


Article

# Parameter Estimation with the Ordered $\ell_2$ Regularization via an Alternating Direction Method of Multipliers

Mahammad Humayoo <sup>1,2,\*</sup>  and Xueqi Cheng <sup>1,2</sup>

<sup>1</sup> CAS Key Laboratory of Network Data Science & Technology, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China; cxq@ict.ac.cn

<sup>2</sup> University of Chinese Academy of Sciences, Beijing 100190, China

\* Correspondence: humayoo@ict.ac.cn

Received: 9 September 2019; Accepted: 8 October 2019; Published: 12 October 2019



**Abstract:** Regularization is a popular technique in machine learning for model estimation and for avoiding overfitting. Prior studies have found that modern ordered regularization can be more effective in handling highly correlated, high-dimensional data than traditional regularization. The reason stems from the fact that the ordered regularization can reject irrelevant variables and yield an accurate estimation of the parameters. How to scale up the ordered regularization problems when facing large-scale training data remains an unanswered question. This paper explores the problem of parameter estimation with the ordered  $\ell_2$ -regularization via Alternating Direction Method of Multipliers (ADMM), called ADMM- $O\ell_2$ . The advantages of ADMM- $O\ell_2$  include (i) scaling up the ordered  $\ell_2$  to a large-scale dataset, (ii) predicting parameters correctly by excluding irrelevant variables automatically, and (iii) having a fast convergence rate. Experimental results on both synthetic data and real data indicate that ADMM- $O\ell_2$  can perform better than or comparable to several state-of-the-art baselines.

**Keywords:** ADMM; big data; feature selection; optimization; ridge regression; ordered regularization; elastic net

## 1. Introduction

In the machine learning literature, one of the most important challenges involves estimating parameters accurately and selecting relevant variables from highly correlated, high-dimensional data. Researchers have noticed many highly correlated features in high-dimensional data [1]. Models often overfit or underfit high-dimensional data because they have a large number of variables but only a few of them are actually relevant; most others are irrelevant or redundant. An underfitting model contributes to estimation bias (i.e., high bias and low variance) in the model fitting because it keeps out relevant variables, whereas an overfitting raises estimation error (i.e., low bias and high variance) since it includes irrelevant variables in the model.

To illustrate an application of our proposed method, consider a study of gene expression data. This is a high-dimensional dataset and contains highly correlated genes. The geneticist always likes to determine which variants/genes contribute to changes in biological phenomena (e.g., increases in blood cholesterol level, etc.) [2]. Therefore, the aim is to explicitly identify all relevant variants. The penalized regularization models such as  $\ell_1$ ,  $\ell_2$ , and so forth have recently become a topic of great interest within machine learning, statistics [1], and optimization [3] communities as classic approaches to estimate parameters. The  $\ell_1$ -based method is not a preferred selection method for groups of variables among which pairwise correlations are significant because the lasso arbitrarily selects a single variable from the group without any consideration of which one to select [4]. Furthermore, if the selected value of

a parameter is too small, the  $\ell_1$ -based method would select many irrelevant variables, thus degrading its performance. On the other hand, a large value of the parameter would yield a large bias [5]. Another point worth noting is that few  $\ell_1$  regularization methods are either adaptive, computationally tractable, or distributed, but no such method contains all three properties together. Therefore, the aim of this study is to develop a model for parameter estimation and to determine relevant variables in highly correlated, high-dimensional data based on the ordered  $\ell_2$ . This model has the following three properties all together: adaptive (Our method is adaptive in the sense that it reduces the cost of including new relevant variables as more variables are added to the model due to rank-based penalization properties.), tractable (A computationally intractable method is a computer algorithm that takes a very long time to execute a mathematical solution. A computationally tractable method is exactly the opposite of intractable method.), and distributed.

Several adaptive and nonadaptive methods have been proposed for parameter estimation and variable selection in large-scale datasets. Different principles are adopted in these procedures to estimate parameters. For example, an adaptive solution (i.e., an ordered  $\ell_1$ ) [5] is a norm and, therefore, convex. Regularization parameters are sorted in non-increasing order in the ordered  $\ell_1$ , in which the ordered regularization penalizes regression coefficients according to their order, with higher orders closer to the top and having larger penalties. Pan et al. [6] proposed a partial sorted  $\ell_p$  norm, which is non-convex and non-smooth. In contrast, the ordered  $\ell_2$  regularization is convex and smooth, just as the standard  $\ell_2$  norm is convex and smooth in Reference [7]. Pan et al. [6] considered p-values between  $0 < p \leq 1$  that do not cover  $\ell_2, \ell_\infty$  norms, and so forth. Pan et al. [6] also did not provide details of other partially sorted norms when  $p \geq 2$  and used random projection and the partial sorted  $\ell_p$  norm to complete the parameter estimation, whereas we have used ADMM with the ordered  $\ell_2$ . A nonadaptive solution (i.e., an elastic net) [8] is a mixture of both ordinary  $\ell_1$  and  $\ell_2$ . In particular, it is a useful model when the number of predictors (p) is much larger than the number of observations (n) or in any situation where the predictor variables are correlated.

Table 1 presents the important properties of the regularizers. As seen in Table 1,  $\ell_2$  and the ordered  $\ell_2$  regularizers are suitable methods for highly correlated, high-dimensional grouping data rather than  $\ell_1$  and the ordered  $\ell_1$  regularizers. The ordered  $\ell_2$  encourages grouping, whereas most of  $\ell_1$ -based methods promote sparsity. Here, grouping signifies a group of strongly correlated variables in high-dimensional data. We used the ordered  $\ell_2$  regularization in our method instead of  $\ell_2$  regularization because the ordered  $\ell_2$  regularization is adaptive. Finally, ADMM has a parallel behavior for solving large-scale convex optimization problems. Our model also employs ADMM and inherits distributed properties of native ADMM [9]. Hence, our model is also distributed. Bogdan et al. [5] did not provide any details about how they applied ADMM in the ordered  $\ell_1$  regularization.

In this paper, we propose “Parameter Estimation with the Ordered  $\ell_2$  Regularization via ADMM” called ADMM- $O\ell_2$  to find relevant parameters from a model.  $\ell_2$  is a ridge regression; similarly, the ordered  $\ell_2$  becomes an ordered ridge regression. The main contribution of this paper is not to present a superior method but rather to introduce a quasi-version of the  $\ell_2$  regularization method and to concurrently raise awareness of the existing methods. As part of this research, we introduced a modern ordered  $\ell_2$  regularization method and proved that the square root of the ordered  $\ell_2$  is a norm and, thus, convex. Therefore, it is also tractable. In addition, the regularization method used an ordered elastic net method to combine the widely used ordered  $\ell_1$  penalty with modern ordered  $\ell_2$  penalty for ridge regression. The ordered elastic net is also proposed by the scholars in this paper. To the best of our knowledge, this is one of the first method to use the ordered  $\ell_2$  regularization with ADMM for parameter estimation and variable selection. In Sections 3 and 4, we explain the integration of ADMM with the ordered  $\ell_2$  and further details about it.

The rest of the paper is arranged as follows. Related works are discussed in Section 2, along with a presentation of the ordered  $\ell_2$  regularization in Section 3. Section 4 describes the application of ADMM to the ordered  $\ell_2$ . Section 5 presents the experiments conducted. Finally, Section 6 closes the paper with a conclusion.

**Table 1.** Properties of the different regularizers: Correlation is a method that shows how strong the relationship between explanatory variables is. In the correlation column, “Yes” means a strong correlation between variables and “No” means weaker (or no) correlation between variables. Stable means that estimates of the  $\ell_2$ -based methods are more stable when the explanatory variables are strongly correlated.

Regularizers	Promoting	Convex	Smooth	Adaptive	Tractable	Correlation	Stable
$\ell_0$ [10,11]	Sparsity	No	No	No	No	No	No
$\ell_1$ [12]	Sparsity	Yes	No	No	Yes	No	No
The Ordered $\ell_1$ [5]	Sparsity	Yes	No	Yes	Yes	No	No
$\ell_2$ [13]	Grouping	Yes	Yes	No	Yes	Yes	Yes
The Ordered $\ell_2$	Grouping	Yes	Yes	Yes	Yes	Yes	Yes
Partial Sorted $\ell_p$ [6]	Sparsity	No	No	Yes	No	No	No

## 2. Related Work

### 2.1. $\ell_1$ and $\ell_2$ Regularization

Deng et al. [14] presented efficient algorithms for group sparse optimization with mixed  $\ell_{2,1}$  regularization for the estimation and reconstruction of signals. Their technique is rooted in a variable splitting strategy and ADMM. Zou and Hastie [8] suggested that an elastic net, a generalization of the lasso, is a linear combination of both  $\ell_1$  and  $\ell_2$  norm. It contributes to sparsity without permitting a coefficient to become too large. However, Candes and Tao [15] introduced a new estimator called the Dantzig selector for a linear model when the parameters are larger than the number of observations for which they established optimal  $\ell_2$  rate properties under a sparsity. Chen et al. [16] enforced sparse embedding to ridge regression, obtaining solutions  $\tilde{x}$  with  $\|\tilde{x} - x^*\|_2 \leq \epsilon \|x^*\|$  small, where  $x^*$  is optimal, and also did this in  $\mathcal{O}(\text{nnz}(A) + n^3/\epsilon^2)$  time, where  $\text{nnz}(A)$  is the number of nonzero entries of  $A$ . Recently, Bogdan et al. [5] have proposed an ordered  $\ell_1$ -regularization technique inspired from a statistical viewpoint, in particular, by a focus on controlling the false discovery rate (FDR) for variable selection in linear regressions. Our proposed method is similar but focused on parameter estimation based on the ordered  $\ell_2$  regularization and ADMM. Several methods have been proposed based on Reference [5] and similar ideas. For example, Bogdan et al. [2] introduced a new model-fitting strategy called Sorted L-One Penalized Estimation (SLOPE), which regularizes least-squares estimates with rank-dependent penalty coefficients. Zeng and Figueiredo [17] proposed DWSL1 as a generalization of octagonal shrinkage and clustering algorithm (OSCAR) that aims to promote feature grouping without previous knowledge of the group structure. Pan et al. [6] have introduced an image restoration method based on a random projection and a partial sorted  $\ell_p$  norm. In this method, an input signal is decomposed into two components: a low-rank component and a sparse component. The low-rank component is approximated by random projection, and the sparse one is recovered by the partial sorted  $\ell_p$  norm. Our method can potentially be used in various other domains such as cyber security [18] and recommendation [19].

### 2.2. ADMM

Researchers have paid a significant amount of attention to ADMM because of its capability of dealing with objective functions independently and simultaneously. Second, ADMM has proved to be a genuine fit in the field of large-scale data-distributed optimization. However, ADMM is not a new algorithm; it was first introduced by References [20,21] in the mid-1970s, with roots as far back as the mid-1950s. In addition, ADMM originated from an augmented method with a Lagrangian multiplier [22]. It became more popular when Boyd et al. [9] published papers about ADMM. The classic ADMM algorithm applies to the following “ADMM-ready” form of problems.

$$\begin{cases} \text{minimize} & f(x) + g(z) \\ \text{s.t.} & Ax + Bz = c \end{cases} \tag{1}$$

The wide range of applications have also inspired the study of the convergence properties of ADMM. Under mild assumptions, ADMM can converge for all choices of the step size. Ghadimi et al. [23] provided some advice on tuning over-relaxed ADMM in the quadratic problems. Deng and Yin [24] have also suggested linear convergence results under the consideration of only a single strongly convex term, given that linear operator’s A and B are full-rank matrices. These convergence results bound error as measured by an approximation to the primal–dual gap. Goldstein et al. [25] created an accelerated version of ADMM that converges more quickly than traditional ADMM under an assumption that both objective functions are strongly convex. Yan and Yin [26] explained, in detail, the different kinds of convergence properties of ADMM and their prerequisite rules for converging. For further studies on ADMM, see Reference [9].

### 3. Ridge Regression with the Ordered $\ell_2$ Regularization

#### 3.1. The Ordered $\ell_2$ Regularization

The proposed parameter estimation and variable selection method in this paper is computationally manageable and adaptive. This procedure depends on the ordered  $\ell_2$  regularization. Let  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_p)$  be a decreasing sequence of positive scalars that satisfy the following condition:

$$\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots \geq \lambda_p \geq 0 \tag{2}$$

The ordered  $\ell_2$  regularization of a vector  $x \in \mathbb{R}^p$  when  $\lambda_1 > 0$  can be defined as follows:

$$J_\lambda(x) = \lambda_1 x_{(1)}^2 + \lambda_2 x_{(2)}^2 + \dots + \lambda_p x_{(p)}^2 = \sum_{k=1}^p \lambda_{BH^{(k)}} x_{(k)}^2 \tag{3}$$

where  $\lambda_{BH^{(k)}}$  is called a BHq method [27], which generates an adaptive and a non-increasing value for  $\lambda$  (Reference [2], Section 1.1). The details of  $\lambda_{BH^{(k)}}$  are available in Section 4.2. For ease of presentation, we have written  $\lambda_k$  in place of  $\lambda_{BH^{(k)}}$  in the rest of the paper.  $x_{(1)}^2 \geq x_{(2)}^2 \geq x_{(3)}^2 \geq \dots \geq x_{(p)}^2$  is the order statistic of the magnitudes of  $x$  [28]. The subscript  $k$  of  $x$  enclosed in parentheses indicates the  $k$ th-order statistic of a sample. Suppose that  $x$  is a sample size of 4. Hence, four numbers are observed in  $x$  if the sample values of  $x = (-2.1, -0.5, 3.2, 7.2)$ . The order statistics of  $x$  would be  $x_{(1)}^2 = 7.2^2, x_{(2)}^2 = 3.2^2, x_{(3)}^2 = 2.1^2, x_{(4)}^2 = 0.5^2$ . The ordered  $\ell_2$  regularization is expressed as the first largest value of  $\lambda$  times the square of the first largest entry of  $x$ , plus the second largest value of  $\lambda$  times the square of the second largest entry of  $x$ , and so on.  $A \in \mathbb{R}^{n \times p}$  and  $b \in \mathbb{R}^n$  are a matrix and a vector, respectively. The ordered  $\ell_2$  regularized loss minimization can be expressed as follows:

$$\min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \frac{1}{2} \{ \lambda_1 x_{(1)}^2 + \dots + \lambda_p x_{(p)}^2 \} \tag{4}$$

**Theorem 1.** *If the square root of  $J_\lambda(x)$  (Equation (3)) is a norm on  $\mathbb{R}^p$  and a function  $\|\cdot\| : \mathbb{R}^p \rightarrow \mathbb{R}$  satisfying the following three properties, then the following Corollaries 1 and 2 are true.*

- i (Positivity)  $\|x\| \geq 0$  for any  $x \in \mathbb{R}^p$  and  $\|x\| = 0$  if and only if  $x = 0$ .
- ii (Homogeneity)  $\|cx\| = |c|\|x\|$  for any  $x \in \mathbb{R}^p$  and  $c \in \mathbb{R}$ .
- iii (Triangle inequality)  $\|x + y\| \leq \|x\| + \|y\|$  for any  $x, y \in \mathbb{R}^p$ .

Note:  $\|x\|$  and  $\|x\|_2$  are used interchangeably.

**Corollary 1.** *When all the  $\lambda_k$ 's take on an equal positive value,  $J_\lambda(x)$  reduces to the square of the usual  $\ell_2$  norm.*

**Corollary 2.** *When  $\lambda_1 > 0$  and  $\lambda_2 = \dots = \lambda_p = 0$ , the square root of  $J_\lambda(x)$  reduces to the  $\ell_\infty$  norm.*

Proofs of the theorem and corollaries are provided in Appendix A. Table 2 shows the notations used in this paper and their meanings.

**Table 2.** Notations and explanations.

Notations	Explanations	Notations	Explanations
Matrix	denoted by uppercase letter	f	loss convex function
Vector	denoted by lowercase letter	g	Regularizer part ( $\ell_1$ or $\ell_2$ etc.)
$\ \cdot\ _1$	$\ell_1$ norm	$\partial f(x)$	subdifferential of convex function f at x
$\ \cdot\ _2$	$\ell_2$ norm	$\partial g(z)$	subdifferential of convex function g at z
$\ x\ _1 = \sum_{k=1}^n  x_k $	$\ell_1$ norm	L1	the $\ell_1$ norm or the lasso
$\ x\ _2 = \sqrt{\sum_{k=1}^n  x_k ^2}$	$\ell_2$ norm	OL1	ordered $\ell_1$ norm or ordered lasso
$\ x\ _2^2 = \sum_{k=1}^n  x_k ^2$	square of $\ell_2$ norm	OL2	ordered $\ell_2$ norm or ordered ridge regression
$J_\lambda(\cdot)$	the ordered norm	Eq.	equation

**Note:** We often use the ordered  $\ell_2$  norm/regularization, OL2, and ADMM-OL2 interchangeably.

### 3.2. The Ordered Ridge Regression

We propose an ordered ridge regression in Equation (5) and call it the ordered ridge regression because we used an ordered  $\ell_2$  regularization with an objective function instead of a standard  $\ell_2$  regularization. The ordered ridge regression is commonly used for parameter estimation and variable selection, particularly when data are strongly correlated and highly dimensional. The ordered ridge regression can be defined as follows:

$$\begin{aligned} & \min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \frac{1}{2} J_\lambda(x) \\ & = \min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \frac{1}{2} \sum_{k=1}^p \lambda_k |x_{(k)}|^2 \end{aligned} \tag{5}$$

where  $x \in \mathbb{R}^p$  denotes an unknown regression coefficient,  $A \in \mathbb{R}^{n \times p}$  ( $p \gg n$ ) is a known matrix,  $b \in \mathbb{R}^n$  represents a response vector, and  $J_\lambda(x)$  is the ordered  $\ell_2$  regularization. The optimal parameter choice for the ordered ridge regression is much more stable than that for a regular lasso; also, it achieves adaptivity in the following senses.

- i For decreasing ( $\lambda_k$ ), each parameter  $\lambda_k$  marks the entry or removal of some variable from the current model (therefore, its coefficient becomes either zero or nonzero); thus, coefficients remain constant in the model. We achieved this by putting some threshold values for  $\lambda_k$  (Reference [5], Section 1.4).
- ii We observed that the price of including new variables declines as more variables are added to the model when the  $\lambda_k$  decreases.

### 4. Applying ADMM to the Ordered Ridge Regression

In order to apply ADMM to the problem in Equation (5), we first transform it into an equivalent form of the problem in Equation (1) by introducing an auxiliary variable  $z$ .

$$\min_{x, z \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \frac{1}{2} J_\lambda(z) \quad \text{s.t.} \quad x - z = 0 \tag{6}$$

We can see that Equation (6) has two blocks of variables (i.e.,  $x$  and  $z$ ). Its objective function is separable in the form of Equation (1) since  $f(x) = \frac{1}{2} \|Ax - b\|_2^2$  and  $g(z) = \frac{1}{2} J_\lambda(z) = \frac{1}{2} \lambda \|z\|_2^2 = \frac{1}{2} \sum_{k=1}^p \lambda_k |x_{(k)}|^2$ , where  $A = I$  and  $B = -I$ . Therefore, ADMM is applicable to Equation (6). An augmented Lagrangian form of Equation (6) can be defined as follows:

$$\begin{aligned} \mathcal{L}_p(x, z, y) &= \frac{1}{2} \|Ax - b\|_2^2 + \frac{1}{2} \lambda \|z\|_2^2 + y^T(x - z) \\ &+ \frac{\rho}{2} \|x - z\|_2^2 \end{aligned} \tag{7}$$

where  $y \in \mathbb{R}^p$  is a Lagrangian multiplier and  $\rho > 0$  denotes a penalty parameter. Next, we apply ADMM to the augmented Lagrangian equation of Equation (7) (Reference [9], Section 3.1), which renders ADMM iterations as follows:

$$\begin{aligned} x^{k+1} &:= \operatorname{argmin}_{x \in \mathbb{R}^p} \mathcal{L}_p(x, z^k, y^k) \\ z^{k+1} &:= \operatorname{argmin}_{z \in \mathbb{R}^p} \mathcal{L}_p(x^{k+1}, z, y^k) \\ y^{k+1} &:= y^k + \rho(x^{k+1} - z^{k+1}) \end{aligned} \tag{8}$$

Proximal gradient methods are well known for solving convex optimization problems for which the objective function is the sum of a smooth loss function and a non-smooth penalty function [9,29,30]. A well-studied example is  $\ell_1$  regularized least squares [1,5]. It should be noted that an ordered  $\ell_1$  norm is convex but not smooth. Therefore, these researchers used a proximal gradient method. In contrast, we have employed an ADMM method because ADMM can solve convex optimization problems for which the objective function is either the sum of a smooth loss function and a non-smooth penalty function or both loss and penalty function are smooth and ADMM also supports parallelism. In the ordered ridge regression, both loss and penalty function are smooth, whereas, in the ordered elastic net, loss function is smooth and penalty function is non-smooth.

#### 4.1. Scaled Form

We can also define ADMM in scaled form by merging a linear and a quadratic term in augmented Lagrangian and then a scaled dual variable, which is shorter and more appropriate. The scaled dual form of ADMM iterations in Equation (8) can be expressed as follows:

$$x^{k+1} := \operatorname{argmin}_{x \in \mathbb{R}^p} (f(x) + (\rho/2)\|x - z^k + u^k\|_2^2) \tag{9a}$$

$$z^{k+1} := \operatorname{argmin}_{z \in \mathbb{R}^p} (g(z) + (\rho/2)\|x^{k+1} - z + u^k\|_2^2) \tag{9b}$$

$$u^{k+1} := u^k + x^k - z^k \tag{9c}$$

where  $u = \frac{1}{\rho}y$  and  $u$  is the scaled dual variable. Next, we can minimize the augmented Lagrangian in Equation (7) with respect to  $x$  and  $z$ , successively. Minimizing Equation (7) with respect to  $x$  becomes the  $x$  subproblem of Equation (9a), and it can be expressed as follows:

$$\begin{aligned} &\min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + (\rho/2)\|x - z^k + u^k\|_2^2 \\ &= \min_{x \in \mathbb{R}^p} \frac{1}{2} \{x^T A^T Ax - 2b^T Ax\} + \frac{\rho}{2} \{x^2 - 2(z^k - u^k)^T x\} \end{aligned} \tag{10a}$$

After computing a derivative of Equation (10a) with respect to  $x$ , then the setting of the derivative of  $x$  becomes equal to zero. Notice that this is a convex problem; therefore, it minimizes to solve the following linear system of Equation (10b):

$$\begin{aligned} &\Leftrightarrow A^T Ax + \rho x - b^T A - \rho(z^k - u^k)^T = 0 \\ &x^{k+1} = (A^T A + \rho * I)^{-1} (A^T b + \rho(z^k - u^k))^T \end{aligned} \tag{10b}$$

Minimizing problem Equation (7) w.r.t.  $z$ , we obtain Equation (9b), and it results in the following  $z$  subproblem:

$$\begin{aligned} &\min_{z \in \mathbb{R}^p} \frac{1}{2} \lambda \|z\|_2^2 + \frac{\rho}{2} \|x^{k+1} + u^k - z\|_2^2 \\ &= \min_{z \in \mathbb{R}^p} \frac{1}{2} \lambda_k z^T z + \frac{\rho}{2} \{(x^{k+1} + u^k - z)^T (x^{k+1} + u^k - z)\} \end{aligned} \tag{11a}$$



After computing a derivative of Equation (11a) with respect to  $z$ , then the setting of the derivative of  $z$  becomes equal to zero. Notice that this is a convex problem; therefore, it minimizes to solve the following linear system of Equation (11b):

$$\Leftrightarrow \frac{1}{2}2\lambda_k z + \frac{\rho}{2}\{2z - 2(x^{k+1} + u^k)^T\} = 0$$

$$z^{k+1} = (\lambda_k + \rho * I)^{-1}\rho(x^{k+1} + u^k) \tag{11b}$$

Finally, the multiplier (i.e., the scaled dual variable  $u$ ) is updated in the following way:

$$u^{k+1} = u^k + (x^k - z^k) \tag{12}$$

Optimality conditions: Primal and dual feasibility are essential and adequate optimality conditions for ADMM in Equation (6) [9]. Dual residual ( $S^{k+1}$ ) and primal residual ( $\gamma^{k+1}$ ) can be defined as follows:

$$\begin{aligned} \text{Dual residual } (S^{k+1}) \text{ at iteration } k + 1 &= \rho(z^k - z^{k+1}) \\ \text{Primal residual } (\gamma^{k+1}) \text{ at iteration } k + 1 &= x^{k+1} - z^{k+1} \end{aligned}$$

Stopping criteria: The stopping criterion for an ordered ridge regression is that primal and dual residuals must be small:

$$\begin{aligned} \|\gamma^k\|_2 &\leq \epsilon^{pri} && \text{where } \epsilon^{pri} = \sqrt{\rho}\epsilon^{abs} + \epsilon^{rel} \max\{\|x^k\|_2, \|z^k\|_2\} \\ \|S^k\|_2 &\leq \epsilon^{dual} && \text{where } \epsilon^{dual} = \sqrt{n}\epsilon^{abs} + \epsilon^{rel} \|\rho * u^k\|_2 \end{aligned}$$

We set  $\epsilon^{abs} = 10^{-4}$  and  $\epsilon^{rel} = 10^{-2}$ . For further details about this choice, see Reference [9], Section 3).

#### 4.2. Over-Relaxed ADMM Algorithm

By comparing Equations (1) and (6), we can write Equation (6) in the over-relaxation form as follows:

$$\alpha x^{k+1} + (1 - \alpha)z^k \quad /* \text{ where } A = I, B = -I \text{ and } c = 0 \text{ in our case */ \tag{13}$$

Substituting  $x^{k+1}$  with Equation (13) into  $z$  of Equation (11b) and  $u$  of Equation (12) updates the results in a relaxation form. Algorithm 1 presents an ADMM iteration for the ordered ridge regression of Equation (6).

---

#### Algorithm 1: Over-relaxed ADMM for the ordered ridge regression

---

- 1: Initialize  $x^0 \in \mathbb{R}^p, z^0 \in \mathbb{R}^p, u^0 \in \mathbb{R}^p, \rho > 0$
  - 2: **while** ( $\|\gamma^k\|_2 \leq \epsilon^{pri}$  &&  $\|S^k\|_2 \leq \epsilon^{dual}$ ) **do**
  - 3:  $x^{k+1} \leftarrow (A^T A + \rho * I)^{-1}(A^T b + \rho(z^k - u^k))^T$
  - 4:  $\lambda_k \leftarrow \text{SortedLambda}(\{\lambda_k\}); \triangleright$  refer to Algorithm 2
  - 5:  $z^{k+1} \leftarrow (\lambda_k + \rho * I)^{-1}\rho(\alpha x^{k+1} + (1 - \alpha)z^k + u^k)$
  - 6:  $u^{k+1} \leftarrow u^k + \alpha(x^{k+1} - z^{k+1}) + (1 - \alpha)(z^k - z^{k+1})$
  - 7: **end while**
- 

We observed that ADMM Algorithm 1 computes an exact solution for each subproblem and that their convergence is guaranteed by existing ADMM theory [24,25,31]. The most important and computationally intensive operation here is matrix inversion in line 3 of Algorithm 1. Here, matrix  $A$  is high-dimensional ( $p \gg n$ ) and  $(A^T A + \rho * I)$  takes  $\mathcal{O}(np^2)$  and its inverse (i.e.,  $(A^T A + \rho * I)^{-1}$ )

takes  $\mathcal{O}(p^3)$ . We compute  $(A^T A + \rho * I)^{-1}$  and  $A^T b$  outside loop; then, we are left with (inverse \*  $(A^T b + \rho(z^k - u^k))^T$ ), which is  $\mathcal{O}(p^2)$ , while addition and subtraction take  $\mathcal{O}(p)$ .  $(A^T A + \rho * I)^{-1}$  is also cacheable, so the complexity is just  $\mathcal{O}(p^3) + k * \mathcal{O}(np^2 + p)$  heuristically with  $k$  number of iteration.

**Generating the ordered parameter  $(\lambda_k)$ :** As mentioned in the beginning, we set out to identify a computationally tractable and adaptive solution. The regularizing sequences play a vital role in achieving this goal. Therefore, we generated adaptive values of  $(\lambda_k)$  such that regressor coefficients are penalized according to their respective order. Our regularizing sequence procedure is motivated by the BHq procedure [27]. The BHq method generates  $(\lambda_k)$  sequences as follows:

$$\lambda_{BH^{(k)}} = \Phi^{-1}\left(1 - q * \frac{k}{2p}\right) \tag{14}$$

$$\lambda_k = \lambda_{BH^{(k)}} \sqrt{1 + \frac{\sum_{j < k} \lambda_{BH^{(j)}}^2}{n - k}} \tag{15}$$

where  $k > 0$ ,  $\Phi^{-1}(\alpha)$  is  $\alpha$ th quantile of a standard normal distribution, and  $q$  is a parameter, namely  $q \in [0; 1]$ . We started with  $\lambda_1 = \lambda_{BH^{(1)}}$  as an initial value of the ordered parameter  $(\lambda_k)$ .

Algorithm 2 presents a method for generating sorted  $(\lambda_k)$ . The difference between lines 5 and 6 in Algorithm 2 is that line 5 is for low-dimensional ( $n \leq p$ ) data and that line 6 is for high-dimensional data ( $p \gg n$ ). Finally, we used the ordered  $(\lambda_k)$  from Algorithm 2 (i.e., the adaptive value of  $(\lambda_k)$ ) in the ordered ridge regression of Equations (6) and (7) instead of ordinary  $\lambda$ . This makes the ordered  $\ell_2$  adaptive and different from standard  $\ell_2$ .

---

**Algorithm 2:** Sorted Lambda  $(\{\lambda_k\})$

---

- 1: Initialize  $q \in [0; 1], k > 0, p, n \in N$
  - 2:  $\lambda_1 \leftarrow \lambda_{BH^{(1)}}; \triangleright \lambda_{BH^{(1)}}$  is from Equation(14) where  $k = 1$
  - 3: **for**  $k \in \{2, \dots, K\}$  **do**
  - 4:  $\lambda_{BH^{(k)}} \leftarrow \Phi^{-1}\left(1 - q * \frac{k}{2p}\right)$
  - 5:  $\lambda_k \leftarrow \lambda_{BH^{(k)}} * \sqrt{1 + \frac{\sum_{j < k} \lambda_{BH^{(j)}}^2}{p - k - 1}}; \triangleright$  when  $n = p$
  - 6:  $\lambda_k \leftarrow \lambda_{BH^{(k)}} * \sqrt{1 + \frac{\sum_{j < k} \lambda_{BH^{(j)}}^2}{2p - k - 1}}; \triangleright$  when  $n = 2p$
  - 7: **end for**
- 

### 4.3. The Ordered Elastic Net

A standard  $\ell_2$  (or an ordered  $\ell_2$ ) regularization is a commonly used tool to estimate parameters for microarray datasets (strongly correlated grouping). However, a key drawback of the  $\ell_2$  regularization is that it cannot automatically select relevant variables because the  $\ell_2$  regularization shrinks coefficient estimates closer but not exactly equal to zero (Reference [12], Chapter 6.2). On the other hand, a standard  $\ell_1$  (or an ordered  $\ell_1$ ) regularization can automatically determine relevant variables due to its sparsity property. However, the  $\ell_1$  regularization also has a limitation. Especially when different variables are highly correlated, the  $\ell_1$  regularization tends to pick only a few of them and to remove the remaining ones—even important ones that might be better predictors. To overcome the limitations of both  $\ell_1$  and  $\ell_2$  regularization, we proposed another method called an ordered elastic net (the ordered  $\ell_{1,2}$  regularization or ADMM- $\mathcal{O}\ell_{1,2}$  or  $-\mathcal{O}\ell_{1,2}$ ), similar to a standard elastic net [8], by combining the ordered  $\ell_2$  regularization with the ordered  $\ell_1$  regularization and the elastic net. By doing so, the ordered  $\ell_{1,2}$  regularization automatically selects relevant variables in a way similar to the ordered  $\ell_1$  regularization. In addition, it can select groups of strongly correlated variables. The key difference between the ordered elastic net and the standard elastic net is a regularization term. We apply



the ordered  $\ell_1$  and  $\ell_2$  regularization in the ordered elastic net instead of the standard  $\ell_1$  and  $\ell_2$  regularization. This approach means that the ordered elastic net inherits the sparsity, grouping, and adaptive properties of the ordered  $\ell_1$  and  $\ell_2$  regularization. We have also employed ADMM to solve the ordered  $\ell_{1,2}$  regularized loss minimization as follows:

$$\Leftrightarrow \min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \alpha \lambda_{BH} \|x\|_1 + \frac{1}{2} (1 - \alpha) \lambda_{BH} \|x\|_2^2$$

For simplicity, let  $\lambda_1 = \alpha \lambda_{BH}$  and  $\lambda_2 = (1 - \alpha) \lambda_{BH}$ . The ordered elastic net becomes

$$\Leftrightarrow \min_{x \in \mathbb{R}^p} \frac{1}{2} \|Ax - b\|_2^2 + \lambda_1 \|x\|_1 + \frac{1}{2} \lambda_2 \|x\|_2^2$$

Now, we can transform the above ordered elastic net equation into an equivalent form of Equation (1) by introducing an auxiliary variable  $z$ .

$$\Leftrightarrow \min_{x,z \in \mathbb{R}^p} \underbrace{\frac{1}{2} \|Ax - b\|_2^2}_{f(x)} + \underbrace{\lambda_1 \|z\|_1 + \frac{1}{2} \lambda_2 \|z\|_2^2}_{g(z)} \text{ s.t. } x - z = 0 \tag{16}$$

We can minimize Equation (16) w.r.t.  $x$  and  $z$  in the same way as we minimized the ordered  $\ell_2$  regularization in Section 4 and Sections 4.1 and 4.2. Therefore, we directly present the final results below without any details. The + sign means to select max (0,value).

$$\begin{aligned} x^{k+1} &= (A^T A + \rho * I)^{-1} (A^T b + \rho (z^k - u^k)) \\ z^{k+1} &= \left( \frac{\rho (x^{k+1} + u^k) - \lambda_1}{\lambda_2 + \rho} \right)_+ - \left( \frac{-\rho (x^{k+1} + u^k) - \lambda_1}{\lambda_2 + \rho} \right)_+ \\ u^{k+1} &= u^k + (x^k - z^k) \end{aligned} \tag{17}$$

### 5. Experiments

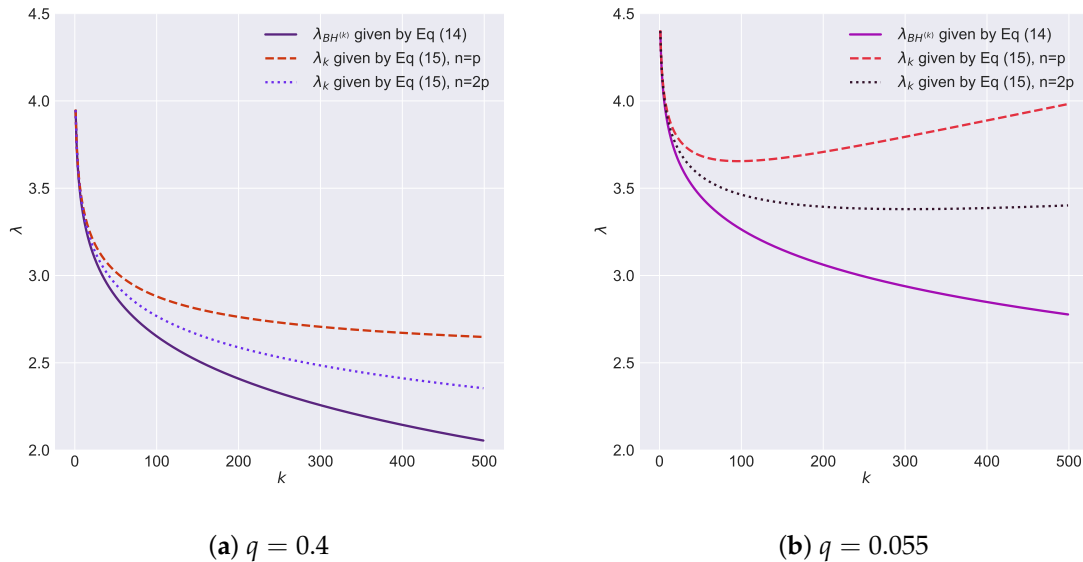
A series of experiments were conducted on both simulated and real data to examine the performance of a proposed method. In this section, first, a concept to select a correct sequence of  $(\lambda_k)$ s is discussed. Second, an experiment on synthetic data is presented that describes the convergence of the lasso, SortedL1, ADMM- $O\ell_2$ , and ADMM- $O\ell_{1,2}$ . Finally, the proposed method is applied to a real feature selection dataset. The performance of the ADMM- $O\ell_{1,2}$  method is analyzed in comparison with state-of-the-art methods: the lasso and SortedL1. These two methods are chosen for comparison because they are very similar to the ADMM- $O\ell_{1,2}$  method except they use the regular lasso and the ordered lasso, respectively, while ADMM- $O\ell_{1,2}$  model employs the ordered  $\ell_{1,2}$  regularization with ADMM.

**Experimental setting:** The algorithms were implemented on Scala Spark™ with Scala code in both distributed and non-distributed versions. A distributed version of experiments was carried out in a cluster of virtual machines with four nodes: one master and three slaves. Each node has 10 GB of memory, 8 cores, CentOS release 6.2, and amd64: core-4.0-noarch. Apache spark™ 1.5.1 was deployed on it. The scholars also used IntelliJ IDEA 15 ULTIMATE as a Scala editor, interactive build tool sbt version 0.13.8, and Scala version 2.10.4. The standalone machine is a Lenovo desktop running Windows 7 Ultimate with an Intel™ Core™ i3 Duo 3.20 GHz CPU and 4 GB of memory. The scholars used MATLAB™ version 8.2.0.701 running on a single machine to draw all figures. The source codes of the lasso, SortedL1, and ADMM- $O\ell_{1,2}$  are available at References [32–34], respectively.

#### 5.1. Adjusting the Regularizing Sequence $(\lambda_k)$ for the Ordered Ridge Regression

Figure 1 was drawn using Algorithm (2), where  $p = 5000$ . As seen in Figure 1, when the value of a parameter ( $q = 0.4$ ) becomes larger, the sequence  $(\lambda_k)$  decreases, while  $(\lambda_k)$  increases for a small

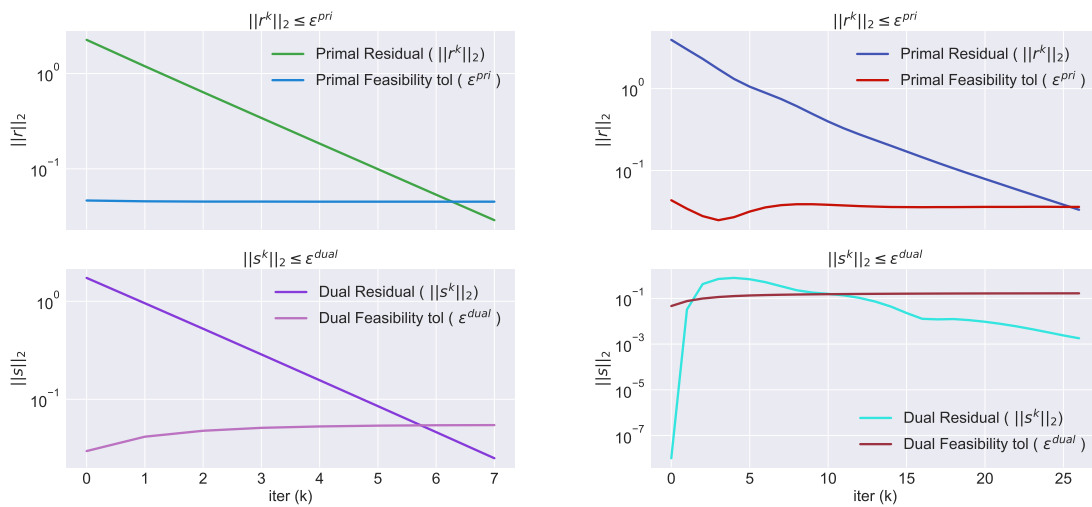
value of  $q = 0.055$ . However, the goal is to obtain a non-increasing order of sequence  $(\lambda_k)$  by adjusting the value of  $q$ , which stimulates convergence. Here, adjusting means tuning the value of the parameter  $q$  using the BHq procedure to yield a suitable sequence  $(\lambda_k)$  such that it improves performance.



**Figure 1.** Illustration of sequence  $\{\lambda_k\}$  for  $p = 5000$ : The solid line of  $\lambda_{BH(k)}$  is given by Equation (14), while the dashed and dotted lines of  $\lambda_k$  are given by Equation (15) for  $n = p$  and  $n = 2p$ , respectively.

5.2. Experimental Results of Synthetic Data

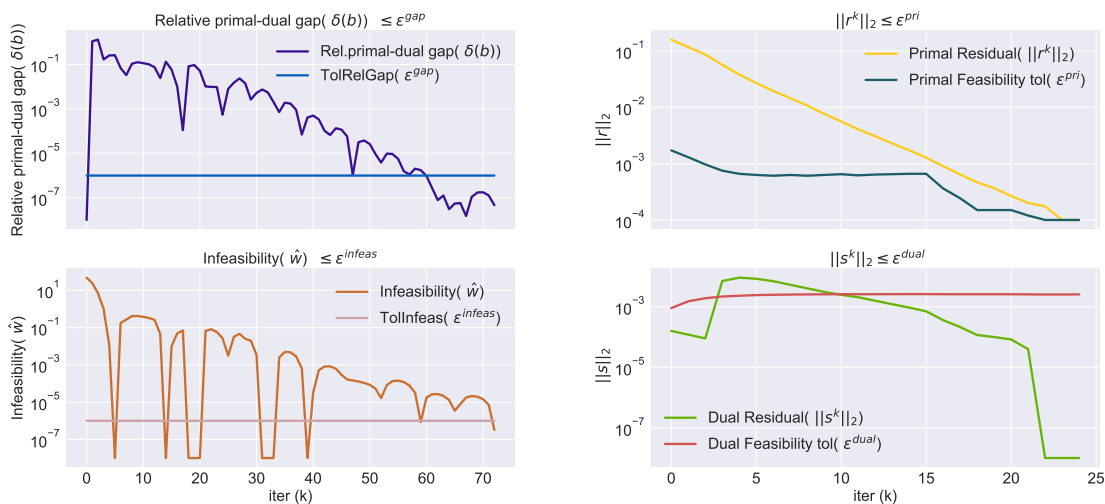
In this section, numerical examples show the convergences of ADMM- $O\ell_{1,2}$ , ADMM- $O\ell_2$ , and other methods. A tiny, dense example of an ordered  $\ell_2$  regularization is examined, where the feature matrix  $A$  has  $n = 1500$  examples and  $p = 5000$  features. Synthetic data is generated as follows: create a matrix  $A$  and choose  $A_{i,j}$  using  $\mathcal{N}(0, 1)$  and then normalize columns of the matrix  $A$  to have the unit  $\ell_2$  norm.  $x^0 \in \mathbb{R}^p$  is generated such that each sampled from  $x^0 \sim \mathcal{N}(0, 0.02)$  is a Gaussian distribution. Label  $b$  is calculated as  $b = A * x^0 + v$ , where  $v \sim \mathcal{N}(0, 10^{-3} * I)$ , which is the Gaussian noise. A penalty parameter  $\rho = 1.0$ , an over-relaxed parameter  $\alpha = 1.0$ , and termination tolerances  $\epsilon^{abs} \leq 10^{-4}$  and  $\epsilon^{rel} \leq 10^{-2}$  are used. Variables  $u^0 \in \mathbb{R}^p$  and  $z^0 \in \mathbb{R}^p$  are initialized to be zero.  $\lambda \in \mathbb{R}^p$  is a non-increasing ordered sequence according to Section 5.1 and Algorithm 2. Figure 2a,b indicates the convergence of ADMM- $O\ell_2$  and ADMM- $O\ell_{1,2}$ , respectively. Figure 3a,b shows the convergence of the ordered  $\ell_1$  regularization and the lasso, respectively. It can be seen from the Figures 2 and 3 that the ordered  $\ell_2$  regularization converges faster than all algorithms. The ordered  $\ell_1$ , lasso, ordered  $\ell_{1,2}$ , and ordered  $\ell_2$  take less than 80, 30, 30, and 10 iterations, respectively to converge. Dual is not guaranteed to be feasible. Therefore, a level of infeasibility of dual is also needed to compute. A numerical experiment terminates whenever both the infeasibility ( $\hat{w}$ ) and relative primal-dual gap ( $\delta(b)$ ) are less than equal to  $\lambda(1)$  (TolInfeas ( $\epsilon^{infeas} = 10^{-6}$ ) and TolRelGap ( $\epsilon^{gap} = 10^{-6}$ ), respectively) the ordered  $\ell_1$  regularization harness synthetic data provided by Reference [5]. The same data is generated for the lasso as for the ordered  $\ell_2$  regularization except for an initial value of  $\lambda$ . For the lasso, set  $\lambda = 0.1 * \lambda_{max}$ , where  $\lambda_{max} = \|A^T * b\|_\infty$ . The researchers also use 10-fold cross-validation (cv) with the lasso. For further details about this step, see Reference [9].



(a) The ordered  $\ell_2$  regularization.

(b) The ordered  $\ell_{1,2}$  regularization.

Figure 2. Primal and dual residual versus primal and dual feasibility, respectively: Input synthetic data.



(a) The ordered  $\ell_1$  regularization.

(b) The lasso.

Figure 3. (a) Relative primal–dual gap versus dual infeasibility, respectively, and (b) primal and dual residual versus primal and dual feasibility, respectively: Input synthetic data.

### 5.3. Experimental Results of Real Data

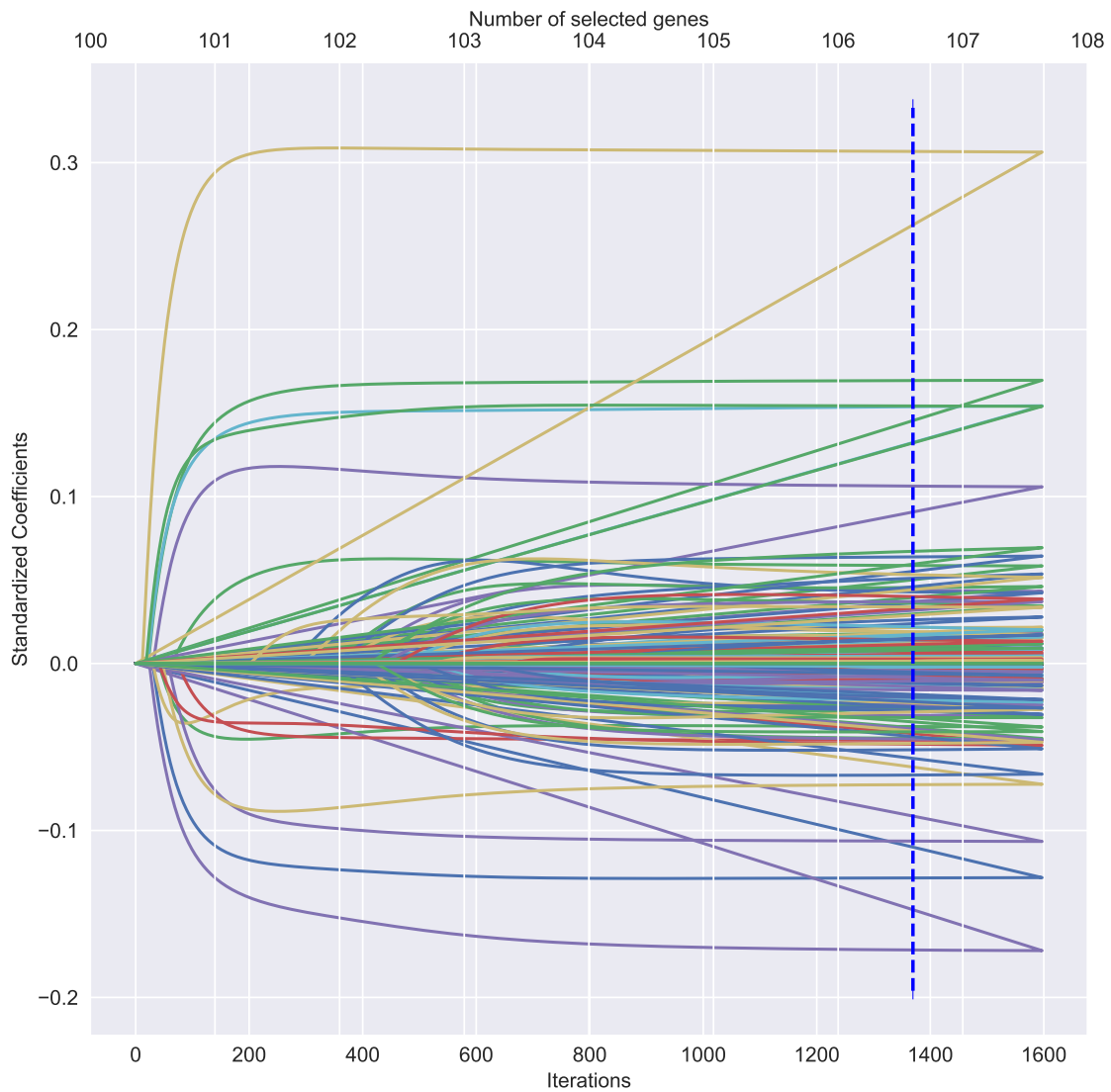
Variable selection difficulty arises when the number of features ( $p$ ) are greater than the number of instances ( $n$ ). The proposed method genuinely handles these types of issues. The practical application of the ADMM method is in many domains such as computer vision and graphics [35], analysis of biological data [36,37], and smart electric grids [38,39]. A biological leukaemia dataset [40] was used to demonstrate the performance of the proposed method. Leukemia is a type of cancer that impairs the body’s ability to build healthy blood cells. Leukemia begins in the bone marrow. There are many types of leukemia, such as acute lymphoblastic leukemia, acute myeloid leukemia, and chronic lymphocytic leukemia. The following two types of leukemia are used in this experiment: acute lymphoblastic leukemia and acute myeloid leukemia. The leukemia dataset consists of 7129 genes and 72 samples [41]. Randomly split the data into training and test sets. In the training set, there are 38 samples, among

which 27 are type I ALL (acute lymphoblastic leukemia) and 11 are type II AML (acute myeloid leukemia). The remaining 34 samples allowed us to test the prediction accuracy. The test set contains 20 type I ALL and 14 type II AML. The data were labeled according to the type of leukemia (ALL or AML). Therefore, before applying an ordered elastic net, the type of leukemia (ALL = -1, or AML = 1) is converted as a (-1, 1) response  $y$ . Predicted response  $\hat{y}$  is set to 1 if  $\hat{y} > 0$ ; otherwise, it is set to -1.  $\lambda \in \mathbb{R}^p$  is a non-increasing, ordered sequence generated according to Section 5.1 and Algorithm 2. For the regular lasso,  $\lambda$  is a single scalar value generated using Equation (14).  $\alpha = 0.1$  is used for the leukemia dataset. All other settings are the same as experiment with synthetic data.

Table 3 illustrates the experiment results of the leukemia dataset for different types of regularization. The lowest average mean square error (MSE) of regularization is the ordered  $\ell_2$ , followed by the ordered  $\ell_{1,2}$  and lasso, while the highest average MSE can be seen in the ordered  $\ell_1$ . Looking at Table 3 first, it is clear that the ordered  $\ell_2$  converges the fastest among all the regularizations. The second fastest converging regularization is the ordered  $\ell_{1,2}$ , while the slowest converging regularization is the ordered  $\ell_1$ . The ordered  $\ell_2$  takes an average iteration around 190 and an average time around 0.15 s to converge. On the other hand, the ordered  $\ell_{1,2}$ , the ordered  $\ell_1$ , and lasso take average iterations around 1381, 10,000, and 10,000, respectively, and average times around: 1.0, 14.0, and 5.0 s, respectively, to converge. It can also be seen from the data in Table 3 that the ordered  $\ell_2$  selected all the variables but that the goal is to select only the relevant variables from strongly correlated, high-dimensional dataset. Therefore, the ordered elastic net was proposed, which only selects relevant variables and discards irrelevant variables. As can be seen from Table 3, average MSE, time, and iteration in the ordered  $\ell_1$  regularization and lasso are significantly more than the ordered  $\ell_{1,2}$  regularization, although an average gene selection in the ordered  $\ell_{1,2}$  regularization is more than that of the ordered  $\ell_1$  regularization and lasso. The ordered  $\ell_1$  and lasso select averages around 84 and 7 variables, respectively, whereas the ordered  $\ell_{1,2}$  selects an average around 107 variables. The lasso performs poorly on the leukemia dataset. The reason for this is that strongly correlated variables are present in the leukemia dataset. In general, the ordered elastic net performs better than the ordered  $\ell_1$  and lasso. Figure 4 shows the ordered elastic net solution paths and the variable selection results.

**Table 3.** Summary of variable selection in leukaemia dataset.

q	Method	Test Error	#Genes	Time	#Iter
0.1	Lasso	2.352941	6	5.459234 s	10,000
	The ordered $\ell_1$	2.235294	56	16.208356 s	10,000
	The ordered $\ell_2$	2.117647	All	0.176351 s	216
	The ordered $\ell_{1,2}$	2.352941	109	1.391347 s	2104
0.2	Lasso	2.352941	6	5.419032 s	10,000
	The ordered $\ell_1$	2.352941	65	14.990179 s	10,000
	The ordered $\ell_2$	2.117647	All	0.167623 s	197
	The ordered $\ell_{1,2}$	2.352941	107	1.046763 s	1597
0.3	Lasso	2.352941	6	5.394828 s	10,000
	The ordered $\ell_1$	2.352941	85	15.477436 s	10,000
	The ordered $\ell_2$	2.117647	All	0.140347 s	185
	The ordered $\ell_{1,2}$	2.117647	108	0.820148 s	1276
0.4	Lasso	2.235294	7	5.470428 s	10,000
	The ordered $\ell_1$	2.352941	90	12.206446 s	10,000
	The ordered $\ell_2$	2.117647	All	0.135451 s	178
	The ordered $\ell_{1,2}$	2.0	107	0.685255 s	1055
0.5	Lasso	2.0	8	5.387423 s	10,000
	The ordered $\ell_1$	2.352941	126	13.226632 s	10,000
	The ordered $\ell_2$	2.117647	All	0.126034 s	172
	The ordered $\ell_{1,2}$	2.0	102	0.603964 s	871
Average	Lasso	2.2588234	6.6	5.426189 s	10,000
	The ordered $\ell_1$	2.3294116	84.4	14.4218098 s	10,000
	The ordered $\ell_2$	2.117647	All	0.1491612 s	189.6
	The ordered $\ell_{1,2}$	2.1647058	106.6	0.9094954 s	1380.6



**Figure 4.** The ordered elastic net coefficients paths: selected genes (the number of nonzero coefficients) are shown on the top of the x-axis and corresponding iterations are shown on the bottom of the x-axis; the optimal ordered elastic net model is given by the fit at an average iteration of 1380.6 with an average selected gene of 106.6 (indicated by a dotted line). Input leukemia data.

### 6. Conclusions

In this paper, we showed a method for optimizing an ordered  $\ell_2$  problem under an ADMM framework, called ADMM- $O\ell_2$ . As an implementation of ADMM- $O\ell_2$ , the ridge regression with the ordered  $\ell_2$  regularization is shown. We also presented a method for variable selection, called ADMM- $O\ell_{1,2}$  which employs the ordered  $\ell_1$  and  $\ell_2$ . We see the ordered  $\ell_{1,2}$  as a generalization of the ordered  $\ell_1$ , which is shown as an important tool for model fitting, feature selection, and parameter estimation.

Experimental results show that the ADMM- $O\ell_{1,2}$  method correctly estimates parameter, selects relevant variables, and excludes irrelevant variables for microarray data. Our method is also computationally tractable, adaptive, and distributed. The ordered  $\ell_2$  regularization is convex and can be optimized efficiently with faster convergence rate. Additionally, we have shown that our algorithm has complexity  $\mathcal{O}(p^3) + k * \mathcal{O}(np^2 + p)$  heuristically, where k is the number of iterations. In future work, we plan to apply our method to other regularization models with complex penalties.

**Author Contributions:** Conceptualization, M.H.; methodology, M.H.; investigation, M.H.; writing—original draft preparation, M.H.; writing—review and editing, M.H. and X.C.; visualization, M.H.; supervision, X.C.

**Funding:** The work is funded by the National Key R&D Program of China under Grant No. 2016QY02D0405 and 973 Program of China under Grant No. 2014CB340401. Mahammad Humayoo is supported by the CAS-TWAS fellowship.

**Acknowledgments:** We gratefully acknowledge the useful comments of the anonymous referees. We would also like to thank the editor for his continuous and quick support during the peer review process. We would also like to thank all students and teachers who supported us in this work. We discussed some issues with users in the online communities. Finally, we want to extend a big thank you to all the users of that online communities who participate in discussion and provide us valuable suggestion.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

### Proof. Positivity of Theorem 1

$$J_{\lambda}(x) = \lambda \|x\|_2^2 = \sum_{k=1}^p \lambda_k x_{(k)}^2 = \sum_{k=1}^p (\sqrt{\lambda_k} x_{(k)})^2$$

Take the square root on both side for the following:

$$\|\sqrt{\lambda}x\|_2 = \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k} x_{(k)})^2}$$

For  $\forall x \in \mathbb{R}^p$ , hold Equation (2); thus,  $(\lambda_{1..p})$  is positive and  $(\lambda_{1..p}) \neq 0$ .  $\|x\|_2$  will never be negative because of the square of  $x$ .

$$\|\sqrt{\lambda}x\|_2 = \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k} x_{(k)})^2} \geq 0$$

If  $x = 0$ , then we have the following:

$$\|\sqrt{\lambda}x\|_2 = \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k} \cdot 0)^2} = 0 \tag{A1}$$

If  $\|x\|_2 = 0$ , then we have the following:

$$\|\sqrt{\lambda}x\|_2 = \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k} x_{(k)})^2} = 0 \Rightarrow x_{(k)} = 0 \forall k \Rightarrow x = 0$$

Since  $(\lambda_k) \neq 0$ ,  $\|x\|_2$  will be only zero if and only if  $x = 0$ .  $\square$

### Proof. Homogeneity of Theorem 1

The first two steps are the same as that of the proof of positivity of Theorem 1; then, we have the following:

$$\begin{aligned} \|\sqrt{\lambda}x\|_2 &= \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k})^2 x_{(k)}^2} = |\sqrt{\lambda}| \sqrt{\sum_{k=1}^p x_{(k)}^2} \\ \|\sqrt{\lambda}x\|_2 &= |\sqrt{\lambda}| \|x\|_2 \end{aligned}$$

where  $(c = \sqrt{\lambda})$ . Then, we have the following:

$$\|cx\|_2 = |c| \|x\|_2$$

$\square$

### Proof. Triangle inequality of Theorem 1



The first two steps are the same as that of the proof of positivity of Theorem 1. Now, with  $x + y$  in place of  $x$ , we have the following:

$$\begin{aligned} \|\sqrt{\lambda}(x + y)\|_2 &= \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k}x_{(k)} + \sqrt{\lambda_k}y_{(k)})^2} \\ \|\sqrt{\lambda}(x + y)\|_2 &= \sqrt{\sum_{k=1}^p (\sqrt{\lambda_k}x_{(k)})^2 + \sum_{k=1}^p (\sqrt{\lambda_k}y_{(k)})^2 + 2\sum_{k=1}^p \lambda_k x_{(k)}y_{(k)}} \end{aligned}$$

From Cauchy–Schwarz inequality,  $x \cdot y \leq \|x\| \cdot \|y\|$ , we have the following:

$$\begin{aligned} \|\sqrt{\lambda}x + \sqrt{\lambda}y\|_2 &\leq \sqrt{\|\sqrt{\lambda}x\|_2^2 + \|\sqrt{\lambda}y\|_2^2 + 2\|\sqrt{\lambda}x\|_2 \cdot \|\sqrt{\lambda}y\|_2} \\ \|\sqrt{\lambda}x + \sqrt{\lambda}y\|_2 &\leq \sqrt{(\|\sqrt{\lambda}x\|_2 + \|\sqrt{\lambda}y\|_2)^2} \\ \|\sqrt{\lambda}x + \sqrt{\lambda}y\|_2 &\leq (\|\sqrt{\lambda}x\|_2 + \|\sqrt{\lambda}y\|_2) \end{aligned}$$

where  $x = \sqrt{\lambda}x$  and  $y = \sqrt{\lambda}y$ . Then, we have the following:

$$\|x + y\| \leq \|x\| + \|y\|$$

□

**Proof. Corollary 1**

From Equation (3):

$$J_\lambda(x) = \lambda_1 x_{(1)}^2 + \lambda_2 x_{(2)}^2 + \dots + \lambda_p x_{(p)}^2$$

All  $\lambda_k$  take on an equal positive value, i.e.,  $\lambda_1 = \lambda_2 = \dots = \lambda_p$

$$J_\lambda(x) = \lambda x_{(1)}^2 + \lambda x_{(2)}^2 + \dots + \lambda x_{(p)}^2$$

$$J_\lambda(x) = \lambda(x_{(1)}^2 + x_{(2)}^2 + \dots + x_{(p)}^2)$$

$$J_\lambda(x) = \lambda \sum_{k=1}^p x_{(k)}^2$$

$$J_\lambda(x) = \lambda \|x\|_2^2$$

where  $\lambda$  is positive scalar. □

**Proof. Corollary 2**

The first two steps are the same as that of the proof of positivity of Theorem 1. When  $\lambda_2 = \dots = \lambda_p = 0$ , we get the first term only and  $p = 1$ . The remaining terms are zero when  $p > 1$ .

$$\begin{aligned} \|\sqrt{\lambda}x\|_2 &= \sqrt{\sum_{k=1}^{p=1} (\sqrt{\lambda_k}x_{(k)})^2} \\ \|\sqrt{\lambda}x\|_2 &= \sqrt{(\sqrt{\lambda_1}x_{(1)})^2} \\ \|\sqrt{\lambda}x\|_2 &= |\sqrt{\lambda_1}x_{(1)}| \end{aligned}$$

where  $x_1 = \sqrt{\lambda_1}x_{(1)}$  and  $\|x\|_\infty = \max|x_k|$ . Then, we have the following:

$$\|x\|_2 = \|x\|_\infty$$

□

**References**

1. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B (Methodol.)* **1996**, *58*, 267–288. [CrossRef]
2. Bogdan, M.; van den Berg, E.; Sabatti, C.; Su, W.; Candès, E.J. SLOPE—Adaptive variable selection via convex optimization. *Ann. Appl. Stat.* **2015**, *9*, 1103. [CrossRef] [PubMed]

3. Bach, F.; Jenatton, R.; Mairal, J.; Obozinski, G. Optimization with sparsity-inducing penalties. *Found. Trends<sup>®</sup> Mach. Learn.* **2012**, *4*, 1–106. [[CrossRef](#)]
4. Efron, B.; Hastie, T.; Johnstone, I.; Tibshirani, R. Least angle regression. *Ann. Stat.* **2004**, *32*, 407–499.
5. Bogdan, M.; van den Berg, E.; Su, W.; Candès, E.J. *Statistical Estimation and Testing via the Ordered L1 Norm*; Stanford University: Palo Alto, CA, USA, 2013.
6. Pan, H.; Jing, Z.; Li, M. Robust image restoration via random projection and partial sorted  $\ell_p$  norm. *Neurocomputing* **2017**, *222*, 72–80. [[CrossRef](#)]
7. Azghani, M.; Kosmas, P.; Marvasti, F. Fast Microwave Medical Imaging Based on Iterative Smoothed Adaptive Thresholding. *IEEE Antennas Wirel. Propag. Lett.* **2015**, *14*, 438–441. [[CrossRef](#)]
8. Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2005**, *67*, 301–320. [[CrossRef](#)]
9. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends<sup>®</sup> Mach. Learn.* **2011**, *3*, 1–122. [[CrossRef](#)]
10. Daducci, A.; Van De Ville, D.; Thiran, J.P.; Wiaux, Y. Sparse regularization for fiber ODF reconstruction: From the suboptimality of  $\ell_2$  and  $\ell_1$  priors to  $\ell_0$ . *Med. Image Anal.* **2014**, *18*, 820–833. [[CrossRef](#)]
11. Gong, P.; Zhang, C.; Lu, Z.; Huang, J.; Ye, J. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In Proceedings of the International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 37–45.
12. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 112.
13. Wang, L.; Zhu, J.; Zou, H. Hybrid huberized support vector machines for microarray classification and gene selection. *Bioinformatics* **2008**, *24*, 412–419. [[CrossRef](#)]
14. Deng, W.; Yin, W.; Zhang, Y. Group sparse optimization by alternating direction method. In *SPIE Optical Engineering+ Applications*; International Society for Optics and Photonics: Bellingham, WA, USA, 2013; p. 88580R.
15. Candès, E.; Tao, T. The Dantzig selector: Statistical estimation when  $p$  is much larger than  $n$ . *Ann. Stat.* **2007**, *35*, 2313–2351. [[CrossRef](#)]
16. Chen, S.; Liu, Y.; Lyu, M.R.; King, I.; Zhang, S. *Fast Relative-Error Approximation Algorithm for Ridge Regression*; UAI: Amsterdam, The Netherlands, 2015; pp. 201–210.
17. Zeng, X.; Figueiredo, M.A. Decreasing Weighted Sorted L1 Regularization. *IEEE Signal Process. Lett.* **2014**, *21*, 1240–1244. [[CrossRef](#)]
18. Albanese, M.; Erbacher, R.F.; Jajodia, S.; Molinaro, C.; Persia, F.; Picariello, A.; Sperli, G.; Subrahmanian, V. Recognizing unexplained behavior in network traffic. In *Network Science and Cybersecurity*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 39–62.
19. Amato, F.; Moscato, V.; Picariello, A.; Sperli, G. Recommendation in social media networks. In Proceedings of the 2017 IEEE Third International Conference on Multimedia Big Data (BigMM), Laguna Hills, CA, USA, 19–21 April 2017; pp. 213–216.
20. Glowinski, R.; Marroco, A. Sur l’approximation, par éléments finis d’ordre un, et la résolution, par pénalisation-dualité d’une classe de problèmes de Dirichlet non linéaires. *Revue Française d’Automatique Informatique Recherche Opérationnelle Analyse Numérique* **1975**, *9*, 41–76. [[CrossRef](#)]
21. Gabay, D.; Mercier, B. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Comput. Math. Appl.* **1976**, *2*, 17–40. [[CrossRef](#)]
22. Hestenes, M.R. Multiplier and gradient methods. *J. Optim. Theory Appl.* **1969**, *4*, 303–320. [[CrossRef](#)]
23. Ghadimi, E.; Teixeira, A.; Shames, I.; Johansson, M. Optimal parameter selection for the alternating direction method of multipliers (ADMM): Quadratic problems. *IEEE Trans. Autom. Control* **2015**, *60*, 644–658. [[CrossRef](#)]
24. Deng, W.; Yin, W. On the global and linear convergence of the generalized alternating direction method of multipliers. *J. Sci. Comput.* **2016**, *66*, 889–916. [[CrossRef](#)]
25. Goldstein, T.; O’Donoghue, B.; Setzer, S.; Baraniuk, R. Fast alternating direction optimization methods. *SIAM J. Imaging Sci.* **2014**, *7*, 1588–1623. [[CrossRef](#)]
26. Yan, M.; Yin, W. Self equivalence of the alternating direction method of multipliers. In *Splitting Methods in Communication, Imaging, Science, and Engineering*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 165–194.
27. Benjamini, Y.; Hochberg, Y. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B (Methodol.)* **1995**, *57*, 289–300. [[CrossRef](#)]

28. David, H.A.; Nagaraja, H.N. *Order Statistics*; John Wiley & Sons, Inc.: New York, NY, USA, 2003.
29. Schmidt, M.; Roux, N.L.; Bach, F.R. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2011; pp. 1458–1466.
30. Parikh, N.; Boyd, S. Proximal algorithms. *Found. Trends<sup>®</sup> Optim.* **2014**, *1*, 127–239. [[CrossRef](#)]
31. Glowinski, R. *Lectures on Numerical Methods for Non-Linear Variational Problems*; Springer: Berlin/Heidelberg, Germany, 2008.
32. Boyd, S. Lasso Solve Lasso Problem via ADMM. 2011. Available online: <https://web.stanford.edu/~boyd/papers/admm/lasso/lasso.html> (accessed on 12 October 2019).
33. Bogdan, M. Sorted L-One Penalized Estimation. 2015. Available online: <https://statweb.stanford.edu/~candes/SortedL1/software.html> (accessed on 12 October 2019).
34. Humayoo, M. ADMM Ordered L2. 2019. Available online: <https://github.com/ADMMOL2/ADMMOL2> (accessed on 12 October 2019).
35. Liu, J.; Musialski, P.; Wonka, P.; Ye, J. Tensor completion for estimating missing values in visual data. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 208–220. [[CrossRef](#)] [[PubMed](#)]
36. Bien, J.; Taylor, J.; Tibshirani, R. A lasso for hierarchical interactions. *Ann. Stat.* **2013**, *41*, 1111. [[CrossRef](#)] [[PubMed](#)]
37. Danaher, P.; Wang, P.; Witten, D.M. The joint graphical lasso for inverse covariance estimation across multiple classes. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2014**, *76*, 373–397. [[CrossRef](#)] [[PubMed](#)]
38. Kraning, M.; Chu, E.; Lavaei, J.; Boyd, S. Dynamic network energy management via proximal message passing. *Found. Trends<sup>®</sup> Optim.* **2014**, *1*, 73–126. [[CrossRef](#)]
39. Kekatos, V.; Giannakis, G.B. Distributed robust power system state estimation. *IEEE Trans. Power Syst.* **2012**, *28*, 1617–1626. [[CrossRef](#)]
40. Chih-Jen, L. Feature Datasets. 2017. Available online: <http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/> (accessed on 12 October 2019).
41. Golub, T.R.; Slonim, D.K.; Tamayo, P.; Huard, C.; Gaasenbeek, M.; Mesirov, J.P.; Coller, H.A.; Loh, M.L.; Downing, J.R.; Caligiuri, M.A.; et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* **1999**, *286*, 531–537. [[CrossRef](#)] [[PubMed](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).