x>
- [4] 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 of the United States of America*, **97**, 10101-10106.
- [5] Zou, H., Hastie, T. and Tibshirani, R. (2006) Sparse Principal Component Analysis. *Journal of Computational and Graphical Statistics*, **15**, 265-286. <http://dx.doi.org/10.1198/106186006X113430>
- [6] Tibshirani, R., Saunders, M., Rosset, S., Zhu, J. and Knight, K. (2005) Sparsity and Smoothness via the Fused Lasso. *Journal of the Royal Statistical Society: Series B*, **67**, 91-108. <http://dx.doi.org/10.1111/j.1467-9868.2005.00490.x>
- [7] Yuan, M. and Lin, Y. (2006) Model Selection and Estimation in Regression with Grouped Variables. *Journal of the Royal Statistical Society: Series B*, **68**, 49-67. <http://dx.doi.org/10.1111/j.1467-9868.2005.00532.x>
- [8] Bondell, H.D. and Reich, B.J. (2008) Simultaneous Regression Shrinkage, Variable Selection, and Supervised Clustering of Predictors with OSCAR. *Biometrics*, **64**, 115-123. <http://dx.doi.org/10.1111/j.1541-0420.2007.00843.x>
- [9] Daye, Z.J. and Jeng, X.J. (2009) Shrinkage and Model Selection with Correlated Variables via Weighted Fusion. *Computational Statistics & Data Analysis*, **53**, 1284-1298. <http://dx.doi.org/10.1016/j.csda.2008.11.007>
- [10] Jenatton, R., Obozinski, G. and Bach, F. (2010) Structured Sparse Principal Component Analysis. *International Con-*

ference on Artificial Intelligence and Statistics (AISTATS).

- [11] Jenatton, R., Audibert, J.Y. and Bach, F. (2011) Structured Variable Selection with Sparsity-Inducing Norms. *Journal of Machine Learning Research*, **12**, 2777-2824.
- [12] Huang, J., Ma, S. and Zhang, C.H. (2011) The Sparse Laplacian Shrinkage Estimator for High-Dimensional Regression. *Annals of Statistics*, **39**, 2021-2046. <http://dx.doi.org/10.1214/11-AOS897>
- [13] Buhlmann, P., Rutimann, P., van de Geer, S. and Zhang, C.H. (2013) Correlated Variables in Regression: Clustering and Sparse Estimation. *Journal of Statistical Planning and Inference*, **143**, 1835-1858. <http://dx.doi.org/10.1016/j.jspi.2013.05.019>
- [14] Besag, J. (1974) Spatial Interaction and the Statistical Analysis of Lattice Systems (with Discussion). *Journal of the Royal Statistical Society, Series B*, **36**, 192-236.
- [15] Efron, B., Johnstone, I., Hastie, T. and Tibshirani, R. (2004) Least Angle Regression. *Annals of Statistics*, **32**, 407-499. <http://dx.doi.org/10.1214/009053604000000067>
- [16] Friedman, J., Hastie, T., Hofling, H. and Tibshirani, R. (2007) Pathwise Coordinate Optimization. *Annals of Applied Statistics*, **1**, 302-332. <http://dx.doi.org/10.1214/07-AOAS131>
- [17] Schafer, J. and Strimmer, K. (2005) A Shrinkage Approach to Large-Scale Covariance Estimation and Implications for Functional Genomics. *Statistical Applications in Genetics and Molecular Biology*, **4**, 32.
- [18] Jolliffe, I.T., Trendafilov, N.T. and Uddin, M. (2003) A Modified Principal Component Technique Based on the Lasso. *Journal of Computational and Graphical Statistics*, **12**, 531-547. <http://dx.doi.org/10.1198/1061860032148>
- [19] Singh, D., Febbo, P., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., Lander, E.S., Loda, M., Kantoff, P.W., Golub, T.R. and Sellers, W.R. (2002) Gene Expression Correlates of Clinical Prostate Cancer Behavior. *Cancer Cell*, **1**, 203-209. [http://dx.doi.org/10.1016/S1535-6108\(02\)00030-2](http://dx.doi.org/10.1016/S1535-6108(02)00030-2)
- [20] Guyon, I., Weston, J., Barnhill, S. and Vaapnik, V. (2002) Gene Selection for Cancer Classification Using Support Vector Machines. *Machine Learning*, **46**, 389-422. <http://dx.doi.org/10.1023/A:1012487302797>

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or [Online Submission Portal](#).

