

Constrained Grouped Sparsity

Yi Guo^{1,*}, Junbin Gao², and Xia Hong³

¹ CSIRO Mathematics and Information Sciences
North Ryde, NSW 1670, Australia

yi.guo@csiro.au

² School of Computing and Mathematics,
Charles Sturt University, Bathurst, NSW 2795, Australia

jbgao@csu.edu.au

³ School of Systems Engineering, University of Reading, Reading, RG6 6AY, UK
x.hong@reading.ac.uk

Abstract. In this paper, we propose an algorithm encouraging group sparsity under some convex constraint. It stems from some applications where the regression coefficients are subject to constraints, for example nonnegativity and the explanatory variables are not suitable to be orthogonalized within groups. It takes the form of the group LASSO based on linear regression model where a L1/L2 norm is imposed on group coefficients to achieve group sparsity. It differs from the original group LASSO in the following ways. First, the regression coefficients must obey some convex constraints. Second, there is no requirement for orthogonality of the variables within individual groups. For these reasons, the simple blockwise coordinate descent for all group coefficients is no longer applicable and a special treatment for the constraint is necessary. The algorithm we proposed in this paper is an alternating direction method, and both exact and inexact solutions are provided. The inexact version simplifies the computation while retaining practical convergence. As an approximation to group L0, it can be applied to data analysis where group fitting is essential and the coefficients are constrained. It may serve as a screening procedure to reduce the number of the groups when the number of total groups is prohibitively high.

1 Introduction

In this paper, we consider the variable selection problem for the following linear regression model

$$\mathbf{y} = \sum_{i=1}^M \mathbf{x}_i \beta_i + \epsilon \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^D$ is the dependent variable, $\mathbf{x}_i \in \mathbb{R}^D$ are the explanatory variables (the i th column from the full explanatory matrix $\mathbf{X}$), $\beta_i \in \mathbb{R}$ are corresponding coefficients, and ϵ is an error term. The variable selection problem in this setting

* The author to whom all the correspondences should be addressed.

means to choose M from N total explanatory variables for (1) to achieve some optimum. For a set of indices $\mathcal{A} = \{a_1, \dots, a_M\}$, let $X_{\mathcal{A}} = [\mathbf{x}_{a_1}, \dots, \mathbf{x}_{a_M}]$ be the $D \times M$ submatrix of $\mathbf{X}$ and $\beta_{\mathcal{A}} = [\beta_{a_1}, \dots, \beta_{a_M}]^T$ the corresponding subvector of β ($\beta = [\beta_1, \dots, \beta_N]^T$). The problem is usually formulated as the following least squares problem

$$\begin{aligned} \min_{\mathcal{A}, \beta_{\mathcal{A}}} & \|\mathbf{y} - \mathbf{X}_{\mathcal{A}}\beta_{\mathcal{A}}\|_2^2 \\ \text{s.t. } & |\mathcal{A}| = M \end{aligned} \quad (2)$$

so that the optimum is described by the least residuals sum of squares (RSS). If M is unknown, which is usually the case, there is a further problem of model selection.

In some real world applications such as spectroscopy, it is often the case that the variables have some group structure. For example, in the unmixing problem [1] for mineral composition analysis, variables (typical mineral spectra) having very high correlations are often from the same group. It is desirable to fit those groups instead of individual variables, i.e. in the regression model, the variables in a group either stay or leave altogether. Formally, we rewrite the model as follows

$$\mathbf{y} = \sum_{g=1}^M \sum_{i=1}^{N_g} \mathbf{x}_{g_i} \beta_{g_i} + \epsilon, \quad (3)$$

where N_g is the number of variables in group g . Accordingly the group wise least square regression becomes

$$\min_{\beta_{g_i}} \|\mathbf{y} - \sum_{g=1}^M \sum_{i=1}^{N_g} \mathbf{x}_{g_i} \beta_{g_i}\|_2^2, \quad (4)$$

and subset selection takes the same form of (2) but each element in $\mathcal{A}$ is now a set, i.e. $a_g = \{a_{g_1}, \dots, a_{g_{N_g}}\}$ corresponding to indices of variables in group g . In other words, the subset selection is performed in a group fashion, that is to select M out of total G groups that minimizes the RSS in (4).

Subset selection is a combinatorial optimization problem in essence and hence has NP complexity. There are quite a few of methods trying to solve (2) heuristically [2,3,4]. Full subset selection [5], a fast exhaustive search based on QR decomposition and branch-and-bound techniques, guarantees the best subset of given size, i.e. M , is found. Its extension, called grouped subset selection (GSS) [6], finds M groups minimising RSS in (4). However it only works for moderate size problems, e.g. less than 200 groups with half a dozen variables in each group on average in a space with several hundred dimensions depending on the computer system.

2 Sparse Models for Variable Selection

To solve the variable selection efficiently, there is a recent trend of using sparse approximation to solve this problem which has attracted attention in statistics,

machine learning and other disciplines. The earliest work is from [7] called the LASSO. By adding an L1 constraint on the coefficients to (2), the original combinatorial optimization problem is converted to a convex optimization problem as follows

$$\min_{\beta_i} \|\mathbf{y} - \sum_{i=1}^N \mathbf{x}_i \beta_i\|_2^2 \quad (5)$$

$$\text{s.t. } \sum_{i=1}^N |\beta_i| < t. \quad (6)$$

Note that in the LASSO, it fits all explanatory variables in the model but forces some of their coefficients to be zero to implement subset selection. The parameter t is the handle for controlling how many variables are actually included in the model. It is normally reformulated as L1 regularization in practice

$$\min_{\beta_i} \|\mathbf{y} - \sum_{i=1}^N \mathbf{x}_i \beta_i\|_2^2 + \lambda \sum_{i=1}^N |\beta_i|, \quad (7)$$

where λ has the same function as t in (5). The LASSO has several advantages. Firstly, it is a convex optimization problem and hence has a global minimum which can be found by efficient algorithms such as coordinate descent [8], and iterative shrinkage-thresholding [9]. The entire regularization path [10] can also be found by various algorithms such as LARS [11], and GPS [12]. Secondly, it is easy to group variables under this framework. For example, the group LASSO [13] has the group selection property by applying the L1/L2 norm to the group coefficients. The optimization is almost the same as that for the LASSO under certain conditions. Coordinate descent type of search e.g blockwise coordinate descent [14] is still applicable in this case with some limitation. It can also be extended to the online learning setting [15] for a continuously growing training data set.

The group LASSO [13] minimizes the following objective function

$$\min. \frac{1}{2} \|\mathbf{y} - \sum_{g=1}^G \mathbf{X}_g \boldsymbol{\beta}_g\|_2^2 + \lambda \sum_{g=1}^G \sqrt{N_g} \|\boldsymbol{\beta}_g\|_2 \quad (8)$$

where $\mathbf{X}_g$ is the submatrix of group g and $\boldsymbol{\beta}_g$ is the vector of corresponding coefficients. It resembles the normal LASSO in Eq. (7) by using a sparse penalty to implement group selection. It differs in replacing the L1 norm with the L2 norm on the group coefficients to impose group sparsity.

There are two limitations in the original group LASSO implementation. First, the regression coefficients are supposed to be free, i.e. no constraints on them. Second, it is required that the variables within individual group are orthonormal, i.e. $\mathbf{X}_g^T \mathbf{X}_g = \mathbf{I}$ and $\mathbf{I}$ is the identity matrix of compatible size. Under these assumptions, the blockwise coordinate descent for all group coefficients is applicable and conveniently derived from the subgradient of $\boldsymbol{\beta}_g$ which is separable.

However, the assumptions do not always hold. The physics in some real world applications require some constraints such as nonnegativity. Furthermore, it is more appropriate to maintain the original form of the group variables than to orthogonalise them.

To tackle these problems, we propose a fast algorithm in this paper based on the alternating direction multiplier method. Both exact and inexact solutions are provided. The inexact solution simplifies the computation while retaining convergence in practice. As an approximation to the group L0 (GSS), it can be applied to data analysis where group fitting is essential and the coefficients are constrained. It has the potential to be employed as a screening tool to eliminate redundant groups so that GSS can work on a reduced problem to find the optimal solution.

The next section introduces the constrained group LASSO algorithms. It is followed by a section which compares the constrained group LASSO and GSS using spectral unmixing in the setting of multiple PC regression. We summarize the paper in the last section.

3 The Constrained Group LASSO Algorithm

As mentioned earlier, in some applications, it is desirable to constrain the regression coefficients, e.g. nonnegativity. Also it has some fixed variables which must be in the model. For example, a vector of all 1 can be used to model the intercept. More generally, fixed variables are used to model the common features of all data, for instance, the background in spectroscopy. So it takes the following form

$$\begin{aligned} \min. \quad & \frac{1}{2} \|\mathbf{y} - \mathbf{B}\boldsymbol{\alpha} - \sum_{g=1}^G \mathbf{X}_g \boldsymbol{\beta}_g\|_2^2 + \lambda \sum_{g=1}^G \sqrt{N_g} \|\boldsymbol{\beta}_g\|_2 \\ \text{s.t. } & \{\boldsymbol{\alpha}, \boldsymbol{\beta}\} \in \mathcal{C} \end{aligned} \quad (9)$$

where $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_{N_b}]$ is the matrix of given fixed variables, $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_{N_b}]^T$ is the vector of corresponding weights, $\boldsymbol{\beta} = [\boldsymbol{\beta}_1^T, \dots, \boldsymbol{\beta}_G^T]^T$, and $\mathcal{C}$ is some convex set. For example, for the nonnegativity constraint, it is $\boldsymbol{\beta}_g \geq 0$. We call this constrained group LASSO, or CGL for short. Note that requiring $\mathcal{C}$ to be a convex set is to make the whole optimization problem convex. If $\mathcal{C}$ is not a convex set, we must seek nonconvex optimization.

The algorithm in [13] does not handle constraints such as the one in (9). Moreover, it requires the variables in one group to be orthonormal as mentioned before. To handle the convex constraint, we introduce an indicator function attached to the objective function. We put a coordinate descent for coefficients within a group to drop the orthonormal condition which is similar to the work in [16].

We write $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$ the set of all the regression coefficients, and

$$f(\boldsymbol{\theta}) = \frac{1}{2} \|\mathbf{y} - \mathbf{B}\boldsymbol{\alpha} - \sum_{g=1}^G \mathbf{X}_g \boldsymbol{\beta}_g\|_2^2 + \lambda \sum_{g=1}^G \sqrt{N_g} \|\boldsymbol{\beta}_g\|_2$$

to avoid cluttered notation. We minimize the following equivalent target

$$\begin{aligned} \min_{\boldsymbol{\theta}, \mathbf{z}} \quad & f(\boldsymbol{\theta}) + \mathcal{I}(\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{z} = \boldsymbol{\theta} \end{aligned} \quad (10)$$

where $\mathcal{I}(\mathbf{x})$ is an indicator function associated with the convex set $\mathcal{C}$ corresponding to the constraint. It returns 1 if $\mathbf{x} \in \mathcal{C}$ and 0 otherwise. Douglas-Rachford splitting with the proximal operator of (10) gives us the so-called alternating direction multiplier method (ADMM) [17] with a feasible set projection. It is equivalent to minimizing the following

$$\min_{\boldsymbol{\theta}, \mathbf{z}} \quad f(\boldsymbol{\theta}) + \mathcal{I}(\mathbf{z}) + \mathbf{a}^T(\mathbf{z} - \boldsymbol{\theta}) + \frac{\rho}{2} \|\mathbf{z} - \boldsymbol{\theta}\|_2^2$$

where $\mathbf{a}$ is the dual variable from the equality condition in (10) and $\frac{\rho}{2} \|\mathbf{z} - \boldsymbol{\theta}\|_2^2$ is from the augmented Lagrange multiplier method for better stability. It can be shown that this is equivalent to minimizing the final scaled dual variable form

$$\min_{\boldsymbol{\theta}, \mathbf{z}, \mathbf{u}} \quad f(\boldsymbol{\theta}) + \mathcal{I}(\mathbf{z}) + \frac{\rho}{2} \|\mathbf{z} - \boldsymbol{\theta} + \mathbf{u}\|_2^2$$

where $\mathbf{u}$ is the scaled dual variable $\mathbf{a}$ absorbed inside of the quadratic form. We alternatively optimize $\boldsymbol{\theta}$, $\mathbf{z}$, $\mathbf{u}$ with other parameters fixed until the objective function converges. The detailed algorithm is shown in Table 1 and Table 2

Table 1. Constrained Grouped LASSO algorithm. $P_{\mathcal{C}}(\mathbf{x})$ is the Euclidean projection of $\mathbf{x}$ on to the convex set $\mathcal{C}$. For a nonnegativity constraint, it is as simple as setting $\beta_{g_i} = \frac{1 + \text{sign}(\beta_{g_i})}{2} \beta_{g_i}$ meaning if β_{g_i} is negative then set it to be zero and just leave it otherwise.

Constrained grouped LASSO for (9) - exact solution

Input: $\mathbf{y}$, $\mathbf{X}$, $\mathbf{B}$, λ and group information

Output: $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ (regression coefficients)

$$\{\boldsymbol{\alpha}^0, \boldsymbol{\beta}^0\} = \arg \min_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \frac{1}{2} \|\mathbf{y} - \mathbf{B}\boldsymbol{\alpha} - \sum_{g=1}^G \mathbf{X}_g \boldsymbol{\beta}_g\|_2^2$$

$$\mathbf{z}^0 = \mathbf{u}^0 = \boldsymbol{\beta}^0, k = 0.$$

Loop if convergence is not reached

$$1. \quad \boldsymbol{\theta}^{k+1} = \arg \min_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) + \frac{\rho}{2} \|\boldsymbol{\theta} - \mathbf{z}^k + \mathbf{u}^k\|_2^2$$

$$2. \quad \mathbf{z}^{k+1} = P_{\mathcal{C}}(\boldsymbol{\theta}^{k+1} + \mathbf{u}^k)$$

$$3. \quad \mathbf{u}^{k+1} = \mathbf{u}^k + \boldsymbol{\theta}^{k+1} - \mathbf{z}^{k+1}$$

$$4. \quad k = k + 1$$

End loop

In Tables 1 and 2, the skeleton of the algorithm is the alternating direction method. ρ is the augmented Lagrange multiplier which is typically set to 1. It is possible to apply some penalty rules for ρ to accelerate the computation [18].

Table 2. Block coordinate descent for solving step 1 in Table 1. $\mathbf{z}_g$ and $\mathbf{u}_g$ are sub-vectors of $\mathbf{z}$ and $\mathbf{u}$ corresponding to β_g . See the text about Line 6 for details.

Solve step 1 in Table 1 - Block coordinate descent

Input: $\mathbf{y}$, $\mathbf{X}$, $\mathbf{B}$, λ and group information

Output: β , α (regression coefficients)

Loop until β and α converge

1. For $g = 1$ to G

2. if $\left\| \frac{\mathbf{X}_g^T \mathbf{r} + \rho(\mathbf{z}_g^k - \mathbf{u}_g^k)}{\lambda \sqrt{N_g}} \right\|_2 \leq 1$ then $\beta_g^{k+1} = 0$ ($\mathbf{r} = \mathbf{y} - \sum_{s \neq g} \mathbf{X}_s \beta_s^k - \mathbf{B} \alpha^k$)

3. else

4. loop until β_g converge (coordinate descent within β_g)

5. For $i = 1$ to N_g

6. $\beta_{g_i}^{k+1} = \arg \min_{\beta_{g_i}} \frac{1}{2} \|\mathbf{r} - \mathbf{X}_g \beta_g\|_2^2 + \lambda \sqrt{N_g} \|\beta_g\|_2 + \frac{\rho}{2} \|\beta_g - \mathbf{z}_g^k + \mathbf{u}_g^k\|_2^2$

7. End for

8. end loop

9. End for

10. $\alpha^{k+1} = \mathbf{V} \mathbf{D}^{-1} \mathbf{U}^T (\mathbf{y} - \mathbf{X} \beta)$ and ($\mathbf{B} = \mathbf{U} \mathbf{D} \mathbf{V}^T$)

End loop

Line 1 of Table 1 is implemented by block coordinate descent, as detailed in Table 2. Line 2 in Table 2 is group wise soft thresholding resembling the soft threshold operator in [19]. It sets the coefficients of a whole group to 0 if the 2-norm of the rescaled biased correlation between group variables and residuals from fitting other groups is less than 1. It comes from the fact that the subgradient of $\|\beta_g\|_2$ is $\|\beta_g\|_2 \leq 1$ when $\beta_g = 0$. When $\beta_g \neq 0$, we have to estimate each individual coefficient in β_g because $\mathbf{X}_g$ is not necessarily assumed to be orthonormal. It is another coordinate descent process (Line 4 to Line 8 in Table 2) and each element in β_g is searched by a one dimensional optimization, i.e.

$$\beta_{g_i}^{k+1} = \arg \min_{\beta_{g_i}} \frac{1}{2} \|\mathbf{r} - \mathbf{X}_g \beta_g\|_2^2 + \lambda \sqrt{N_g} \|\beta_g\|_2 + \frac{\rho}{2} \|\beta_g - \mathbf{z}_g^k + \mathbf{u}_g^k\|_2^2$$

with some abuse of notation. We minimize β_{g_i} with other elements in β_g fixed and the new version of β_{g_i} will be used to estimate the next ones immediately after it becomes available. So once we have $\beta_{g_1}^{k+1}$ which is estimated by using $\beta_{g_j}^k$ for $j \neq 1$, we use it to optimize the next element in β_g , that is $\beta_{g_2}^{k+1}$ and so on. After we obtain a new β , we update α . Note that in Table 2 Line 10, we use SVD which is a numerically stable solution of $(\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B} (\mathbf{y} - \mathbf{X} \beta)$. What follows is the update of $\mathbf{z}$ by feasible set projection and then an update of $\mathbf{u}$. Updating $\mathbf{z}$ is by Euclidean projection of the estimated coefficients to the feasible set $\mathcal{C}$ which is derived from the proximal operator on the indicator function $\mathcal{I}(\mathbf{x})$. This originates from optimizing $\mathbf{z}$ while fixing other variables.

The algorithm converges very quickly typically. The stopping criterion could be the difference between adjacent updates of the objective function being less than a threshold. Note that for $\lambda = 0$ or some values close to 0, we can simply obtain β by normal least squares replacing Line 1 to Line 9 in Table 2. This algorithm can certainly handle various constraints as long as they are convex. What we need to do is to use a suitable feasible set projection P_C in Table 1 Line 2.

A very interesting observation inspired by the work in [20] is that the solution of β^{k+1} and α^{k+1} in Table 1 Line 1 implemented in Table 2 does not need to be exact. The algorithm in Table 2 is dedicated to finding the exact minimum subject to a small tolerance for numerical implementation. There are two loops there to drive the solution towards the descent direction, one for α and β and the other for β_g in Table 2 Line 4-8. It turns out that both of these loops can be skipped. The result is that the algorithm still converges but more quickly. The revised algorithm is shown in Table 3 which we call the inexact solution for constrained group LASSO.

Table 3. Inexact algorithm for constrained grouped LASSO

Constrained grouped LASSO for (9) - inexact solution

Input: $\mathbf{y}$, $\mathbf{X}$, $\mathbf{B}$, λ and group information

Output: α , β (regression coefficients)

$$\{\alpha^0, \beta^0\} = \arg \min_{\theta} \frac{1}{2} \|\mathbf{y} - \mathbf{B}\alpha - \sum_{g=1}^G \mathbf{X}_g \beta_g\|_2^2$$

$$\mathbf{z}^0 = \mathbf{u}^0 = \beta^0, k = 0.$$

Loop until α and β converge

1. For $g = 1$ to G

2. if $\left\| \frac{\mathbf{X}_g^T \mathbf{r} + \rho(\mathbf{z}_{gi}^k - \mathbf{u}_g^k)}{\lambda \sqrt{N_g}} \right\|_2 \leq 1$ then $\beta_g^{k+1} = 0$ ($\mathbf{r} = \mathbf{y} - \sum_{s \neq g} \mathbf{X}_s \beta_s^k - \mathbf{B}\alpha^k$)

3. else

4. For $i = 1$ to N_g

5. $\beta_{gi}^{k+1} = \arg \min_{\beta_{gi}} \frac{1}{2} \|\mathbf{r} - \mathbf{X}_g \beta_g\|_2^2 + \lambda \sqrt{N_g} \|\beta_g\|_2 + \frac{\rho}{2} \|\beta_g - \mathbf{z}_g^k + \mathbf{u}_g^k\|_2^2$

6. End for

7. End for

8. $\alpha^{k+1} = \mathbf{V} \mathbf{D}^{-1} \mathbf{U}^T (\mathbf{y} - \mathbf{X} \beta^{k+1})$ and $(\mathbf{B} = \mathbf{U} \mathbf{D} \mathbf{V}^T)$

9. $\mathbf{z}^{k+1} = P_C(\beta^{k+1} + \mathbf{u}^k)$

10. $\mathbf{u}^{k+1} = \mathbf{u}^k + \beta^{k+1} - \mathbf{z}^{k+1}$

11. $k = k + 1$

End loop

As we can see from Table 3, there is only one iteration for β^{k+1} and α^{k+1} and the same for β_g^{k+1} . This inexact solution saves several iterations required in the original algorithm. It updates the introduced variable $\mathbf{z}$ and scaled dual variable $\mathbf{u}$ more frequently than the original version does, which ensures that the algorithm is heading towards the right direction and hence convergence. It

makes sense because both the exact and inexact solutions of β^{k+1} and α^{k+1} are only suboptimum along the optimization path. Since they both guarantee to find the optimum, the faster algorithm is preferable in practice.

4 Experimental Results

4.1 Simulation

We performed a simulation to test CGL to see how well it can approach the correct group sparsity. The baseline is provided by GSS. We simply sampled 60 groups of various randomly chosen sizes from 1 to 4 from a standard normal distribution $\mathcal{N}(0, \mathbf{I})$. $\mathbf{y}$ is generated by adding randomly chosen groups with positive weights sampled from a uniform distribution on $[0, 1]$. We added Gaussian noise from $\mathcal{N}(0, \sigma \mathbf{I})$ to $\mathbf{y}$ and varied σ to see the influence of this particular noise. We ran this simulation 1000 times for each value of σ to collect the correct rate. In the noise free case ($\sigma = 0$), CGL performed very well with 97.7% correct rate. CGL has some noise resistance as shown in Table 4 as its correct rate is stable when $\sigma < 0.01$. When the noise level increases, the correct rates of both CGL and GSS degenerate. Interestingly, when $\sigma = 1$, it turns out that CGL is even better than GSS. In this case, $\mathbf{y}$ is almost overwhelmed by noise. Since GSS is RSS driven exhaustive search, it is guaranteed to find the groups with minimum RSS, which is 0 in this simulation. However, when the noise level is high, it deviates from the truth. On the other hand, CGL needs to balance RSS and regularization which provides CGL some robustness.

Table 4. Simulation on CGL and GSS. When $\sigma = 0$, it is noise free

σ	0	0.0001	0.001	0.01	0.1	1
CGL	97.7%	97.2%	97.2%	97.5%	94.5%	61.1%
GSS	100.0%	100.0%	100.0%	100.0%	95.5%	55.8%

4.2 Experiments on Spectral Unmixing

The Data and Model. We applied the fast constrained group LASSO algorithm to a spectral unmixing problem with a reference library [21,22] which is a typical subset selection problem based on linear regression model. Grouped level fitting is of important interest. It can be used in two ways. First is to fit a group of variables (typical spectra of materials) as a bundle since the materials in the group are similar and highly correlated. Second is to apply multiple principal components regression where a group is a bunch of PC's extracted from training samples of some material. Please refer to [23] for details.

We started with a library with 493 spectra from 60 classes. We used the following rules to determine how many PC's were used for each class: a. at least 95% of the variation should be explained by the PC's used; b. If the first PC explains at least 95% of the variation and the first two PC's explain at least 98%

of the variation, then the second PC should be included; c. The total number of PC's for a class should be no more than 2. The purpose of these rules is to include PC's conservatively. We only model a class with two PC's when it is necessary in terms of capturing enough variation. Note that the rules are not optimized yet. We will optimize the number of PC's chosen for each class in our future work. There are 7 fixed variables which must be fitted in the model. So the multiple PC regression model is similar to (3) as

$$\mathbf{y} = \sum_{i=1}^{N_b} \mathbf{b}_i \alpha_i + \sum_{g=1}^M \sum_{i=1}^{N_g} \mathbf{x}_{gi} \beta_{gi} + \epsilon \quad (11)$$

and in this case each group is one or two principal components associated with a class. The task is to find M out of a total of G groups that best fit the above model regarding the overall error described by RSS. Because the first PC is associated with a particular material and therefore its weight has physical meaning, it is natural that the weights for *all first PC's* should be non-negative. So in our experiment the optimization problem becomes

$$\begin{aligned} \min_{\mathcal{S}, \alpha, \beta_g} \quad & \frac{1}{2} \|\mathbf{y} - \mathbf{B}\alpha - \sum_{g \in \mathcal{S}} \mathbf{X}_g \beta_g\|_2^2 \\ \text{s.t.} \quad & \beta_{g_1} \geq 0, |\mathcal{S}| = M. \end{aligned} \quad (12)$$

The test data are from 4 fully characterized evaluation data sets [22]. So we have ground truth for each sample in these data sets to evaluate different methods. The ground truth tells us how many groups are in a certain sample and what they are, but no proportions are given, which is less important than getting the groups correct. This also frees us from model selection. Knowing M for each test sample, we can focus on the group selection which will make the comparison more informative.

The Results. The method we compared with is GSS. We used bisection to choose a suitable λ which retains only M groups in each case for CGL. Note that it is also possible to use variable (group) screening procedures such as the sequential strong rules in [24]. However, it is necessary to work out the right form of the rules in the circumstances with constrained variables.

We applied two methods to the evaluation data sets. Since GSS is exhaustive search which guarantees finding the solution of M groups with minimum RSS and CGL is only an approximation, we should expect that CGL's performance to be bounded by GSS. However, the ground truth is not always the combination of groups with minimum RSS because the mixing process is not fully understood and/or not perfectly modeled.

We summarise the number of incorrectly unmixed samples in Table 5. CGL has comparable performance with GSS on one data sets but is significantly worse on the other three. It is not surprising that GSS outperforms CGL in general because CGL is a convex relaxation of the cardinality minimization or L0 norm

Table 5. Comparison of CGL and GSS on four evaluation data sets

Data set Names.	No. of Samples	Method	No. (%) Incorrect
Dobbyn1	88	GSS	37 (42.0%)
		CGL	65 (77.3%)
GL14	88	GSS	34 (38.6%)
		CGL	44 (50%)
Sasha	81	GSS	15 (18.5%)
		CGL	36 (44.4%)
BEU162	82	GSS	25 (30.5%)
		CGL	24 (29.3%)

in group fashion. Nevertheless, the combinations with the lowest RSS may not be the right answers to the samples for noise or model inadequacy. CGL is slightly better than GSS on the BEU162 data set. This suggests that the noise level may be a bit high for this data set on which a more robust method is preferable.

An advantage of CGL is that it is much faster than GSS given a suitable λ . Based on the comparison in Table 5, CGL may be a fast approximation to give a crude interpretation to the data. It is very useful in the cases where there are too many groups such that GSS is no longer applicable. Furthermore, it is possible that we use CGL as a screening tool to remove groups that can never get into the model [22].

Figure 1 shows the fitting of GSS and CGL for sample No. 53 in the Dobbyn1 data set which is a mixture of Kaolinites, and Gypsum. GSS correctly identified all materials while CGL substituted Gypsum by White Micas. Both fitted curves follow the main features of the original spectrum largely. GSS captured the doublet at 1900nm reasonably well while CGL missed it because of the incorrect substitution of White Micas for Gypsum. As we can see that the visual fits of these two methods are very similar. This reveals that CGL approximates the optimum to some extent. There are quite a lot of cases like this in this

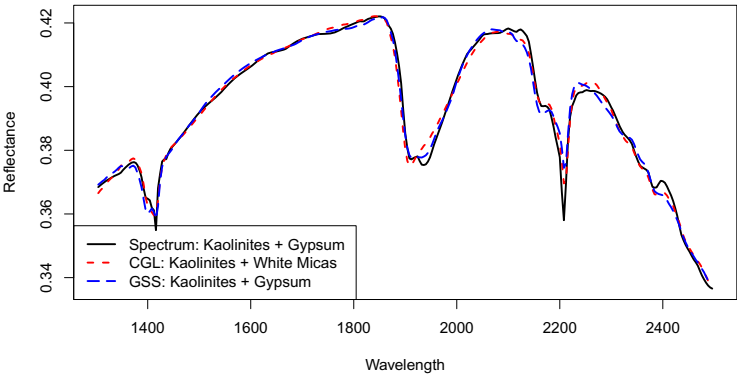


Fig. 1. The fitting of GSS and CGL for sample No. 53 in Dobbyn1 data set

experiment where CGL missed subtle features such as the little doublet which is a compromise between RSS and regularisation.

5 Discussion

As a convex approximation to GSS, constrained group LASSO uses $L1/L2$ norm $\sum_{g=1}^G \sqrt{N_g} \|\beta_g\|_2$ to encourage group sparsity under some convex constraints on regression coefficients. We have proposed the constrained group LASSO (CGL) algorithm in the form of (9) for some real world applications e.g. spectral unmixing. CGL does not require the variables within a group to be orthonormal which simplifies the problem in some applications. The algorithm converges very quickly. The inexact algorithm is even faster while maintaining the same convergence.

Although CGL is only an approximation to the optimum and therefore its performance is inferior to GSS in this sense, an attractive advantage of CGL is that it is much faster than GSS. As an exhaustive search algorithm in essence, GSS is not suitable for large scale problems where CGL is applicable. Another promising method for problems with thousands of groups/variables is to screen groups/variables [25,24] to about 100 or less and then apply GSS to find the accurate solution. CGL, smoothed $L0$ [26] and some other approximation methods may be good candidates.

References

1. Berman, M., Bischof, L., Lagerström, R., Guo, Y., Huntington, J., Mason, P.: An unmixing algorithm based on a large library of shortwave infrared spectra. Technical Report EP117468, CMIS (2011)
2. Khana, J.A., Aelstb, S.V., Zamara, R.H.: Building a robust linear model with forward selection and stepwise procedures. *Computational Statistics And Data Analysis* 52, 239–248 (2007)
3. Hofmann, M., Gatu, C., Kontoghiorghes, E.J.: Efficient algorithms for computing the best subset regression models for large-scale problems. *Computational Statistics and Data Analysis* 52(1), 16–29 (2007)
4. Cho, S.J., Hermsmeier, M.A.: Genetic algorithm guided selection: Variables selection and subset selection. *Journal of Chemical Information and Computer Sciences*, 927–936 (2002)
5. Miller, A.: *Subset Selection in Regression*, 2nd edn. Monographs on Statistics and Applied Probability, vol. 95. Chapman & Hall/CRC (2002)
6. Guo, Y., Berman, M., Gao, J.: Grouped subset selection for least squares regression (in preparation, 2012)
7. Tibshirani, R.: Regression shrinkage and selection via the Lasso. *Journal of Royal Statistical Society* 58(1), 267–288 (1996)
8. Friedman, J.H., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33(1), 1–22 (2010)
9. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* 2(1), 183–202 (2009)

10. Hastie, T., Rosset, S., Tibshirani, R., Zhu, J.: The entire regularization path for the support vector machine. *Journal of Machine Learning Research* 5, 1391–1415 (2004)
11. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. *Annals of Statistics* 32, 407–499 (2004)
12. Friedman, J.: Fast sparse regression and classification. In: *The 23rd International Workshop on Statistical Modelling*, pp. 27–57 (2008)
13. Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68(1), 49–67 (2006)
14. Liu, H., Palatucci, M., Zhang, J.: Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In: *The Proceedings of 26th International Conference of Machine Learning* (2009)
15. Yang, H., Xu, Z., King, I., Lyu, M.R.: Online learning for group lasso. In: *The Proceedings of 27th International Conference of Machine Learning* (2010)
16. Friedman, J., Hastie, T., Tibshirani, R.: A note on the group lasso and a sparse group lasso. Technical report, Stanford University (2010)
17. Boyd, S., Vandenberghe, L.: *Convex Optimization*. Cambridge University Press (2004)
18. Lin, Z., Liu, R., Su, Z.: Linearized alternating direction method with adaptive penalty for low rank representation. In: *NIPS* (2011)
19. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal of Imaging Sciences* 2(1), 183–202 (2009)
20. Lin, Z., Chen, M., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Technical Report UILU-ENG-09-2215, University of Illinois at Urbana-Champaign (2009)
21. Berman, M., Bischof, L., Huntington, J.: Algorithms and software for the automated identification of minerals using field spectra or hyperspectral imagery. In: *Thirteenth International Conference on Applied Geologic Remote Sensing* (1999)
22. Guo, Y., Berman, M.: A comparison between subset selection and L1 regularisation with an application in spectroscopy. *Chemometrics and Intelligent Laboratory Systems* (to appear)
23. Guo, Y., Berman, M.: An investigation of some regularization approaches to the unmixing of mineral reflectance spectra. Technical Report CMIS 09/119, CMIS (2010)
24. Tibshirani, R., Bien, J., Friedman, J., Hastie, T., Simon, N., Taylor, J., Tibshirani, R.J.: Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society, Series B* 74, 245–266 (2012)
25. Fan, J., Lv, J.: Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 70, 849–911 (2008)
26. Mohimani, H., Babaie-Zadeh, M., Jutten, C.: A fast approach for overcomplete sparse decomposition based on smoothed l0 norm. *IEEE Transactions on Signal Processing* 57(1), 289–301 (2009)