Proximal Weighted L^1 and L^0 Method with Global Convergence

Anonymous authors
Paper under double-blind review

Abstract

This paper develops a joint weighted L^1- and L^0-norm (WL1L0) regularization method by leveraging proximal operators and translation mapping techniques to mitigate the bias introduced by the L^1-norm in applications to high-dimensional data. A weighting parameter α is incorporated to control the influence of both regularizers. Our broadly applicable model is nonconvex and nonsmooth, but we prove global convergence for the alternating direction method of multipliers (ADMM), Peaceman–Rachford splitting method (PRSM) and strictly contractive Peaceman–Rachford splitting method (SCPRSM). Moreover, we evaluate the effectiveness of our model on both simulated and real high-dimensional genomic datasets by comparing with adaptive versions of the LASSO, smoothly clipped absolute deviation (SCAD) and minimax concave penalty (MCP). The results show that WL1L0 outperforms the LASSO, SCAD, and MCP by consistently achieving the lowest mean squared error (MSE) across all datasets, indicating its superior ability to handling large high-dimensional data.

1 Introduction

High-dimensional statistics is a rapidly growing field of research that deals with statistical analysis in the presence of a large number of variables (predictors), often much larger than the sample size. For example, high-throughput measurements in genomics contain thousands or millions of variables, such as single nucleotide polymorphism (SNP) markers and gene expression data for each individual. In such settings, traditional statistical methods often fail due to issues like overfitting, multicollinearity and computational complexity. In recent years, a number of regularization methods have been developed that impose a penalty on the size of the regression coefficients, which encourages sparsity and reduces the number of variables in the model (Fan et al., 2011; Fan & Lv, 2010; Heinze et al., 2018). Sparse learning techniques are essential in analyzing high-dimensional data for increased prediction accuracy, reduced computational complexity and enhanced interpretability of the results (Bühlmann & Van De Geer, 2011; Giraud, 2015; Wainwright, 2019).

Among various sparsity-inducing methods, the L^1 regularizer (also known as LASSO) stands out for its convex nature and computational efficiency (Tibshirani, 1996). It adds a penalty term to the loss function proportional to the absolute value of the coefficients (L^1-norm), which tends to shrink the coefficients towards zero and force some coefficients exactly zero. Shrinking these coefficients helps to avoid overfitting, which can happen when a model memorizes the training data too well and does not perform well on new data. By reducing overfitting, LASSO can lead to more accurate predictions on unseen data. However, when coefficients are being shrunk, LASSO tends to favor keeping larger coefficients over smaller ones. This can lead to a bias towards larger coefficients in the model estimation process. In cases where variables are highly correlated, LASSO may randomly select variables. In cases where variables are highly correlated, LASSO may randomly select variables. For example, when two or more variables are highly correlated due to high-dimensionality, LASSO may choose one variable over another. In the specific context of genomic data, where the goal is often to identify genes associated with certain traits or diseases, this inaccurate selection can lead to the inclusion of incorrect genes in the model. This also poses a risk in terms of prediction, as the estimated coefficients of the selected genes contribute to predicting the outcome of interest (Fan

et al., 2014b; Fan & Li, 2001; Johnstone & Titterington, 2009; Toloşi & Lengauer, 2011). Furthermore, after shrinkage, very small coefficient values can be produced, making them difficult to interpret and offering minimal information. Hence, LASSO requires the fulfillment of the irrepresentable condition to obtain valid estimations (Zhao & Yu, 2006). In cases where the underlying datasets fail to meet this condition, the LASSO method may not accurately select the appropriate variables, leading to incorrect discoveries and wrong conclusions. In practice, implementing the irrepresentable condition can be challenging. Studies show that nonconvex regularizers such as SCAD and MCP reduce bias and have better prediction properties than the L^1 regularizer (Bertsimas et al., 2020; Fan & Li, 2001; Zhang, 2010).

On the other hand, L^0 regularization, which is also known as best subset selection (Hocking & Leslie, 1967), directly penalizes the number of non-zero coefficients in the model. It encourages sparsity, meaning it tends to produce models with fewer non-zero coefficients without any shrinkage. This results in a model that only includes the most relevant variables, simplifying the model and potentially improving its predictive performance by reducing overfitting. However, finding the optimal subset of variables using L^0-norm is an NP-hard problem, meaning the computational cost grows exponentially with the number of variables (Natarajan, 1995). While L^1 regularization is commonly used because of its convex nature, the L^0-norm is computationally expensive and often intractable, and hence not frequently used on large data sets (Hastie et al., 2020).

In this paper, we propose combining L^1 and L^0 regularization. We now revise a regression model

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{b} + \boldsymbol{\epsilon}, \tag{1}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is the response vector, $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ is the predictor matrix, $\boldsymbol{b} \in \mathbb{R}^p$ is the vector of regression coefficients, and $\boldsymbol{\epsilon} \in \mathbb{R}^n$ is a noise (error) vector. For a vector $\boldsymbol{b}$, we write the q-norm notation as

$$||\boldsymbol{b}||_q = \begin{cases} \sum_i \mathbf{1}\left(b_i \neq 0\right), & \text{if } q = 0, \\ \left(\sum_i |b_i|^q\right)^{1/q}, & \text{if } 0 < q < \infty, \\ \max_i |b_i|, & \text{if } q = \infty. \end{cases}$$

Here, the $||\boldsymbol{b}||_0$ is the L^0-norm that is the number of nonzero elements in $\boldsymbol{b}$. It is noteworthy that the L^0-norm does not meet the criteria of a norm, specifically lacking the homogeneity property (Beck, 2017). Despite this, the term is widely used in the literature, and for the sake of consistency, we will retain its adoption.

To achieve higher sparsity, one can use an L^1 regularizer with a constrained best-subsets estimator that has fewer nonzero coefficients than the LASSO as

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda||\boldsymbol{b}||_1 \quad \text{s.t.} \quad ||\boldsymbol{b}||_0 \leq s, \tag{2}$$

where $\hat{\boldsymbol{b}}$ represents the estimate of the vector of regression coefficients, $\lambda > 0$ is the regularization parameter, and s is the desired level of sparsity (i.e., the maximum number of nonzero coefficients). The optimization problem (2) can be expressed in the Lagrangian form as

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda_1||\boldsymbol{b}||_1 + \lambda_2||\boldsymbol{b}||_0. \tag{3}$$

We extend (3) by introducing a weight paremeter $\alpha \in (0, 1)$ and employing a common regularization parameter λ as

$$\text{WL1L0: } \hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda\left(\alpha||\boldsymbol{b}||_1 + (1-\alpha)||\boldsymbol{b}||_0\right). \tag{4}$$

Hence, we propose WL1L0 (4), which aims to combine the benefits of both L^1 and L^0 regularization using a common regularization parameter λ and a weight parameter α that determines the relative importance of both L^1 and L^0 regularization.

Later in our main work, we split $\boldsymbol{b}$ into the sparse components $\boldsymbol{c}$ and $\boldsymbol{d}$ using L^1 and L^0 regularization, respectively. By using L^1 regularization, we encourage $\boldsymbol{c}$ to be sparse, meaning many entries in $\boldsymbol{c}$ will be zero. By using L^0 regularization, we ensure that $\boldsymbol{d}$ is sparse, meaning it has a small number of non-zero

entries. Here, L^0 debiases the LASSO, and since we first fit the LASSO, the computation of L^0 is now feasible.

Our primary contribution lies in developing a model that mitigates the bias introduced by regularization methods in high-dimensional data, specifically for prediction tasks. This is achieved through a novel integration of the proximal operator and translation mapping. Our proposed model (4) is nonconvex and nonsmooth. Nowadays, nonconvex nonsmooth problems arise in various practical applications across numerous fields including statistical genetics, signal processing, and control theory. Analyzing convergence for these types of problems ensures that optimization algorithms can be applied effectively to a wide range of real-world scenarios. Therefore, we provide convergence proofs for ADMM, PRSM, and SCPRSM when applied to our model. We also implement adaptive versions of all LASSO, SCAD and MCP using the ADMM algorithm. Finally, we assess the efficacy of our proposed method through comprehensive evaluations on simulated and real-world datasets.

2 Related Work

In the rapidly evolving landscape of technology and data, prediction has become a cornerstone for making informed decisions across various domains. Regularization techniques are pivotal in enhancing the performance and generalizability of predictive models, particularly when dealing with complex datasets and high-dimensional data. By imposing penalties on the model parameters, regularization helps prevent overfitting, ensuring that the model captures the underlying patterns in the data. In this section, we will review key related works on regularization methods, highlighting significant advancements and methodologies.

Ridge regression, also known as Tikhonov regularization was introduced by Hoerl & Kennard (1970), uses an L^2 penalty term that shrinks all the coefficients and reduces their magnitudes. The ridge regression can be formulated as

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda ||\boldsymbol{b}||_2^2. \tag{5}$$

This method is particularly effective in addressing multicollinearity in linear regression models. However, it does not necessarily set any coefficients to zero. Hence, ridge regression does not produce a sparse solution of estimated coefficients. On the other hand, LASSO regression

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda ||\boldsymbol{b}||_1, \tag{6}$$

incorporates an L^1 regularization penalty, which encourages sparsity in the solution by setting some coefficients exactly to zero (Tibshirani, 1996).

Other penalty functions are introduced to provide a balance between inducing sparsity and reducing estimation bias, aiming to solve the optimization problem as

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + P_\lambda(\boldsymbol{b}). \tag{7}$$

For example, the SCAD penalty function was introduced by Fan & Li (2001) as an improvement over Lasso regularization, particularly for bias reduction. The SCAD penalty function is defined as

$$P_\lambda^{SCAD}(\boldsymbol{b}) = \begin{cases} \lambda|\boldsymbol{b}| & \text{if } |\boldsymbol{b}| \le \lambda, \\ \frac{-|\boldsymbol{b}|^2 + 2a\lambda|\boldsymbol{b}| - \lambda^2}{2(a-1)} & \text{if } \lambda < |\boldsymbol{b}| \le a\lambda, \\ \frac{(a+1)\lambda^2}{2} & \text{if } |\boldsymbol{b}| > a\lambda, \end{cases} \tag{8}$$

where $\lambda > 0$ and $a > 0$ are unknown parameters. Fan & Li (2001) suggested that $a = 3.7$ is a good choice for various problems, and λ needs to be tuned.

The MCP is another type of penalty function introduced by Zhang (2010). The MCP penalty function is defined as

$$P_\lambda^{MCP}(\boldsymbol{b}) = \begin{cases} \lambda|\boldsymbol{b}| - \frac{\boldsymbol{b}^2}{2a} & \text{if } |\boldsymbol{b}| \le \lambda a, \\ \frac{a\lambda^2}{2} & \text{if } |\boldsymbol{b}| > a\lambda. \end{cases} \tag{9}$$

According to the estimation theorems of Zhang (2010), $a = 3$ is a good choice for MCP, and λ still needs to be tuned. MCP was developed to address the estimation bias of the LASSO and is generally easier to optimize computationally compared to SCAD.

Both SCAD and MCP aim to eliminate unimportant variables while preserving important ones, achieving the 'oracle property' as the sample size grows ($n \to \infty$). They both asymptotically select the correct model and produce normal, accurate coefficient estimates. MCP is effective with many sparse predictor groups but struggles with tightly clustered non-zero coefficients while SCAD has weaker grouping behavior compared to MCP (Ogutu & Piepho, 2014). We maintain the use of $a = 3.7$ for SCAD and $a = 3$ for MCP throughout the paper.

Another example is the elastic net regression which combines both L^1 and L^2 regularization penalties, providing a balanced approach to prediction accuracy on future data and model interpretation in linear regression models. It is formulated as

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda_1 ||\boldsymbol{b}||_1 + \lambda_2 ||\boldsymbol{b}||_2^2, \tag{10}$$

which has two regularization parameters λ_1 and λ_2 to tune (Zou & Hastie, 2005). The LAVA regression model is based on the splitting of the regression component into one sparse and one dense part $\boldsymbol{b} = \boldsymbol{c} + \boldsymbol{d}$ and thereby obtaining the following optimization problem

$$\hat{\boldsymbol{c}}, \hat{\boldsymbol{d}} = \underset{\boldsymbol{c},\boldsymbol{d}}{\operatorname{argmin}} \, \|\boldsymbol{y} - \boldsymbol{X}(\boldsymbol{c} + \boldsymbol{d})\|_2^2 + \lambda_1 \|\boldsymbol{c}\|_1 + \lambda_2 \|\boldsymbol{d}\|_2^2, \tag{11}$$

where the resulting estimator $\hat{\boldsymbol{b}} = \hat{\boldsymbol{c}} + \hat{\boldsymbol{d}}$ (Chernozhukov et al., 2017).

Waldmann (2021) developed a proximal operator algorithm based on the recent LAVA regularization method that jointly performs L^1- and L^2-norm regularization. Our paper develops a novel method for combined L^1- and L^0-norm regularization.

3 Methodological Framework

We study the performance of the proposed method in the ADMM, PRSM and SCPRSM settings using the augmented Lagrangian method that combines the original objective function with the constraints of the optimization problem into a single function. Here, the augmented Lagrangian's advantage lies in enabling the study of convergence for the proposed methods without requiring assumptions like strict convexity (Boyd et al., 2011). First, for clarity, we rewrite (4) as

$$\begin{aligned} \hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \, \{f(\boldsymbol{b}) + g(\boldsymbol{b})\} \quad \Longleftrightarrow \quad \hat{\boldsymbol{b}} = & \underset{\boldsymbol{b}}{\operatorname{argmin}} \, \{f(\boldsymbol{b}) + g(\boldsymbol{u})\} \\ & \text{subject to} \quad \boldsymbol{b} = \boldsymbol{u}, \end{aligned} \tag{12}$$

where $f(\boldsymbol{b})$ is a loss function and $g(\boldsymbol{b})$ is a penalty function. We now write the augmented Lagrangian function corresponding to (12) as

$$L_\gamma(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{z}) = f(\boldsymbol{b}) + g(\boldsymbol{u}) + \boldsymbol{z}^T(\boldsymbol{b} - \boldsymbol{u}) + \frac{\gamma}{2} ||\boldsymbol{b} - \boldsymbol{u}||_2^2, \tag{13}$$

where $\boldsymbol{z}$ is a dual variable or Lagrange multiplier and $\gamma > 0$ is a learning rate. Here, $\boldsymbol{b}$ and $\boldsymbol{u}$ are called primal variables.

3.1 Method of Multipliers and ADMM Framework

The method of multipliers jointly minimizes the two primal variables whereas the ADMM efficiently solves optimization problems by alternately updating primal and dual variables, effectively decomposing complex problems into manageable subproblems (Boyd et al., 2011). A convenient way to implement ADMM, PRSM and SCPRSM using proximal operator is obtained by completing the square with the dual variable $\boldsymbol{z}$ and

residual $\boldsymbol{b}-\boldsymbol{u}$ in the augmented Lagrangian function (13) (Parikh & Boyd, 2013). Now, we introduce the scaled dual variable $\boldsymbol{m}=\frac{1}{\gamma}\boldsymbol{z}$. Then, $\boldsymbol{z}^T(\boldsymbol{b}-\boldsymbol{u})+\frac{\gamma}{2}||\boldsymbol{b}-\boldsymbol{u}||_2^2=\frac{\gamma}{2}||\boldsymbol{b}-\boldsymbol{u}+\boldsymbol{m}||_2^2-\frac{\gamma}{2}||\boldsymbol{m}||_2^2$. Consequently, we write the scaled form of (13) as

$$L_\gamma(\boldsymbol{b},\boldsymbol{u},\boldsymbol{m})=f(\boldsymbol{b})+g(\boldsymbol{u})+\frac{\gamma}{2}||\boldsymbol{b}-\boldsymbol{u}+\boldsymbol{m}||_2^2-\frac{\gamma}{2}||\boldsymbol{m}||_2^2. \tag{14}$$

The method of multipliers for (14) can be written as

$$(\boldsymbol{b}^{(k+1)},\boldsymbol{u}^{(k+1)}):=\underset{\boldsymbol{b},\boldsymbol{u}}{\operatorname{argmin}}\, L_\gamma(\boldsymbol{b}^{(k)},\boldsymbol{u}^{(k)},\boldsymbol{m}^{(k)}), \tag{15}$$

$$\boldsymbol{m}^{(k+1)}:=\boldsymbol{m}^{(k)}+\boldsymbol{b}^{(k+1)}-\boldsymbol{u}^{(k+1)}. \tag{16}$$

The method of multipliers is generally not an implementable method since the primal update step (15), can be as hard to solve as the original problem (Beck, 2017; Boyd et al., 2011). To overcome this challenge, ADMM employs an iterative approach in the primal update step. In this approach, $\boldsymbol{b}$ and $\boldsymbol{u}$ are updated sequentially in an alternating fashion, which is why the method is called the alternating direction method of multipliers.

An iterative scheme for the scaled ADMM associated with (14) is presented as

$$\boldsymbol{b}^{(k+1)}:=\underset{\boldsymbol{b}}{\operatorname{argmin}}\, L_\gamma(\boldsymbol{b}^{(k)},\boldsymbol{u}^{(k)},\boldsymbol{m}^{(k)}), \tag{17a}$$

$$\boldsymbol{u}^{(k+1)}:=\underset{\boldsymbol{u}}{\operatorname{argmin}}\, L_\gamma(\boldsymbol{b}^{(k+1)},\boldsymbol{u}^{(k)},\boldsymbol{m}^{(k)}), \tag{17b}$$

$$\boldsymbol{m}^{(k+1)}:=\boldsymbol{m}^{(k)}+\boldsymbol{b}^{(k+1)}-\boldsymbol{u}^{(k+1)}. \tag{17c}$$

3.2 PRSM Framework

Here, unlike ADMM, the PRSM updates the dual variable $\boldsymbol{m}$ twice: once after each primal minimization of the augmented Lagrangian function (He et al., 2014; Li & Yuan, 2015; Peaceman & Rachford, 1955). We write the iterative scheme of the generalized PRSM for the augmented Lagrangian function (14) as

$$\boldsymbol{b}^{(k+1)}:=\underset{\boldsymbol{b}}{\operatorname{argmin}}\, L_\gamma(\boldsymbol{b}^{(k)},\boldsymbol{u}^{(k)},\boldsymbol{m}^{(k)}), \tag{18a}$$

$$\boldsymbol{m}^{(k+\frac{1}{2})}:=\boldsymbol{m}^{(k)}+r(\boldsymbol{b}^{(k+1)}-\boldsymbol{u}^{(k)}), \tag{18b}$$

$$\boldsymbol{u}^{(k+1)}:=\underset{\boldsymbol{u}}{\operatorname{argmin}}\, L_\gamma(\boldsymbol{b}^{(k+1)},\boldsymbol{u}^{(k)},\boldsymbol{m}^{(k+\frac{1}{2})}), \tag{18c}$$

$$\boldsymbol{m}^{(k+1)}:=\boldsymbol{m}^{(k+\frac{1}{2})}+r(\boldsymbol{b}^{(k+1)}-\boldsymbol{u}^{(k+1)}), \tag{18d}$$

where the parameter $r\in(0,1]$ is a relaxation factor. If $r=1$, the algorithm is referred to as PRSM whereas $r\in(0,1)$ enforces the strict contractiveness of the iterative sequence and is usually denoted SCPRSM (He et al., 2014).

3.3 Convergence Analysis

Now, we delve into the convergence properties of ADMM, PRSM and SCPRSM when applied to our developed nonconvex and nonsmooth problem framework. This analysis will elucidate the conditions under which these methods converge and the nature of the solutions they yield.

The derivatives of the Lagrangian (14) with respect to $(\boldsymbol{b},\boldsymbol{u},\boldsymbol{m})$ is given as

$$\partial_{\boldsymbol{b}}L_\gamma(\boldsymbol{b},\boldsymbol{u},\boldsymbol{m})=\nabla f(\boldsymbol{b})+\gamma(\boldsymbol{b}-\boldsymbol{u}+\boldsymbol{m}), \tag{19a}$$

$$\partial_{\boldsymbol{u}}L_\gamma(\boldsymbol{b},\boldsymbol{u},\boldsymbol{m})=\partial g(\boldsymbol{u})-\gamma(\boldsymbol{b}-\boldsymbol{u}+\boldsymbol{m}), \tag{19b}$$

$$\partial_{\boldsymbol{m}}L_\gamma(\boldsymbol{b},\boldsymbol{u},\boldsymbol{m})=\gamma(\boldsymbol{b}-\boldsymbol{u}). \tag{19c}$$

Let $(\boldsymbol{b}^*, \boldsymbol{u}^*, \boldsymbol{m}^*)$ be the equilibrium points (also called critical points) of (19a - 19c). Then, they must satisfy the following condition

$$\begin{aligned} -\gamma \boldsymbol{m}^* &= \nabla f(\boldsymbol{b}^*), \\ \gamma \boldsymbol{m}^* &\in \partial g(\boldsymbol{u}^*), \\ \boldsymbol{b}^* - \boldsymbol{u}^* &= 0. \end{aligned} \tag{20}$$

The convergence analysis of our proposed method is based on the following theorems.

Theorem 1 *Let the sequences $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}_{k=0}^{\infty}$ be generated by the ADMM scheme (17a - 17c) and its Lagrangian is given by (14). Then the following four conditions hold:*

(a) ***Sufficient decrease condition**: For each iteration step k, $\exists \delta_1 > 0$ such that*

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \leq -\delta_1 ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2.$$

(b) ***Boundness condition**: The sequences $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}_{k=0}^{\infty}$ are bounded and its Lagrangian $L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)})$ is lower bounded.*

(c) ***Sub-gradient boundness condition**: There exists $\boldsymbol{\varphi}_1^{(k+1)} \in \partial L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)})$, and $\delta_2 > 0$ such that*

$$||\boldsymbol{\varphi}_1^{(k+1)}||_2^2 \leq \delta_2 ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||^2.$$

(d) ***Global convergence**: The Lagrangian in (14) is a Kurdyka-Łojasiewicz (KŁ) function (see Appendix A.3), hence the proposed method has global convergence.*

Note that the function $f(\boldsymbol{b}) = ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2$ is a continuously differentiable function with respect to $\boldsymbol{b}$. Its gradient is computed as $\nabla f(\boldsymbol{b}) = -2\boldsymbol{X}^T\boldsymbol{y} + 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{b}$. Then the Lipschitz constant for the gradient of the function $f(\boldsymbol{b})$ can be computed as

$$\begin{aligned} ||\nabla f(\boldsymbol{b}^{(k)}) - \nabla f(\boldsymbol{b}^{(k+1)})|| &\leq 2||\boldsymbol{X}^T\boldsymbol{X}||||\boldsymbol{b}^{(k)} - \boldsymbol{b}^{(k+1)}|| \\ &\leq 2\lambda_{\max}(\boldsymbol{X}^T\boldsymbol{X})||\boldsymbol{b}^{(k)} - \boldsymbol{b}^{(k+1)}|| \\ &= l_f||\boldsymbol{b}^{(k)} - \boldsymbol{b}^{(k+1)}||, \end{aligned} \tag{21}$$

where $||\boldsymbol{X}^T\boldsymbol{X}||$ is the largest eigenvalue of $\boldsymbol{X}^T\boldsymbol{X}$ computed as $\lambda_{\max}(\boldsymbol{X}^T\boldsymbol{X})$. We denote the Lipschitz gradient constant as $l_f = 2\lambda_{\max}(\boldsymbol{X}^T\boldsymbol{X})$. From Equation (14), we obtain that $\dfrac{\partial^2 L_\gamma(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{m})}{\partial \boldsymbol{b}^2} = 2\boldsymbol{X}^T\boldsymbol{X} + \gamma\boldsymbol{I}$ which is positive definite. We will frequently use the properties of the Lipschitz gradient constant of $f(\boldsymbol{b})$ and the positive definite properties of $L_\gamma(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{m})$ with respect to $\boldsymbol{b}$ in the proof (see Appendix B).

Theorem 2 *Let the sequences $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}_{k=0}^{\infty}$ be generated by the scheme (18a - 18d) and its Lagrangian is given by (14). Then conditions (a)-(d) in Theorem 1 hold.*

The proof of Theorem 2 is found in Appendix C.

In conclusion, the sufficient decreasing and boundedness conditions are satisfied when the learning rate $\gamma > \max\{\dfrac{2l_f^2}{\rho}, l_f\}$ in both Theorem 1 and Theorem 2. In practice, computing the eigenvalues becomes laborious as the size of $\boldsymbol{X}$ increases. Hence, the learning rate γ can generally be determined using backtracking line search (see Section 3.5). The sufficient decreasing condition can be verified for the mean squared error (MSE) loss as the update step inherently minimizes the loss, ensuring it decreases as the number of iterations increase.

3.4 Implementation

Here, we simplify (17a - 17c) as

$$\begin{aligned}\boldsymbol{b}^{(k+1)} &:= \underset{\boldsymbol{b}}{\operatorname{argmin}}\,\{f(\boldsymbol{b}^{(k)}) + \frac{\gamma}{2}||\boldsymbol{b}^{(k)} - (\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)})||_2^2\},\\ \boldsymbol{u}^{(k+1)} &:= \underset{\boldsymbol{u}}{\operatorname{argmin}}\,\{g(\boldsymbol{u}^{(k)}) + \frac{\gamma}{2}||\boldsymbol{u}^{(k)} - (\boldsymbol{b}^{(k+1)} + \boldsymbol{m}^{(k)})||_2^2\},\\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k)} + \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}.\end{aligned} \tag{22}$$

With proximal operators, we can now easily write the ADMM scheme for (22) as

$$\begin{aligned}\boldsymbol{b}^{(k+1)} &:= \operatorname{prox}_{f\gamma}(\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}),\\ \boldsymbol{u}^{(k+1)} &:= \operatorname{prox}_{g\gamma}(\boldsymbol{b}^{(k+1)} + \boldsymbol{m}^{(k)}),\\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k)} + \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}.\end{aligned} \tag{23}$$

Similarly, the proximal version of equation (18a - 18d) can be written as

$$\begin{aligned}\boldsymbol{b}^{(k+1)} &:= \operatorname{prox}_{f\gamma}(\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}),\\ \boldsymbol{m}^{(k+\frac{1}{2})} &:= \boldsymbol{m}^{(k)} + r(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k)}),\\ \boldsymbol{u}^{(k+1)} &:= \operatorname{prox}_{g\gamma}(\boldsymbol{b}^{(k+1)} + \boldsymbol{m}^{(k+\frac{1}{2})}),\\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k+\frac{1}{2})} + r(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}).\end{aligned} \tag{24}$$

We are interested in splitting $\boldsymbol{b}$ into $\boldsymbol{c}$ and $\boldsymbol{d}$ with respect to L^1 and L^0 regularization, respectively. Hence, we implement the WL1L0-ADMM, WL1L0-PRSM, and WL1L0-SCPRSM schemes by defining two translation functions: $\mathcal{T}(\boldsymbol{u}) = f(\boldsymbol{u}+\boldsymbol{v}) - \boldsymbol{v}$ and $\mathcal{T}(\boldsymbol{v}) = f(\boldsymbol{v}+\boldsymbol{u}) - \boldsymbol{u}$. Then, for WL1L0-ADMM we obtain

$$\begin{aligned}\boldsymbol{c}^{(k+1)} &:= \operatorname{prox}_{\mathcal{T}(\boldsymbol{u})\gamma}(\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}),\\ \boldsymbol{u}^{(k+1)} &:= \operatorname{prox}_{g\gamma}(\boldsymbol{c}^{(k+1)} + \boldsymbol{m}^{(k)}),\\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k)} + \boldsymbol{c}^{(k+1)} - \boldsymbol{u}^{(k+1)},\\ \boldsymbol{d}^{(k+1)} &:= \operatorname{prox}_{\mathcal{T}(\boldsymbol{v})\delta}(\boldsymbol{v}^{(k)} - \boldsymbol{w}^{(k)}),\\ \boldsymbol{v}^{(k+1)} &:= \operatorname{prox}_{h\delta}(\boldsymbol{d}^{(k+1)} + \boldsymbol{w}^{(k)}),\\ \boldsymbol{w}^{(k+1)} &:= \boldsymbol{w}^{(k)} + \boldsymbol{d}^{(k+1)} - \boldsymbol{v}^{(k+1)}.\end{aligned} \tag{25}$$

For WL1L0-PRSM and WL1L0-SCPRSM we obtain

$$\begin{aligned}\boldsymbol{c}^{(k+1)} &:= \operatorname{prox}_{\mathcal{T}(\boldsymbol{u})\gamma}(\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}),\\ \boldsymbol{m}^{(k+\frac{1}{2})} &:= \boldsymbol{m}^{(k)} + r(\boldsymbol{c}^{(k+1)} - \boldsymbol{u}^{(k)}),\\ \boldsymbol{u}^{(k+1)} &:= \operatorname{prox}_{g\gamma}(\boldsymbol{c}^{(k+1)} + \boldsymbol{m}^{(k+\frac{1}{2})}),\\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k+\frac{1}{2})} + r(\boldsymbol{c}^{(k+1)} - \boldsymbol{u}^{(k+1)}),\\ \boldsymbol{d}^{(k+1)} &:= \operatorname{prox}_{\mathcal{T}(\boldsymbol{v})\delta}(\boldsymbol{v}^{(k)} - \boldsymbol{w}^{(k)}),\\ \boldsymbol{w}^{(k+\frac{1}{2})} &:= \boldsymbol{w}^{(k)} + r(\boldsymbol{d}^{(k+1)} - \boldsymbol{v}^{(k)}),\\ \boldsymbol{v}^{(k+1)} &:= \operatorname{prox}_{h\delta}(\boldsymbol{d}^{(k+1)} + \boldsymbol{w}^{(k+\frac{1}{2})}),\\ \boldsymbol{w}^{(k+1)} &:= \boldsymbol{w}^{(k+\frac{1}{2})} + r(\boldsymbol{d}^{(k+1)} - \boldsymbol{v}^{(k+1)})\end{aligned} \tag{26}$$

with $r = 1$ and $r \in (0, 1)$, respectively. Here, $\operatorname{prox}_{g\gamma}(\boldsymbol{c}+\boldsymbol{m})$ is the soft-thresholding function with learning rate γ, defined as

$$\operatorname{prox}_{g\gamma}(\boldsymbol{c}+\boldsymbol{m}) = \mathcal{S}_\gamma(\boldsymbol{c}+\boldsymbol{m}) = \max(0, |\boldsymbol{c}+\boldsymbol{m}| - \gamma)\operatorname{sgn}(\boldsymbol{c}+\boldsymbol{m}), \tag{27}$$

and $\text{prox}_{h\delta}(\boldsymbol{d}+\boldsymbol{w}) = \mathcal{H}_{\sqrt{2\delta}}(\boldsymbol{d}+\boldsymbol{w})$ is hard thresholding operator defined by

$$\mathcal{H}_{\sqrt{2\delta}}(\boldsymbol{d}+\boldsymbol{w}) = \begin{cases} 0, & \text{if } |\boldsymbol{d}+\boldsymbol{w}| < \sqrt{2\delta}, \\ \boldsymbol{d}+\boldsymbol{w}, & \text{if } |\boldsymbol{d}+\boldsymbol{w}| > \sqrt{2\delta}, \\ \{0, \boldsymbol{d}+\boldsymbol{w}\}, & \text{if } |\boldsymbol{d}+\boldsymbol{w}| = \sqrt{2\delta}. \end{cases} \tag{28}$$

The iterations are terminated when convergence is reached according to $\|(\boldsymbol{c}^{(k)} + \boldsymbol{d}^{(k)}) - (\boldsymbol{u}^{(k)} + \boldsymbol{v}^{(k)})\|_\infty \leq \beta(1 + \|\boldsymbol{m}^{(k)} + \boldsymbol{w}^{(k)})\|_\infty)$ for tolerance parameter β which was set to 10^{-5}.

For comparison purposes, we implement LASSO-ADMM (23). Here, we use

$$\text{prox}_{g\gamma}(\boldsymbol{b}+\boldsymbol{m}) = \max(0, |\boldsymbol{b}+\boldsymbol{m}| - \gamma)\text{sgn}(\boldsymbol{b}+\boldsymbol{m}). \tag{29}$$

We also implement adaptive versions of both SCAD and MCP using the ADMM algorithm. The iterative scheme for the SCAD can be formulated as

$$\begin{aligned} \boldsymbol{b}^{(k+1)} &:= \text{prox}_{f\gamma}(\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}), \\ \boldsymbol{u}^{(k+1)} &:= \text{prox}_{\text{scad}\gamma}(\boldsymbol{b}^{(k+1)} + \boldsymbol{m}^{(k)}), \\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k)} + \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}. \end{aligned} \tag{30}$$

Similarly, the iterative scheme for the MCP is given as

$$\begin{aligned} \boldsymbol{b}^{(k+1)} &:= \text{prox}_{f\gamma}(\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}), \\ \boldsymbol{u}^{(k+1)} &:= \text{prox}_{\text{mcp}\gamma}(\boldsymbol{b}^{(k+1)} + \boldsymbol{m}^{(k)}), \\ \boldsymbol{m}^{(k+1)} &:= \boldsymbol{m}^{(k)} + \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}. \end{aligned} \tag{31}$$

The closed-form proximal mappings of the SCAD (8) and MCP (9) penalty functions can be found in (Fan & Li, 2001; Liao et al., 2023; Wang & Liu, 2024; Yun et al., 2021). Here, we utilize the scaled versions

$$\text{prox}_{\text{scad}\gamma}(\boldsymbol{b}) = \begin{cases} \mathcal{S}_{\gamma\lambda}(\boldsymbol{b}+\boldsymbol{m}) & \text{if } |\boldsymbol{b}+\boldsymbol{m}| \leq (1+\gamma)\lambda, \\ \frac{(a-1)(\boldsymbol{b}+\boldsymbol{m}) - \text{sign}(\boldsymbol{b}+\boldsymbol{m})a\lambda\gamma}{a-1-\gamma} & \text{if } (1+\gamma)\lambda < |\boldsymbol{b}+\boldsymbol{m}| \leq a\lambda, \\ \boldsymbol{b}+\boldsymbol{m} & \text{if } |\boldsymbol{b}+\boldsymbol{m}| > a\lambda, \end{cases} \tag{32}$$

$$\text{prox}_{\text{mcp}\gamma}(\boldsymbol{b}+\boldsymbol{m}) = \begin{cases} \frac{a\gamma}{a\gamma-1}\mathcal{S}_{\gamma\lambda}(\boldsymbol{b}+\boldsymbol{m}) & \text{if } |\boldsymbol{b}+\boldsymbol{m}| \leq a\gamma\lambda, \\ \boldsymbol{b}+\boldsymbol{m} & \text{otherwise}, \end{cases} \tag{33}$$

with respect to SCAD and MCP, respectively. All iterations of LASSO-ADMM, SCAD-ADMM, and MCP-ADMM terminate upon achieving convergence, defined by the condition $\|\boldsymbol{b}^{(k)} - \boldsymbol{u}^{(k)}\|_\infty \leq \beta(1 + \|\boldsymbol{m}^{(k)}\|_\infty)$, where the tolerance parameter β is set to 10^{-5}.

3.5 Determining the Learning Rate

Choosing the learning rate (step size) is crucial for optimization algorithms' efficiency and convergence. There are two main methods for determining the learning rates γ and δ (Beck, 2017; Bertsekas, 2016; Boyd & Vandenberghe, 2004): 1. Constant learning rate uses a fixed learning rate throughout the process. If set correctly, it can lead to quick convergence, but it's sensitive to the chosen value. 2. Backtracking line search adjusts the learning rate based on certain criteria, making it more robust and often faster, though more

computationally expensive. We applied backtracking line search to determine γ following

$$\begin{aligned}
&\text{Initialize} \quad \tau = 0.5, \quad \gamma^{(k=2)} = 0.9 \\
&\text{For each iteration} \quad k \\
&\gamma^{(k)} = \gamma^{(k-1)} \\
&\text{while} \quad f(\boldsymbol{u}^{(k)}) > \{f(\boldsymbol{c}^{(k)}) + \\
&\qquad \nabla f(\boldsymbol{c}^{(k)})^{\intercal}(-\boldsymbol{c}^{(k)}) + \\
&\qquad (\frac{1}{2}\gamma^{(k)})||-(\boldsymbol{c}^{(k)})||_2^2||\} \\
&\qquad \text{repeat} \quad \gamma^{(k)} = \tau\gamma^{(k)} \\
&\text{end}
\end{aligned} \tag{34}$$

Here, $\nabla f(\boldsymbol{c}^{(k)}) = \boldsymbol{X}^T(\boldsymbol{X}(\boldsymbol{c}^{(k)}) - \boldsymbol{m})$ represents the gradient. Likewise, the procedure applies to δ by replacing $\boldsymbol{c}^{(k)}$ and $\boldsymbol{u}^{(k)}$ with $\boldsymbol{d}^{(k)}$ and $\boldsymbol{v}^{(k)}$, respectively.

3.6 Bayesian Optimization for Hyperparameter Tuning

Tuning the regularization parameter λ, the weight parameter α and the relazation factor r via cross-validation or grid search can be computationally expensive. Bayesian Optimization (BO) is a more advanced, data-driven approach which offers a probabilistic model-based method for hyperparameter tuning (Gao et al., 2021; Shahriari et al., 2015). For the latest advancements, see Wang et al. (2023) and Yang et al. (2024).

BO uses a surrogate model, often a Gaussian Processes (GP), to approximate the true objective function. Hyperparameters are collected in $\vartheta = [\alpha, \lambda, r]$ and the objective function $\iota[\vartheta]$ is modeled as $\iota[\vartheta] \sim \mathcal{GP}(m[\vartheta], k[\vartheta, \vartheta'])$, where $m[\vartheta]$ is its mean and $k[\vartheta, \vartheta']$ the kernel (variance) function. The objective function is evaluated at t sequential points $\text{MSE}^{(t)} = \iota(\vartheta^{(t)})$, with $\text{MSE}^{(t)} \sim N(\iota(\vartheta^{(t)}), \sigma^2)$. This process induces a posterior over the acquisition function, guiding the selection of the next hyperparameters. Common acquisition functions include probability of improvement (PI), expected improvement (EI), upper confidence bound (UCB), and mutual information (MI) (Snoek et al., 2012). BO starts with an initial set of hyperparameters and objective function values to train the surrogate model. The acquisition function balances the posterior mean ($\varpi(\vartheta)$) for exploitation and variance ($\upsilon(\vartheta)$) for exploration. The GP-UCB is given by

$$\vartheta^{(t+1)} = \underset{\vartheta}{\text{argmax}}\{\varpi(\vartheta) + \varkappa\upsilon(\vartheta)\},$$

where $\varpi(\vartheta)$ is driven by the mean function $m(\vartheta)$, $\upsilon(\vartheta)$ by the variance function $k(\vartheta)$, and $\varkappa$ determines the trade-off between exploitation and exploration. Contal et al. (2014) improved GP-UCB with the Gaussian Process Mutual Information algorithm (GP-MI) $\vartheta^{(t+1)} = \underset{\vartheta}{\text{argmax}}\{\mu(\vartheta^{(t)}) + \sqrt{\log(2/\varrho)}(\sqrt{\Sigma(\vartheta^{(t)}) + \varsigma^{(t-1)}} - \sqrt{\varsigma^{(t-1)}})\}$, where ς controls exploration, $0 < \varrho < 1$, and $\Sigma(\vartheta^{(t)})$ is the variance function at $\vartheta^{(t)}$. Hence, BO algorithm iteratively tunes hyperparameters by optimizing the acquisition function to find the next point $\vartheta^{(t)}$, sampling the objective function at $\vartheta^{(t)}$ to obtain $\text{MSE}^{(t)}$, augmenting the dataset with this observation, and updating the GP accordingly.

3.7 Materials

We evaluate our proposed method using one simulated genomic dataset as well as two real-world genomic datasets.

3.7.1 Simulated QTLMAS 2010 Dataset

The dataset comprises 3226 individuals across 5 generations, with 20 founders (5 males and 15 females) (Szydlowski & Paczyńska, 2011). Each female mates once, producing approximately 30 progeny per birth. SNP data were simulated using a coalescent model on five autosomal chromosomes, each 100 Mbp long. A total of

10,031 markers were generated, including 263 monomorphic SNPs and 9768 biallelic SNPs. The continuous quantitative trait is controlled by 9 major QTLs at fixed positions, including two pairs of epistatic genes, 3 maternally imprinted genes, and two additive major genes with phenotypic effects of -3 and 3. The additive genes are positioned at SNP indices 4354 and 5327, whereas the major epistatic locus is at SNP 931. Additionally, a dominance locus was positioned at SNP 9212, with an effect of 5.00 assigned to the heterozygote and 5.01 to the upper homozygote. Moreover, an over-dominance locus was placed at SNP 9404, with an effect of 5.00 assigned to the heterozygote, -0.01 to the lower homozygote, and 0.01 to the upper homozygote. After filtering SNPs with MAF < 0.01, 9723 markers were retained and transformed into one-hot encoding, resulting in 29169 genomic markers. Generations 1 to 4 (individuals 1 to 2326) were used for training, and generation 5 (individuals 2327 to 3226) served as test data.

3.7.2 Real Pig Dataset

The Pig dataset contains data from 3,534 individuals, with high-density genotypes and phenotypes for five traits (Cleveland et al., 2012). Using the PorcineSNP60 chip, 52,842 SNPs were assessed and filtered to 50,282 based on a minor allele frequency threshold of <0.01. The chosen trait had a heritability of 0.58. After adjusting phenotypic data and excluding individuals with missing data, the final dataset included 3,152 individuals.

3.7.3 Real Mice Dataset

This dataset comes from an experiment aimed at identifying and locating quantitative trait loci (QTLs) associated with various complex traits in a population of mice. The dataset contains 1814 individuals who were genotyped for 10,346 polymorphic markers and two traits: body length (BL) and body mass index (BMI). In this study, BL is used. The dataset is from the Wellcome Trust and is available in the R package BGLR (Pérez & de Los Campos, 2014).

4 Results

The WL1L0-ADMM, WL1L0-PRSM, WL1L0-SCPRSM, and LASSO-ADMM methods were implemented in Julia 1.10.1 (Bezanson et al., 2017) using the ProximalOperators package (Antonello et al., 2018). For SCAD-ADMM and MCP-ADMM, we wrote our own code manually in Julia. For all methods, the BO was performed with the BayesianOptimization package using an ElasticGPE model and the squared exponential automatic relevance determination (SEArd) kernel (Fairbrother et al., 2018). The initial values of $\hat{\boldsymbol{b}}$, $\hat{\boldsymbol{c}}$ and $\hat{\boldsymbol{d}}$ were set to the marginal covariances between $\boldsymbol{y}$ and $\boldsymbol{X}$, multiplied by 0.0001. By conducting preliminary runs for each set of hyperparameters using BO, we identified the optimal range of parameters. BO with the MI acquisition function was executed for 250 iterations across all methods, with 4 GP function evaluations per iteration. The test MSE was monitored during the optimization process to ensure convergence, indicated by no further decrease in MSE. All analyses were executed on a Linux computing platform equipped with an AMD EPYC 7302P 16-Core Processor and 32GB of system memory.

4.1 Simulated QTLMAS 2010 Dataset

For the BO, the lower and upper bounds for λ regularization were set to 0.01 and 500.0, 0.01 and 1000.0, and 0.1 and 800.0 with respect to LASSO-ADMM, SCAD-ADMM, and MCP-ADMM, respectively. For WL1L0-PRSM, the lower and upper bounds for α were set to 0.01 and 0.99, and for λ, they were set to 0.001 and 500.0, respectively. Similarly, for WL1L0-SCPRSM, the lower and upper bounds for α were set to 0.001 and 0.999, for r they were set to 0.001 and 1.0, and for λ they were set to 0.001 and 500.0, respectively. The best result, with a minimum test MSE of 64.67, was found with WL1L0-SCPRSM at $\lambda = 347.80$, $\alpha = 0.87$, and $r = 0.63$ (Table 1). The timing of the last evaluation with optimized parameters showed that MCP-ADMM was the fastest, taking only 10.44 seconds. It should be noted that those methods with one regularization parameter tend to be faster to train compared to other methods with two or three hyperparameters.

Figure 4 (see Appendix D) illustrates that the MSE of all six methods decreases as the number of BO iterations increases, demonstrating improved error minimization.

Method	min MSE	λ	α	r	Time[a]
LASSO-ADMM	67.27	269.29	-	-	17.47
SCAD-ADMM	66.88	328.74	-	-	11.53
MCP-ADMM	69.78	381.16	-	-	10.44
WL1L0-PRSM	64.92	398.66	0.77	1.00	99.73
WL1L0-SCPRSM	64.67	347.80	0.87	0.63	64.23
WL1L0-ADMM	64.73	445.50	0.77	-	82.84

[a]The time in seconds corresponds to the last evaluation with optimized parameters.

Table 1: Minimum test MSE and optimal parameters for LASSO-ADMM, SCAD-ADMM, MCP-ADMM, WL1L0-PRSM, WL1L0-SCPRSM and WL1L0-ADMM are evaluated on the simulated QTLMAS data.

4.1.1 Real Pig Dataset

For the Pig dataset, we employed 5-fold cross-validation with random allocations into training and test data to obtain the minimum test MSE, which was averaged over the folds. Here, for all methods, BO was executed for 100 iterations with 3 GP function evaluations per iteration due to the large dataset size. The lower and upper bounds for λ were set to 200.0 and 2000.0, 0.01 and 1500.0, and 0.01 and 400.0 with respect to LASSO-ADMM, SCAD-ADMM, and MCP-ADMM, respectively. For WL1L0-ADMM, the lower and upper bounds for α were set to 0.001 and 0.999, and for λ, they were set to 0.001 and 3000.0. Similarly, for WL1L0-PRSM and WL1L0-SCPRSM, the lower and upper bounds for α and λ were set identically to those for WL1L0-ADMM. Finally, for WL1L0-SCPRSM, the lower and upper bounds for r were set to 0.001 and 1.0, respectively. We observed little variability in the minimum test MSE across the CV-folds for all methods. Hence, we report the mean minimum test MSE using the average estimates of the respective parameters for all methods. The best result, with the mean minimum test MSE of 4.520, was found with WL1L0-SCPRSM at $(\lambda = 444.60, \alpha = 0.30, r = 0.62)$ (Table 2). The timing of the last evaluation with optimized parameters showed that MCP-ADMM was the fastest, taking only 24.15 seconds.

Method	min MSE	λ	α	r	Time[a]
LASSO-ADMM	4.94	248.91	-	-	26.78
SCAD-ADMM	4.72	135.88	-	-	24.19
MCP-ADMM	4.76	75.01	-	-	24.15
WL1L0-PRSM	4.523	1406.25	0.09	1.00	632.05
WL1L0-SCPRSM	4.520	444.60	0.30	0.62	655.90
WL1L0-ADMM	4.54	2631.36	0.04	-	490.83

[a]The time in seconds corresponds to the last evaluation with optimized parameters.

Table 2: Mean minimum test MSE and optimal parameters over 5 CV-folds for LASSO-ADMM, SCAD-ADMM, MCP-ADMM, WL1L0-PRSM, WL1L0-SCPRSM and WL1L0-ADMM were evaluated on the pig data.

4.1.2 Real Mice Dataset

Similar to the Pig dataset, we employed 5-fold cross-validation. The lower and upper bounds for λ were set to 0.01 and 2000.0, 0.1 and 3000.0, and 0.01 and 1800.0 with respect to LASSO-ADMM, SCAD-ADMM, and MCP-ADMM, respectively. For WL1L0-ADMM, the lower and upper bounds for α were set to 0.01 and 0.99, and for λ, they were set to 0.001 and 500.0. Similarly, for WL1L0-PRSM, the lower and upper bounds for α and λ were set identically to those for WL1L0-ADMM. Finally, for WL1L0-SCPRSM, the lower and upper bounds were set to 0.01 and 0.99, 0.001 and 500.0, and 0.001 and 1.0, with respect to α, λ, and r, respectively. The best result, with the mean minimum test MSE of 0.261, was found with WL1L0-SCPRSM at $(\lambda = 78.13, \alpha = 0.16, r = 0.53)$ (Table 3). The timing of the last evaluation with optimized parameters showed that MCP-ADMM was the fastest, taking only 3.16 seconds.

Method	min MSE	λ	α	r	Time[a]
LASSO-ADMM	0.273	750.01	-	-	5.48
SCAD-ADMM	0.273	1125.10	-	-	4.86
MCP-ADMM	0.273	675.01	-	-	3.16
WL1L0-PRSM	0.267	234.38	0.10	1.00	67.13
WL1L0-SCPRSM	0.261	78.13	0.16	0.53	46.54
WL1L0-ADMM	0.265	234.38	0.10	-	66.58

[a]The time in seconds corresponds to the last evaluation with optimized regularization parameters.

Table 3: Mean minimum test MSE and optimal parameters over 5 CV-folds for LASSO-ADMM, SCAD-ADMM, MCP-ADMM, WL1L0-PRSM, WL1L0-SCPRSM and WL1L0-ADMM were evaluated on the mice data.

5 Discussion

WL1L0-SCPRSM consistently achieves the lowest MSE across all datasets, indicating its superior ability to minimize prediction errors. This makes it highly effective in terms of accuracy and reliability for various types of data. The WL1L0 method is specifically designed to address the bias introduced by regularization methods like LASSO, SCAD, and MCP. By combining the weighted L^1 and L^0-norms, WL1L0-SCPRSM effectively reduces bias while maintaining model sparsity and interpretability. The weighting parameter α provides flexibility in tuning the regularization effect, making the method adaptable to different datasets and problem settings. This adaptability enhances its robustness and applicability across diverse scenarios.

The use of SCPRSM introduces an additional parameter r, allowing for finer control over the optimization process that potentially leads to better convergence properties and more precise model fitting. It is well known that L^0 regularization is computationally infeasible, as it is an NP-hard problem. Despite its higher computational time compared to some other methods, WL1L0-SCPRSM remains scalable for larger datasets due to its structured approach to optimization. The sparsity induced by the L^1 component and the precise variable selection by the L^0 component make the resulting model more interpretable. This is crucial in many scientific and industrial applications where understanding the model is as important as its predictive power.

Furthermore, WL1L0-PRSM and WL1L0-ADMM consistently achieve lower minimum MSE than LASSO-ADMM, SCAD-ADMM, and MCP-ADMM across all datasets, although both are outperformed by WL1L0-SCPRSM. Recent studies have shown that SCPRSM outperforms ADMM (Li & Yuan, 2015; Li et al., 2021), and SCAD and MCP often outperform the LASSO (Fan et al., 2014a; Fan & Li, 2001; Zhang, 2010).

6 Conclusion

This paper introduces a novel joint weighted L^1- and L^0-norm method based on proximal mapping and translations, aiming to debias the bias introduced by regularization methods for handling high-dimensional data. Our model introduces a weighting parameter, α, allowing for the adjustment of the influence of both regularizers. All parameters are optimized using Bayesian optimization, a data-driven method. The WL1L0 model outperforms all known regularization methods (LASSO, SCAD, and MCP). The global convergence of WL1L0-ADMM, WL1L0-PRSM, and WL1L0-SCPRSM is proved under reasonable assumptions. Furthermore, WL1L0-SCPRSM consistently achieves the lowest MSE across all datasets, indicating its superior ability to minimize prediction errors. The WL1L0-SCPRSM's superior performance across different datasets demonstrates its versatility. Our current paper focuses on prediction rather than variable selection (estimation). This leaves room for future work to address variable selection properties specifically.

References

Niccolò Antonello, Lorenzo Stella, Panagiotis Patrinos, and Toon Van Waterschoot. Proximal gradient algorithms: Applications in signal processing. *arXiv preprint arXiv:1803.01621*, 2018.

Hédy Attouch, Jérôme Bolte, Patrick Redont, and Antoine Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on theKurdyka-łojasiewicz inequality. *Mathematics of Operations Research*, 35(2):438–457, 2010.

Hedy Attouch, Jérôme Bolte, and Benar Fux Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods. *Mathematical Programming, Series A*, 137(1):91–129, 2013.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Dimitri P. Bertsekas. *Nonlinear programming, 3rd ed*. Athena Scientific, Nashua, NH, 2016.

Dimitris Bertsimas, Jean Pauphilet, and Bart Van Parys. Sparse regression: Scalable algorithms and empirical performance. *Statistical Science*, 35(4):555–578, 2020.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1):459–494, 2014.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer-Verlag Berlin Heidelberg, Berlin, Germany, 2011.

Victor Chernozhukov, Christian Hansen, and Yuan Liao. A lava attack on the recovery of sums of dense and sparse signals. *The Annals of Statistics*, 45(1):39–76, 2017.

Francis H Clarke, Yuri S Ledyaev, Ronald J Stern, and Peter R Wolenski. *Nonsmooth analysis and control theory*, volume 178. Springer Science & Business Media, 2008.

Matthew A Cleveland, John M Hickey, and Selma Forni. A common dataset for genomic analysis of livestock populations. *G3: Genes| Genomes| Genetics*, 2(4):429–435, 2012.

Emile Contal, Vianney Perchet, and Nicolas Vayatis. Gaussian process optimization with mutual information. In *International Conference on Machine Learning*, volume 32, pp. 253–261. PMLR, 2014.

Jamie Fairbrother, Christopher Nemeth, Maxime Rischard, Johanni Brea, and Thomas Pinder. Gaussianprocesses. jl: A nonparametric bayes package for the julia language. *arXiv preprint arXiv:1812.09064*, 2018.

Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

Jianqing Fan and Jinchi Lv. A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20(1):101–148, 2010.

Jianqing Fan, Jinchi Lv, and Lei Qi. Sparse high dimensional models in economics. *Annual Review of Economics*, 3(1):291–317, 2011.

Jianqing Fan, Yingying Fan, and Emre Barut. Adaptive robust variable selection. *Annals of Statistics*, 42 (1):324–351, 2014a.

Jianqing Fan, Fang Han, and Han Liu. Challenges of big data analysis. *National Science Review*, 1(2): 293–314, 2014b.

Masao Fukushima and Hisashi Mine. A generalized proximal point algorithm for certain non-convex minimization problems. *International Journal of Systems Science*, 12(8):989–1000, 1981.

Haiping Gao, Shifa Zhong, Wenlong Zhang, Thomas Igou, Eli Berger, Elliot Reid, Yangying Zhao, Dylan Lambeth, Lan Gan, Moyosore A. Afolabi, et al. Revolutionizing membrane design using machine learning-Bayesian optimization. *Environmental Science & Technology*, 56(4):2572–2581, 2021.

Christophe Giraud. *Introduction to High-Dimensional Statistics*. CRC Press, Boca Raton, FL, 2015.

Trevor Hastie, Robert Tibshirani, and Ryan Tibshirani. Best subset, forward stepwise or Lasso? Analysis and recommendations based on extensive comparisons. *Statistical Science*, 35(4):579–592, 2020.

Bingsheng He, Han Liu, Zhaoran Wang, and Xiaoming Yuan. A strictly contractive Peaceman–Rachford splitting method for convex programming. *SIAM Journal on Optimization*, 24(3):1011–1040, 2014.

Georg Heinze, Christine Wallisch, and Daniela Dunkler. Variable selection–a review and recommendations for the practicing statistician. *Biometrical Journal*, 60(3):431–449, 2018.

R R Hocking and R N Leslie. Selection of the best subset in regression analysis. *Technometrics*, 9:531–540, 1967.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Iain M Johnstone and D Michael Titterington. Statistical challenges of high-dimensional data, 2009.

Alexander Kaplan and Rainer Tichatschke. Proximal point methods and nonconvex optimization. *Journal of global Optimization*, 13:389–406, 1998.

Peixuan Li, Yuan Shen, Suhong Jiang, Zehua Liu, and Caihua Chen. Convergence study on strictly contractive peaceman–rachford splitting method for nonseparable convex minimization models with quadratic coupling terms. *Computational Optimization and Applications*, 78:87–124, 2021.

Xinxin Li and Xiaoming Yuan. A proximal strictly contractive Peaceman-Rachford splitting method for convex programming with applications to imaging. *SIAM Journal on Imaging Sciences*, 8(2):1332–1365, 2015.

Xingran Liao, Xuekai Wei, and Mingliang Zhou. Minimax concave penalty regression for superresolution image reconstruction. *IEEE Transactions on Consumer Electronics*, 70(1):2999–3007, 2023.

Boris S Mordukhovich. *Variational analysis and generalized differentiation II: Applications*, volume 331. Springer, 2006.

Balas Kausik Natarajan. Sparse approximate solutions to linear systems. *SIAM journal on computing*, 24 (2):227–234, 1995.

Joseph O Ogutu and Hans-Peter Piepho. Regularized group regression methods for genomic prediction: Bridge, mcp, scad, group bridge, group lasso, sparse group lasso, group mcp and group scad. In *BMC proceedings*, volume 8, pp. 1–9. Springer, 2014.

Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3): 123–231, 2013.

Donald W Peaceman and Henry H Rachford, Jr. The numerical solution of parabolic and elliptic differential equations. *Journal of the Society for Industrial and Applied Mathematics*, 3(1):28–41, 1955.

Paulino Pérez and Gustavo de Los Campos. Genome-wide regression and prediction with the bglr statistical package. *Genetics*, 198(2):483–495, 2014.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25, 2012.

Maciej Szydlowski and Paulina Paczyńska. Qtlmas 2010: simulated dataset. In *BMC proceedings*, volume 5, pp. 1–3. Springer, 2011.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Laura Toloşi and Thomas Lengauer. Classification with correlated features: unreliability of feature ranking and solutions. *Bioinformatics*, 27(14):1986–1994, 2011.

Martin J Wainwright. *High-dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press, Cambridge, United Kingdom, 2019.

Patrik Waldmann. A proximal lava method for genome-wide association and prediction of traits with mixed inheritance patterns. *BMC Bioinformatics*, 22(1):1–16, 2021.

Hao Wang, Zhanglei Shi, Chi-Sing Leung, and Hing Cheung So. Admm-mcp framework for sparse recovery with global convergence. *IEEE Transactions on Signal Processing*, 2018.

Ting Wang and Hongwei Liu. A class of modified accelerated proximal gradient methods for nonsmooth and nonconvex minimization problems. *Numerical Algorithms*, 95(1):207–241, 2024.

Xilu Wang, Yaochu Jin, Sebastian Schmitt, and Markus Olhofer. Recent advances in Bayesian optimization. *ACM Computing Surveys*, 55(13s):1–36, 2023.

Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78:29–63, 2019.

Kaixin Yang, Long Liu, and Yalu Wen. The impact of Bayesian optimization on feature selection. *Scientific Reports*, 14(1):3948, 2024.

Jihun Yun, Aurélie C Lozano, and Eunho Yang. Adaptive proximal gradient methods for structured neural networks. *Advances in Neural Information Processing Systems*, 34:24365–24378, 2021.

Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

Peng Zhao and Bin Yu. On model selection consistency of lasso. *The Journal of Machine Learning Research*, 7:2541–2563, 2006.

Tuo Zhao, Han Liu, and Tong Zhang. Pathwise coordinate optimization for sparse learning: Algorithm and theory. *The Annals of Statistics*, 46(1):180–218, 2018.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

A Appendix

A.1 Theoretical Background

We now provide selected theory that are most useful for solving (4). Our approach closely follows (Beck, 2017) and (Bertsekas, 2016). For an extended real-valued function $f : \mathbb{R}^p \to [-\infty, \infty]$, we define the following:

(a) The domain of f is the set

$$\text{dom}(f) = \{\boldsymbol{b} \in \mathbb{R}^p : f(\boldsymbol{b}) < \infty\}.$$

(b) f is proper if $\text{dom}(f) \neq \varnothing$ and f is never $-\infty$.

(c) The epigraph of f is defined by

$$\text{epi}(f) = \{(\boldsymbol{b}, a) \in \mathbb{R}^p \times \mathbb{R} : f(\boldsymbol{b}) \leq a\}.$$

(d) The function f is closed if its epigraph is closed.

(e) f is called lower semicontinuous at $\boldsymbol{b} \in \mathbb{R}^p$ if

$$f(\boldsymbol{b}) \leq \liminf_{n\to\infty} f(\boldsymbol{b}_n)$$

for any sequence $\{\boldsymbol{b}_n\}_{n\geq 1} \subseteq \mathbb{R}^p$ for which $\boldsymbol{b}_n \to \boldsymbol{b}$ as $n \to \infty$.

(f) For any $\eta \in \mathbb{R}$, the η-level set of a function f is the set

$$\text{Lev}(f, \eta) = \{\boldsymbol{b} \in \mathbb{R}^p : f(\boldsymbol{b}) \leq \eta\}.$$

(g) A proper function f is called coercive if

$$\lim_{||\boldsymbol{b}||\to\infty} f(\boldsymbol{b}) = \infty.$$

For any set $\mathbb{S} \subseteq \mathbb{R}^n$ and any point $\boldsymbol{b} \in \mathbb{R}^p$, the distance from $\boldsymbol{b}$ to $\mathbb{S}$ is defined as $D(\boldsymbol{b}, \mathbb{S}) := \inf\{||\boldsymbol{m}-\boldsymbol{b}||, \boldsymbol{m} \in \mathbb{S}\}$, and $D(\boldsymbol{b}, \mathbb{S}) = \infty$ for all $\boldsymbol{b}$ when $\mathbb{S} = \varnothing$.

A proper closed and coercive function f attains its minimal value over $\mathbb{S}$ for a nonempty closed set satisfying $\mathbb{S} \cap \text{dom}(f) = \emptyset$. Moreover, a closed coercive function possesses a minimizer on any closed set that has a nonempty intersection with the domain of the function (Beck, 2017). For an extended real-valued function $f : \mathbb{R}^p \to [-\infty, \infty]$, the following three claims are equivalent:

i f is lower semicontinuous.

ii f is closed.

iii For any ($\eta \in \mathbb{R}$), the level set

$$\text{Lev}(f, \eta) = \{\boldsymbol{b} \in \mathbb{R}^p : f(\boldsymbol{b}) \leq \eta\}$$

is closed.

The proof of these claims can be found in (Beck, 2017), see Theorem 2.6.

A.2 Subdifferentials of nonconvex and nonsmooth functions

Subdifferentials are important in analyzing complex functions, especially when dealing with nonsmooth and nonconvex functions. Following Clarke et al. (2008) and Mordukhovich (2006), we explore subdifferentiability.

Let $g : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper and lower semicontinuous function. Then

(i) For a given $\boldsymbol{b} \in \text{dom } g$, the Fréchet subdifferential of g at $\boldsymbol{b}$, denoted by $\partial g(\boldsymbol{b})$, is the set of all vectors $\boldsymbol{u} \in \mathbb{R}^p$ which satisfy

$$\liminf_{\boldsymbol{m}\to\boldsymbol{b}} \frac{g(\boldsymbol{m}) - g(\boldsymbol{b}) - \langle \boldsymbol{u}, \boldsymbol{m} - \boldsymbol{b}\rangle}{||\boldsymbol{m} - \boldsymbol{b}||} \geq 0,$$

and we set $\partial g = \varnothing$ when $\boldsymbol{b} \notin \text{dom } g$.

(ii) The limiting-subdifferential, or simply the subdifferential, of g at $\boldsymbol{b}$, written by $\partial g(\boldsymbol{b})$, is defined by

$$\partial g(\boldsymbol{b}) := \{\boldsymbol{u} \in \mathbb{R}^n : \exists \boldsymbol{b}_k \to \boldsymbol{b}, g(\boldsymbol{b}_k) \to g(\boldsymbol{b}) \text{ and } \partial g(\boldsymbol{b}_k) \overset{k\to\infty}{\longrightarrow} \boldsymbol{u}\}.$$

(iii) A point $\boldsymbol{b}^*$ is called critical point or stationary point of g if it satisfies $0 \in \partial g(\boldsymbol{b}^*)$.

A.3 The Kurdyka–Łojasiewicz Inequality and its Property

The Kurdyka-Łojasiewicz (KŁ) inequality deals with the behavior of certain functions near their critical points. It is an important tool for analyzing the global convergence of nonconvex nonsmooth optimization problems (Attouch et al., 2013; 2010; Bolte et al., 2014). We now revice the KŁ property.

Let $g : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper lower semicontinuous function. Then,

(a) The function $g : \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$ is said to have the KŁ property at $\boldsymbol{b}^* \in \operatorname{dom} \partial g$ if there exist $\eta \in (0, +\infty]$, a neighborhood U of $\boldsymbol{b}^*$, and a continuous concave function $\phi : [0, \eta) \to \mathbb{R}^+$ such that

(i) $\phi(0) = 0$,
(ii) ϕ is continuously differentiable on $(0, \eta)$,
(iii) $\forall a \in (0, +\infty]$, $\phi'(a) > 0$,
(IV) For all $\boldsymbol{b} \in U \cap \{g(\boldsymbol{b}^*) < g(\boldsymbol{b}) < g(\boldsymbol{b}^*) + \eta\}$, the KŁ property holds:

$$\phi'(g(\boldsymbol{b}) - g(\boldsymbol{b}^*)) d(0, \partial g(\boldsymbol{b})) \geq 1. \tag{8}$$

(b) Proper lower semicontinuous functions which satisfy the KŁ inequality at each point of $\operatorname{dom} \partial g$ are called KŁ functions.

A.4 Proximal Operators

Proximal operators are a fundamental concept in optimization, especially for problems involving non-smooth or non-convex functions, which are increasingly common in a wide range of real-world applications (Fukushima & Mine, 1981; Kaplan & Tichatschke, 1998; Parikh & Boyd, 2013). A proximal operator, denoted as $\operatorname{prox}_f(\boldsymbol{u})$, aims to find a point closer to $\boldsymbol{u}$ that also minimizes a specific objective function, $f(\boldsymbol{v})$ in a specific optimization subproblem. This subproblem is assumed to be more manageable to solve than the original problem. The proximal operator can be mathematically expressed as

$$\operatorname{prox}_f(\boldsymbol{u}) = \underset{\boldsymbol{v}}{\operatorname{argmin}} \{f(\boldsymbol{v}) + (1/2)\|\boldsymbol{v} - \boldsymbol{u}\|_2^2\}, \tag{35}$$

where $\boldsymbol{u}$ and $\boldsymbol{v}$ are vectors of length p. Here, $\operatorname{prox}_f(\boldsymbol{u})$ is a point that compromises between minimizing f and being close to $\boldsymbol{u}$. Note that the right-hand side of (35) is strongly convex, hence there is a unique minimizer for every $\boldsymbol{u} \in \mathbb{R}^p$. Introducing the parameter $\gamma > 0$ that represents a trade-off parameter between the two terms $\boldsymbol{v}$ and $\boldsymbol{u}$ yields a scaled version of (35), in which $\frac{1}{2}$ is replaced by $\frac{1}{2\gamma}$. The proximal operator has useful properties (Beck, 2017). For example, for an affine transformation $f(\boldsymbol{u}) = \langle \boldsymbol{m}, \boldsymbol{u} \rangle + a$, the proximal operator defined in (35) becomes $\operatorname{prox}_f(\boldsymbol{u}) = \boldsymbol{u} - \boldsymbol{m}$, which represents a translation mapping. Therefore, we can express a translation function as $\mathcal{T}(\boldsymbol{u}) = f(\boldsymbol{u} + \boldsymbol{m}) - \boldsymbol{m}$. Another important property arises in the context of separable sum functions $f(\boldsymbol{u}, \boldsymbol{m}) = g(\boldsymbol{u}) + h(\boldsymbol{m})$, where the proximity operator is written in $\operatorname{prox}_f(\boldsymbol{u}, \boldsymbol{m}) = \operatorname{prox}_g(\boldsymbol{u}) + \operatorname{prox}_h(\boldsymbol{m})$. In the following section, we discuss the problem (4) in detail.

A.5 Discussion of the Problem

A.5.1 Optimality Conditions

We write (4) as

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}}\ F(\boldsymbol{b}) := \Gamma(\boldsymbol{b}) + \lambda(1 - \alpha)||\boldsymbol{b}||_0, \tag{36}$$

where $\Gamma(\boldsymbol{b}) := ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2 + \lambda\alpha||\boldsymbol{b}||_1$. Since $\Gamma(\boldsymbol{b})$ is a convex function, it has a global minimum value. By the Weierstrass theorem, a continuous function over a nonempty compact set attains a minimum. The existence of an optimal solution is guaranteed if a function is continuous over a closed set and coercive over the set (Bertsekas, 2016). Beck (2017) demonstrates that the latter extends to closed functions, i.e. a closed and coercive function over a closed set attains an optimal solution.

For the case $||\boldsymbol{b}||_0 = \sum_{i=1}^{p} \mathbf{1}(b_i \neq 0)$ with $\lambda(1-\alpha) > 0$, we need to show it is a closed function. Let $g(\boldsymbol{b}) = \sum_{i=1}^{p} I(b_i)$ then for any $\boldsymbol{b} \in \mathbb{R}^p$ we have

$$I(t) = \begin{cases} \lambda(1-\alpha), & t \neq 0, \\ 0, & t = 0. \end{cases}$$

The function $I(.)$ is closed since its level sets, given by

$$\text{Lev}(I, \eta) = \begin{cases} \emptyset, & \eta < 0, \\ \{0\}, & \eta \in [0, 1), \\ \mathbb{R}, & \eta \geq 1, \end{cases} \tag{37}$$

are closed sets. Here, g is a sum of closed functions. Hence, g is closed. Furthermore, Using Theorem 2.6 in (Beck, 2017), the closedness of $\|\boldsymbol{b}\|_0$ implies its lower semi-continuity.

A vector $\boldsymbol{b}^*$ is a local minimum of the function F, if there exists $\varepsilon > 0$ such that $F(\boldsymbol{b}^*) \leq F(\boldsymbol{b})$ for all $\boldsymbol{b} \in \mathbb{R}^p$ with $||\boldsymbol{b} - \boldsymbol{b}^*|| < \varepsilon$. A vector $\boldsymbol{b}^*$ is a global minimum if $F(\boldsymbol{b}^*) \leq F(\boldsymbol{b})$ for all $\boldsymbol{b} \in \mathbb{R}^p$.

The function F is nonconvex because any point between the endpoints A and B, as indicated by the solid red line in Figure 1, lies outside the domain of F. In fact, the shape of the function F is similar to that of nonconvex regularization methods such as SCAD and MCP (Fan & Li, 2001; Zhang, 2010; Zhao et al., 2018). Various shapes of the function F for different values of α are depicted in Figure 3. Now, we define a

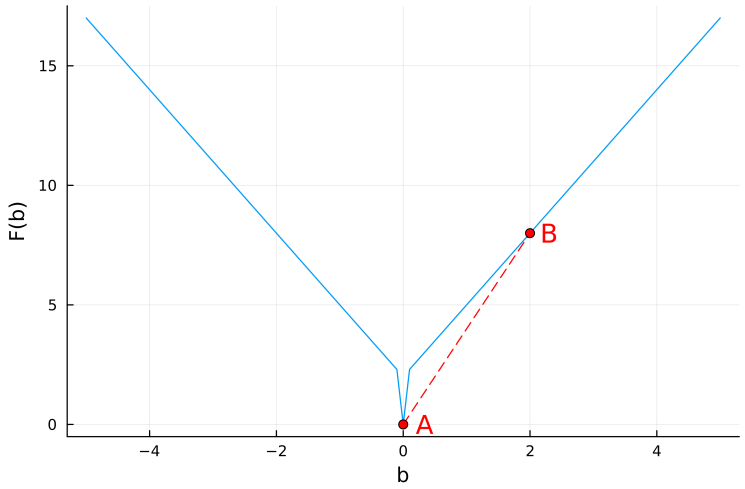

Figure 1: Illustration of the nonconvexity of the function F

set $\mathbb{S}_\eta = \{\boldsymbol{b} \in \mathbb{R}^p : F(\boldsymbol{b}) \leq \eta\}$, as shown in the Figure 2. This set represents the η-sublevel set of F, where $\eta \in \mathbb{R}$. Within this set, a global minimum $\boldsymbol{b}^*$ exists such that $F(\boldsymbol{b}^*) \leq \eta$. However, for any $\boldsymbol{b}$ not in $\mathbb{S}_\eta$, we have $F(\boldsymbol{b}) > \eta \geq F(\boldsymbol{b}^*)$.

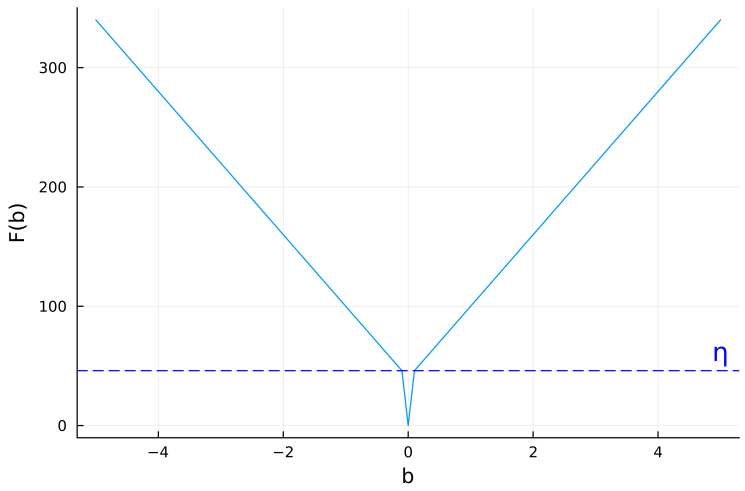

Figure 2: η-sublevel set of F

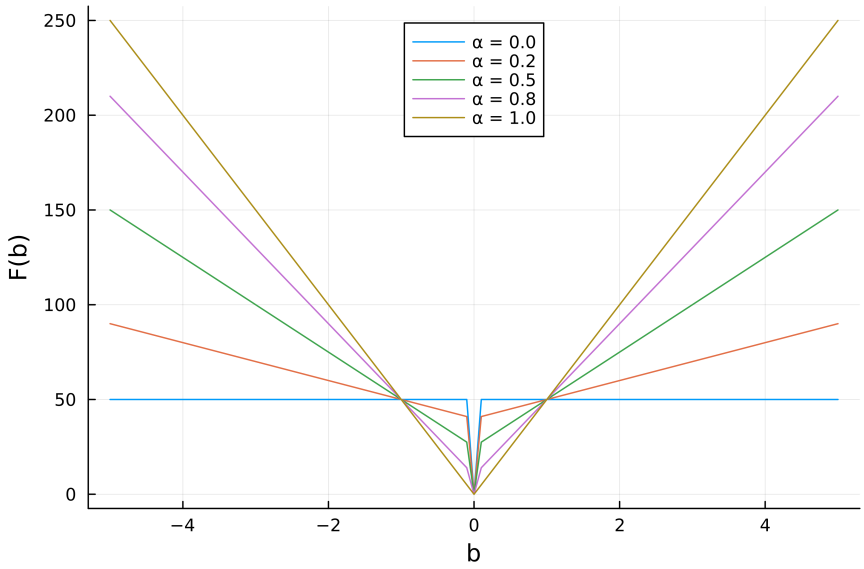

Figure 3: Different function values of F with regularizer $\lambda = 1$ and $\alpha = 0$ (L^0), 0.2, 0.5, 0.8, 1 (L^1).

B Appendix

Proof for Theorem 1: While our model differs from that of Wang et al. (2019; 2018), we adopt a similar proof framework.

Proof (a): From (17a), $\boldsymbol{b}^{(k+1)}$ minimizes $L_r(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)})$ and since $L_r(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{m})$ is strongly convex with respect to $\boldsymbol{b}$, the Lagrangian function satisfies the following inequality (Beck, 2017):

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) - L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \leq -\frac{\rho}{2}||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2, \tag{38}$$

where $L_\gamma(\cdot)$ is a ρ-strongly convex function ($\rho > 0$). From the augmented Lagrangian function in (14), we have

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k)}) = \gamma \left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}\right)^\intercal \left(\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}\right). \tag{39}$$

Now we rewrite (17c) as

$$\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} = \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}. \tag{40}$$

From (17b) and (19a), we get

$$\nabla f(\boldsymbol{b}^{(k+1)}) + \gamma(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k)}) = 0. \tag{41}$$

Substituting (17c) into (41), we obtain

$$\nabla f(\boldsymbol{b}^{(k+1)}) = -\gamma \boldsymbol{m}^{(k+1)}. \tag{42}$$

Using (40) and (42), (39) becomes

$$\gamma\left(||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}||^2\right) = \gamma\left(|| - \frac{1}{\gamma}\nabla f(\boldsymbol{b}^{(k+1)}) + \frac{1}{\gamma}\nabla f(\boldsymbol{b}^{(k)})||^2\right) \leq \frac{l_f^2}{\gamma}\left(||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||^2\right). \tag{43}$$

Hence,

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k)}) \leq \frac{l_f^2}{\gamma}\left(||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||^2\right). \tag{44}$$

Here, the term $l_f \geq 0$ denotes a Lipschitz gradient of the function $f(\boldsymbol{b})$. From (17b) we have that

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \leq 0. \tag{45}$$

Finally, combining (38), (44) and (45), we obtain the desired inequality as

$$\begin{aligned}
&L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \\
&= L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k)}) \\
&\quad + L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \\
&\quad + L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) - L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \\
&\leq \left(\frac{l_f^2}{\gamma} - \frac{\rho}{2}\right)||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2 \\
&= -\delta_1||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2,
\end{aligned} \tag{46}$$

where $\sigma_1 = \dfrac{\rho}{2} - \dfrac{l_f^2}{\gamma}$ and $\gamma > \dfrac{2l_f^2}{\rho}$. Hence, the sufficient decreasing condition is met.

Proof (b): We utilize the descent lemma to prove that $L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)})$ is lower bounded for any k.

Descent lemma: Let the function f belongs to the class of continuously differentiable functions with a constant l_f Lipschitz continuous gradients. Then for any two points $\boldsymbol{b}^{(k)}$ and $\boldsymbol{u}^{(k)}$,

$$f(\boldsymbol{u}^{(k)}) \leq f(\boldsymbol{b}^{(k)}) + \nabla f(\boldsymbol{b}^{(k)})^\top(\boldsymbol{u}^{(k)} - \boldsymbol{b}^{(k)}) + \frac{l_f}{2}||\boldsymbol{u}^{(k)} - \boldsymbol{b}^{(k)}||_2^2. \tag{47}$$

The proof of the descent lemma can be found in (Beck, 2017), see Lemma 5.7.

As a result of the lemma, the sequence is lower bounded as

$$\begin{aligned}
L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) =& f(\boldsymbol{b}) + g(\boldsymbol{u}) + \frac{\gamma}{2}||\boldsymbol{b} - \boldsymbol{u} + \frac{1}{\gamma}\boldsymbol{m}||_2^2 - \frac{\gamma}{2}||\frac{1}{\gamma}\boldsymbol{m}||_2^2 \\
=& f(\boldsymbol{b}^{(k)}) + g(\boldsymbol{u}^{(k)}) + \boldsymbol{m}^T(\boldsymbol{b}^{(k)} - \boldsymbol{u}^{(k)}) + (\gamma/2)\|\boldsymbol{b}^{(k)} - \boldsymbol{u}^{(k)}\|_2^2 \\
&\geq f(\boldsymbol{u}^{(k)}) + g(\boldsymbol{u}^{(k)}) + \left(\frac{\gamma}{2} - \frac{l_f}{2}\right)||\boldsymbol{u}^{(k)} - \boldsymbol{m}^{(k)}||_2^2 \\
&\geq -\infty \text{ for } \gamma \geq l_f.
\end{aligned} \tag{48}$$

Hence, from (48), $L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)})$ is lower bounded.

As established in the proof of (a), the sufficient descent property implies that $L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)})$ is upper-bounded by $L_\gamma(\boldsymbol{b}^0, \boldsymbol{u}^0, \boldsymbol{m}^0)$. To prove the sequence $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}$ is bounded, we start by rewriting (46) as

$$\begin{aligned} ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2 \leq & \frac{1}{\delta_1}(L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)})) \\ \sum_{k=0}^{l} ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2 \leq & \frac{1}{\delta_1}(L_\gamma(\boldsymbol{b}^0, \boldsymbol{u}^0, \boldsymbol{m}^0) - L_\gamma(\boldsymbol{b}^{l+1}, \boldsymbol{u}^{l+1}, \boldsymbol{m}^{l+1})) \\ & < \infty. \end{aligned} \tag{49}$$

Equation (49) also holds as $l \to \infty$. Hence, $\boldsymbol{b}^{(k)}$ is bounded.

From (43), we obtain

$$||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}||_2^2 \leq \frac{l_f^2}{\gamma^2}||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2.$$

$$\sum_{k=0}^{l} ||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}||_2^2 < \infty. \tag{50}$$

This implies that $\boldsymbol{m}^{(k)}$ is bounded.

Finally, from (40) we obtain $\boldsymbol{u}^{(k+1)} = \boldsymbol{b}^{(k+1)} - \boldsymbol{m}^{(k+1)} + \boldsymbol{m}^{(k)}$ and $\boldsymbol{u}^{(k)} = \boldsymbol{b}^{(k)} - \boldsymbol{m}^{(k)} + \boldsymbol{m}^{k-1}$. Then

$$\begin{aligned} ||\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}||_2^2 = & ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)} + \boldsymbol{m}^{(k)} - \boldsymbol{m}^{(k+1)} + \boldsymbol{m}^{k-1} - \boldsymbol{m}^{(k)}||_2^2 \\ & \leq ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2 + ||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}||_2^2 + ||\boldsymbol{m}^{(k)} - \boldsymbol{m}^{k-1}||_2^2. \end{aligned}$$

Consequently, we obtain

$$\sum_{k=1}^{\infty} ||\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}||_2^2 < \infty. \tag{51}$$

Hence, the sequence $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}$ is bounded.

Proof (c):

$$\frac{\partial L}{\partial \boldsymbol{b}}(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) = \nabla f(\boldsymbol{b}^{(k+1)}) + \gamma\left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k+1)}\right) = \gamma\left(\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}\right). \tag{52}$$

$$\frac{\partial L}{\partial \boldsymbol{u}}(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) = \partial g(\boldsymbol{u}^{(k+1)}) - \gamma\left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k+1)}\right).$$

Since $0 \in \partial g(\boldsymbol{u}^{(k+1)}) - \gamma\left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k+1)}\right)$, we have

$$\gamma\left(2\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}\right) \in \partial g(\boldsymbol{u}^{(k+1)}). \tag{53}$$

$$\frac{\partial L}{\partial \boldsymbol{m}}(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) = \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} = \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}. \tag{54}$$

Collecting Equations (52) - (54), we have

$$\boldsymbol{\varphi}_1^{(k+1)} := \begin{bmatrix} \gamma\left(\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}\right) \\ \gamma\left(2\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}\right) \\ \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)} \end{bmatrix}.$$

$$\boldsymbol{\varphi}_1^{(k+1)} \in \partial L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}). \tag{55}$$

Following (44), we arrive at the conclusion

$$||\boldsymbol{\varphi}_1^{(k+1)}||_2^2 \leq \sigma_2 ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2, \tag{56}$$

where $\sigma_2 > 0$. Hence, the desired condition is proved.

Proof (d): The augmented Lagrangian function $L_\gamma(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{m}) = f(\boldsymbol{b}) + g(\boldsymbol{u}) + \frac{\gamma}{2}||\boldsymbol{b} - \boldsymbol{u} + \boldsymbol{m}||_2^2 - \frac{\gamma}{2}||\boldsymbol{m}||_2^2$ defined as $L_\gamma : \mathbb{R}^n \to (-\infty, \infty]$ is proper and lower semi-continuous, where $f(\boldsymbol{b}) = ||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{b}||_2^2$, $g(\boldsymbol{u}) = ||\boldsymbol{u}||_1 + ||\boldsymbol{u}||_0$ and $j(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{m}) = \frac{\gamma}{2}||\boldsymbol{b} - \boldsymbol{u} + \boldsymbol{m}||_2^2 - \frac{\gamma}{2}||\boldsymbol{m}||_2^2$. If $L_\gamma(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{m})$ is semi-algebraic, then it satisfies the KŁ property at any point of its domain. Note that both f and j are real polynomial functions, which are semi-algebraic functions (Attouch et al., 2013; Bolte et al., 2014). Both $||\boldsymbol{u}||_0$ and $||\boldsymbol{u}||_1$ have piecewise linear graphs and are therefore semi-algebraic (see Example 3 and 4 in (Bolte et al., 2014), respectively).

Furthermore, consider that $g_1(\boldsymbol{u}) = \lambda\alpha||\boldsymbol{u}||_1$ and $g_2 = \lambda(1-\alpha)||\boldsymbol{u}||_0$. Their proximal operators have piecewise linear graphs and are perfectly known objects (Attouch et al., 2013; Beck, 2017). The proximal operator for $g_1(\boldsymbol{u}) = \lambda\alpha||\boldsymbol{u}||_1$, $\quad \text{prox}_{g_{1(\lambda\alpha)}}(\boldsymbol{u}) = [|\boldsymbol{u}| - \lambda\alpha]_+ \operatorname{sgn}(\boldsymbol{u})$ (the so-called soft thresholding function) is defined as

$$[|\boldsymbol{u}| - \lambda\alpha]_+ \operatorname{sgn}(\boldsymbol{u}) = \begin{cases} \boldsymbol{u} - \lambda\alpha, & \text{if } \boldsymbol{u} \geq \lambda\alpha, \\ 0, & \text{if } |\boldsymbol{u}| < \lambda\alpha, \\ \boldsymbol{u} + \lambda\alpha, & \text{if } \boldsymbol{u} \leq -\lambda\alpha. \end{cases}$$

Hence, $\text{prox}_{g_{1(\lambda\alpha)}}(\boldsymbol{u})$ has a piecewise-linear graph and is semi-algebraic. Now, we show that using the proximal operator for g_2, the prox of g_2 can be written as

$$\text{prox}_{g_{2(\lambda(1-\alpha))}}(\boldsymbol{u}) = \begin{cases} 0, & \text{if } |\boldsymbol{u}| < \sqrt{2\lambda(1-\alpha)}, \\ \boldsymbol{u}, & \text{if } |\boldsymbol{u}| > \sqrt{2\lambda(1-\alpha)}, \\ \{0, \boldsymbol{u}\}, & \text{if } |\boldsymbol{u}| = \sqrt{2\lambda(1-\alpha)}. \end{cases}$$

Clearly, $\text{prox}_{g_{2(\lambda(1-\alpha)}}(\boldsymbol{u})$ is also piecewise linear and semi-algebraic. Note that $\text{prox}_{g_{2(\lambda(1-\alpha)}}(\boldsymbol{u}) = \mathcal{H}_\kappa(\boldsymbol{u})$ the so-called hard thresholding operator, is defined as

$$\mathcal{H}_\nu(\boldsymbol{u}) \equiv \begin{cases} 0, & \text{if } |\boldsymbol{u}| < \nu, \\ \boldsymbol{u}, & \text{if } |\boldsymbol{u}| > \nu, \\ \{0, \boldsymbol{u}\}, & \text{if } |\boldsymbol{u}| = \nu, \end{cases} \tag{57}$$

where $\nu = \sqrt{2\lambda(1-\alpha)}$. Here, $g_1 + g_2$ is also semi-algebraic.

Consequently, for any nonnegative real numbers λ and α, the function $f(\boldsymbol{b}) + \lambda\alpha||\boldsymbol{u}||_1 + \lambda(1-\alpha)||\boldsymbol{u}||_0 + \frac{\gamma}{2}||\boldsymbol{b} - \boldsymbol{u} + \boldsymbol{m}||_2^2 - \frac{\gamma}{2}||\boldsymbol{m}||_2^2$ is semi-algebraic. Hence, we conclude that the Lagrangian function in (14) is a KŁ function.

Since $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}$ is bounded, there exists a subsequence $\{\boldsymbol{b}^{kl}, \boldsymbol{u}^{kl}, \boldsymbol{m}^{kl}\}$ converging to a stationary point $\{\boldsymbol{b}^*, \boldsymbol{u}^*, \boldsymbol{m}^*\}$, where $l \in \mathbb{N}$. Since the Lagrangian function in (14) is a KŁ function (using the lower semicontinuous property), we have

$$L_\gamma(\boldsymbol{b}^*, \boldsymbol{u}^*, \boldsymbol{m}^*) \leq \lim_{l \to \infty} L_\gamma(\boldsymbol{b}^{kl}, \boldsymbol{u}^{kl}, \boldsymbol{m}^{kl}). \tag{58}$$

In conclusion, all the conditions (a)-(d) in Theorem 1 hold.

C Appendix

Proof for Theorem 2: Starting with $r = 1$ (PRSM), we update $\boldsymbol{b}$, $\boldsymbol{m}$, and $\boldsymbol{u}$ iteratively according to (18a - 18d)

$$\boldsymbol{b}^{(k+1)} := \underset{\boldsymbol{b}}{\operatorname{argmin}}\ L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}), \tag{59a}$$

$$\boldsymbol{m}^{(k+\frac{1}{2})} := \boldsymbol{m}^{(k)} + \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k)}, \tag{59b}$$

$$\boldsymbol{u}^{(k+1)} := \underset{\boldsymbol{u}}{\operatorname{argmin}}\ L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k+\frac{1}{2})}), \tag{59c}$$

$$\boldsymbol{m}^{(k+1)} := \boldsymbol{m}^{(k+\frac{1}{2})} + \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}. \tag{59d}$$

Proof (a): From (59a), since $\boldsymbol{b}^{(k+1)}$ minimizes $L_r(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)})$ and Lagrangian is strongly convex with respect to the variable $\boldsymbol{b}$, (38) holds.

Next, using the augmented Lagrangian function in (14), we compute

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{k+\frac{1}{2}}) = \gamma \left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)}\right)^\intercal \left(\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}\right). \tag{60}$$

Now we rewrite (59d) as

$$\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} = \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}. \tag{61}$$

From (59c) and (19a), we obtain

$$\nabla f(\boldsymbol{b}^{(k+1)}) + \gamma(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{k+\frac{1}{2}}) = 0. \tag{62}$$

Substituting (59d) into (62), we obtain (42). Again, from (59a) and (19a), we obtain

$$\nabla f(\boldsymbol{b}^{(k+1)}) + \gamma(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k)} + \boldsymbol{m}^{(k)}) = 0. \tag{63}$$

Substituting (59b) into (63), we obtain

$$\nabla f(\boldsymbol{b}^{(k+1)}) = -\gamma \boldsymbol{m}^{k+\frac{1}{2}}. \tag{64}$$

Using (42),(61) and (64), (60) becomes

$$\gamma \left(||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}||^2\right) = \gamma \left(|| - \frac{1}{\gamma}\nabla f(\boldsymbol{b}^{(k+1)}) + \frac{1}{\gamma}\nabla f(\boldsymbol{b}^{(k+1)})||^2\right) = 0. \tag{65}$$

Hence,

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{k+\frac{1}{2}}) \leq 0. \tag{66}$$

Using (14), we have

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{k+\frac{1}{2}}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) = \gamma \left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k)}\right)^\intercal \left(\boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k)}\right). \tag{67}$$

Next, we reformulate (59b) as

$$\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k)} = \boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k)}. \tag{68}$$

Using (64) and (68), (67) becomes

$$\gamma \left(||\boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k)}||^2\right) = \gamma \left(|| - \frac{1}{\gamma}\nabla f(\boldsymbol{b}^{(k+1)}) + \frac{1}{\gamma}\nabla f(\boldsymbol{b}^{(k)})||^2\right). \tag{69}$$

Therefore,

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{k+\frac{1}{2}}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \leq \frac{l_f^2}{\gamma} \left(||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||^2\right), \tag{70}$$

where $l_f \geq 0$ is a Lipschitz gradient of the function $f(\boldsymbol{b})$. From (59c) we have that

$$L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{k+\frac{1}{2}}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{k+\frac{1}{2}}) \leq 0. \tag{71}$$

Finally, combining (38), (66), (70) and (71), we get the desired inequality as follows.

$$\begin{aligned}
&L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \\
&= L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{k+\frac{1}{2}}) \\
&\quad + L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{k+\frac{1}{2}}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{k+\frac{1}{2}}) \\
&\quad + L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{k+\frac{1}{2}}) - L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \\
&\quad + L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) - L_\gamma(\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}) \\
&\leq \left(\frac{l_f^2}{\gamma} - \frac{\rho}{2}\right) ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2 \\
&= -\delta_1 ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2,
\end{aligned}$$

which is the sufficient decreasing condition (46).

Proof (b): The difference lies in some steps to show the boundedness of $\boldsymbol{u}^{(k)}$ and $\boldsymbol{m}^{(k)}$. The rest is the same as in the proof of Theorem 1(b). From (61) and (68) we obtain $\boldsymbol{u}^{(k+1)} = \boldsymbol{b}^{(k+1)} - \boldsymbol{m}^{(k+1)} + \boldsymbol{m}^{k+\frac{1}{2}}$ and $\boldsymbol{u}^{(k)} = \boldsymbol{b}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}} + \boldsymbol{m}^{(k)}$, respectively. Then

$$\begin{aligned}
||\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}||_2^2 &= ||\boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k+1)} + \boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k)}||_2^2 \\
&\leq ||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}||_2^2 + ||\boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k)}||_2^2,
\end{aligned}$$

This inequality can be rewritten using (65) and (69) as

$$||\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}||_2^2 \leq \frac{l_f^2}{\gamma^2} ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2.$$

Consequently, we obtain

$$\sum_{k=0}^{\infty} ||\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}||_2^2 < \infty. \tag{72}$$

Equation (72) implies that $\boldsymbol{u}^{(k)}$ is bounded. To show that $\boldsymbol{m}^{(k)}$ is bounded, we analyze the difference $\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}$ as follows. $\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)} = \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}} + \boldsymbol{m}^{k+\frac{1}{2}} - \boldsymbol{m}^{(k)}$. Then, we obtain

$$\begin{aligned}
||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}||_2^2 &\leq \frac{l_f^2}{\gamma^2} ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2, \\
\sum_{k=0}^{\infty} ||\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{(k)}||_2^2 &< \infty.
\end{aligned} \tag{73}$$

Hence, $\boldsymbol{m}^{(k)}$ is bounded.

Proof (c):

$$\frac{\partial L}{\partial \boldsymbol{b}}(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) = \nabla f(\boldsymbol{b}^{(k+1)}) + \gamma \left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k+1)}\right) = \gamma \left(\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}\right). \tag{74}$$

$$\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{u}}(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) &= \partial g(\boldsymbol{u}^{(k+1)}) - \gamma \left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k+1)}\right). \\
\gamma \left(2\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}\right) &\in \partial g(\boldsymbol{u}^{(k+1)}).
\end{aligned} \tag{75}$$

Equation (75) holds because $0 \in \partial g(\boldsymbol{u}^{(k+1)}) - \gamma\left(\boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} + \boldsymbol{m}^{(k+1)}\right)$.

$$\frac{\partial L}{\partial \boldsymbol{m}}(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}) = \boldsymbol{b}^{(k+1)} - \boldsymbol{u}^{(k+1)} = \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}. \tag{76}$$

Collecting Equations (74) - (76), we have

$$\boldsymbol{\varphi}_2^{(k+1)} := \begin{bmatrix} \gamma\left(\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}\right) \\ \gamma\left(2\boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}}\right) \\ \boldsymbol{m}^{(k+1)} - \boldsymbol{m}^{k+\frac{1}{2}} \end{bmatrix}.$$

$$\boldsymbol{\varphi}_2^{(k+1)} \in \partial L_\gamma(\boldsymbol{b}^{(k+1)}, \boldsymbol{u}^{(k+1)}, \boldsymbol{m}^{(k+1)}). \tag{77}$$

Hence, the desired condition is proved. Using Equation (44) with a parameter $\sigma_2 > 0$, we obtain

$$||\boldsymbol{\varphi}_2^{(k+1)}||_2^2 \leq \sigma_2 ||\boldsymbol{b}^{(k+1)} - \boldsymbol{b}^{(k)}||_2^2. \tag{78}$$

Hence, the desired condition is proved.

Proof (d): See the proof of Theorem 1 (d).

For the case $r \in (0, 1)$, all conditions are valid. Hence, for the sequences $\{\boldsymbol{b}^{(k)}, \boldsymbol{u}^{(k)}, \boldsymbol{m}^{(k)}\}_{k=0}^{t}$ generated by the SCPRSM scheme (18a - 18d), and its Lagrangian given by (14), the four conditions from Theorem 1 (a)-(d) hold, achieving a worst-case convergence rate of $O(\frac{1}{k})$. Here, a worst-case $O(\frac{1}{k})$ convergence rate indicates that the solution's accuracy, based on specific criteria, improves gradually at a rate proportional to one divided by the number of iterations (k) within an iterative algorithm (He et al., 2014).

D Figure 4

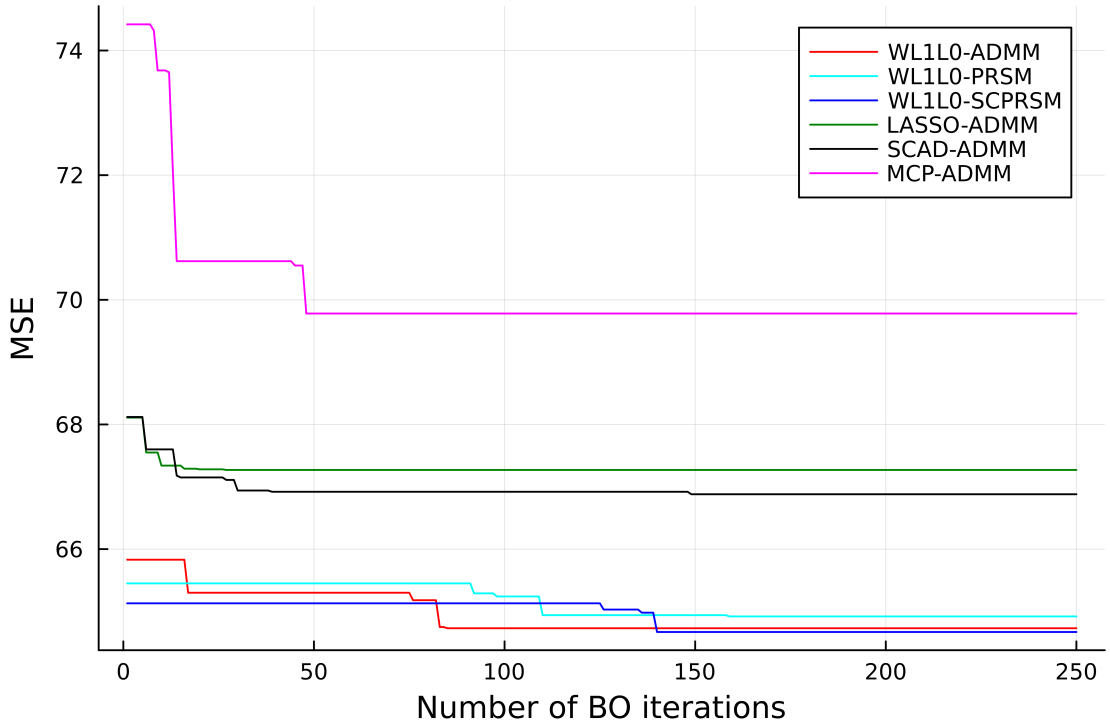

Figure 4: Convergence Speed of MSEs of the Simulated QTLMAS2010 Data.