Statistical Papers (2022) 63:53–95
https://doi.org/10.1007/s00362-021-01229-0

REGULAR ARTICLE

Penalized and constrained LAD estimation in fixed and high dimension

Xiaofei Wu[1] · **Rongmei Liang**[1] · **Hu Yang**[1]

Received: 6 August 2020 / Revised: 27 February 2021 / Published online: 31 March 2021

Abstract

Recently, many literatures have proved that prior information and structure in many application fields can be formulated as constraints on regression coefficients. Following these work, we propose a L_1 penalized LAD estimation with some linear constraints in this paper. Different from constrained lasso, our estimation performs well when heavy-tailed errors or outliers are found in the response. In theory, we show that the proposed estimation enjoys the Oracle property with adjusted normal variance when the dimension of the estimated coefficients p is fixed. And when p is much greater than the sample size n, the error bound of proposed estimation is sharper than $\sqrt{k\log(p)/n}$. It is worth noting the result is true for a wide range of noise distribution, even for the Cauchy distribution. In algorithm, we not only consider an typical linear programming to solve proposed estimation in fixed dimension , but also present an nested alternating direction method of multipliers (ADMM) in high dimension. Simulation and application to real data also confirm that proposed estimation is an effective alternative when constrained lasso is unreliable.

Keywords High dimensional regression · Linear constraints · Variable selection · LADLasso · Oracle property · ADMM

This work is supported by the National Natural Science Foundation of China (Grant No. 11671059) and graduate scientific research and innovation foundation of Chongqing China (Grant No.CYS20041) .

✉ Hu Yang
yh@cqu.edu.cn

Xiaofei Wu
xfwu1016@163.com

[1] College of Mathematics and Statistics, Chongqing University, Chongqing 401331, People's Republic of China

1 Introduction

Motivated by applications in areas as diverse as finance, image reconstruction, and curve estimation, many literatures begin to focus on constrained lasso (hereinafter referred to as classo), such as He (2011), James et al. (2013), Zhou and Lange (2013), Hu et al. (2015b), Gaines et al. (2018), James et al. (2020), etc. Classo is defined as:

$$\arg\min_{\beta} \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + n \sum_{j=1}^{p} \lambda_j \left|\beta_j\right| \text{ subject to } C_1\beta = b_1 \text{ and } C_2\beta \leq b_2, \quad (1)$$

where y_i is the ith element of $y = (y_1, y_2, \ldots, y_n)'$, x_i is the ith row of design matrix $X = (x_1', x_2', \ldots, x_n')'$. We assume that every column of X has been standardized and the constrained matrixs C_1, C_2 have full row rank. λ_j is the penalty level (tuning parameter) which is always nonnegative.

Classo is a very flexible framework for imposing additional knowledge and structure onto the lasso coefficient estimates. This feature makes it have a very wide range of applications. For instance, in economics when people predictor the car sale, one important predictor is personal income. With the increase of income, the amount of sale of cars also increases. The personal income cannot have negative impacts on car price. Therefore non-negativity constraints need to imposed on the corresponding regression efficients. This nonnegative effects are also applied to stock index tracking, because the impact of each component stock on the stock index can not be negative. Another famous example in which linear constraints need to be utilized is the case of isotonic regression. The problem has a unique property that if $x_i \leq x_j$, then $x_i\beta \leq x_j\beta$. In many fields of genomic data analysis, much biological knowledge or pathway information is available. This kind of information has been accumulated from years of biological and medical research and is a precious resource supplementary to statistical gene data analysis. More applicable situations using the classo can be referred to Gaines et al. (2018) and James et al. (2020). However, from James et al. (2013) we know that the near Oracle performance of classo relies heavily on the Gaussian assumptions and a known variance σ^2. In practice, the Gaussian assumption may not hold and the estimation of the standard deviation σ is not a trivial problem. Moreover, in some cases where heavy-tailed errors or outliers are found in the response, the variance of the errors may be unbounded. In this case, the classo method is no longer applicable.

To deal with these problems, we propose the following L_1 penalized constrained least absolute deviation estimation (hereinafter referred to as pcLAD),

$$\arg\min_{\beta} \sum_{i=1}^{n} \left|y_i - x_i'\beta\right| + n \sum_{j=1}^{p} \lambda_j \left|\beta_j\right| \text{ subject to } C_1\beta = b_1 \text{ and } C_2\beta \leq b_2. \quad (2)$$

The least absolute deviation (LAD) type of methods are effective alternative to the least square methods since it doesn't require the distribution of errors. When heavy-tailed errors or outliers are present, these methods have desired robust properties in linear regression models, see for example Bassett and Koenker (1978), Huber (1981), Portnoy and Koenker (1997).

Recently, the penalized version of the LAD method was studied in several papers and the variable selection and estimation properties were discussed. When the dimension of coefficients p is assumed to be fixed, the consistency of the penalized LAD estimator has been proven, one can see Wang et al. (2007), Lambert-Lacroix and Zwald (2011), Wu and Liu (2009). When p is high dimension, Gao and Huang (2010), Belloni and Chernozhukov (2011), Wang (2013) has showed the properties of penalized LAD method in different assumptions. It's remarkable that, Wang (2013) proposed a clear and practical rule for setting the penalty parameter and a sharp bound of estimation error. That is,

$$\lambda_j = c\sqrt{2A(\alpha)\log(p)/n}, \tag{3}$$

$$\left\|\hat{\beta}_{pLAD} - \beta\right\|_2 = O(\sqrt{k\log(p)/n}), \tag{4}$$

where $c > 1$ is a constant, α is a chosen small probability, and $A(\alpha)$ is a constant such that $2p^{-(A(\alpha)-1)} \leq \alpha$. k is the number of nonzero or significant true coefficients.

In this paper, when p is fixed, we use the λ_j suggested by Wang et al. (2007) and develop Oracle property of the equality constrained pcLAD similar to LADlasso. Because of the existence of constraints, the asymptotically normalized variance of pcLAD has an adjustment item compared to that of LADlasso. This adjustment will make pcLAD estimation more effective than LADlasso. When p is large then n , we adopt the form of λ_j in (3), and obtain a L_2 norm of equality constrained pcLAD estimation error bound

$$\left\|\hat{\beta} - \beta\right\|_2 = O(\sqrt{\max(m, k-m)\log(p)/n}), \tag{5}$$

where m is the number of equality constraints and should be less than k. The pcLAD estimation bound have a similar form as (4), but our bound clearly demonstrates the potential improvements in accuracy that can be derived from adding constraints. We can also point out this pcLAD will choose the significance coefficient with a high probability close to 1. For inequality constrained pcLAD, we can get same result if there are some constraints at the boundary. It is worth noting that all the above theoretical results do not assume the error distribution, which makes pcLAD have a good fitting effect in the presence of heavy-tailed errors and outliers.

Compared with least square method, LAD method is freer of error distribution. However, it is more difficult to be solved due to its unsmooth loss function. In the computation of fixed dimensional pcLAD model, a typical approach is to modify the computing method of Wang et al. (2007), which is also used in Gao and Huang (2010) and Wang (2013) when p is larger than n. That is, $Y_{n+j} = 0$ and $x_{n+j,j} = \lambda_j \times I(j = i)$ for $i, j = 1, 2, \ldots p$. Here $I(j = i)$ is the indicator function such that $I(j = i) = 1$ if $j = i$ and $I(j = i) = 0$ if not. Then our pcLAD estimator can be considered as an ordinary LAD estimator satisfying some liner constraints with p unknown coefficients and $p + n$ observations. Hence it can be solved efficiently by **R** package *quantreg*. More details about this linear programing can be found in Sect. 4.1. However, as Yang et al. (2013), Gu et al. (2017) and Yu and Lin (2017) point out, LP scales well to data with moderate sizes, it still comes short

when dealing with high dimensions. This observation motivates us to consider an efficient method to fit the high dimensional constrained LAD regression. Fortunately, alternating direction method of multipliers (ADMM) has been proved to be able to deal with high-dimensional constrained optimization, such as Gaines et al. (2018), Stellato et al. (2018) and so on. Inspired by these work, we propose a nested ADMM to solve pcLAD. In nested ADMM algorithm, the first update is unconstrained LAD regression with combined penalty term, which is solved effectively by ADMM, the second update is a projection onto the affine constrained space and the third update is the renewal of dual variables. Since the first update is a complete ADMM iteration, we call it nested ADMM. Although nested ADMM contains an inner iteration and an outer iteration, every step has an explicit solution. Thus, pcLAD can be calculated fast by nested ADMM. A lot of numerical experiments in Sect. 5 can also confirm this.

Importantly, pcLAD can solve almost all problems that classo can be applied to, such as monotone curve estimation, monotonic order regression estimation, sum to zero or one estimation, and all problems that can be transformed into generalized lasso, fused lasso, nonnegative lasso etc.. Furthermore, when the noise of the above problems does not obey Gaussian distribution, pcLAD is more robust and reliable than classo.

This paper is organized as follows. In Sect. 2, we provide a number of motivating examples which illustrate the wide range of situations where the pcLAD is applicable. Section 3 discusses pcLAD theoretical properties when p is fixed and high dimension. LP and ADMM algorithms are described in detail in Sect. 4. Section 5 will compare the performance of above two algorithms in different dimensions, and present three data simulations which show the pcLAD will do a good job when classo is unreliable. In Sect. 6, some real data examples implie that the pcLAD method has a better performance than classo in applications. We conclude with a discussion about future extensions of this work in Sect. 7. Technical lemmas and proofs of theorems are given in appendix.

2 Motivagting examples

In this section, we will briefly show some applied statistical problems solved by classo that can also be solved by pcLAD.

2.1 Monotone curve fitting

Consider the problem of fitting a smooth function $l(x)$, to a set of observations $\{(x_1, y_1), \ldots,(x_n, y_n)\}$, subject to the constraint that l must be monotone. James et al. (2020) has shown that classo can be applied to monotone curve fitting. We can replace the $g(\beta) = \sum_{i=1}^{n} (y_i - B(x_i)'\beta)^2$ as $g(\beta) = \sum_{i=1}^{n} \left|y_i - B(x_i)'\beta\right|$, then we need to minimize $g(\beta) = \sum_{i=1}^{n} \left|y_i - B(x_i)'\beta\right|$ subject to $C\beta \leq 0$, where the tth row of C is the derivative $B'(v_t)$ of the basis functions evaluated at v_t for a fine grid of points, $v_1, \ldots, v_m$, over the range of x. Enforcing this constraint ensures that the derivative

of l is non-positive, so l will be monotone decreasing. Obviously, this model can be addressed using the pcLAD methodology.

2.2 Monotonic order estimation

Isotonic regression is a monotonic order estimate studied by many literatures such as Wu et al. (2001), Tibshirani et al. (2011), Gaines et al. (2018), etc.. The lasso with a monotonic ordering of the coefficients was referred to by Tibshirani and Suo (2016) as the ordered lasso. Gaines et al. (2018) has implied that both of the above estimates can be solved by classo. Next, we will show the monotonic order estimation can be also solved by pcLAD. Consider pcLAD without equality constraints as follows :

$$\arg\min_{\beta} \sum_{i=1}^{n} \left| y_i - x_i'\beta \right| + n \sum_{j=1}^{p} \lambda_j \left| \beta_j \right| \text{ subject to } C_m\beta \leq 0. \tag{6}$$

When x_i' is the i row of identity matrix, $\lambda_j = 0$ for any j, and the constraints matrix is

$$C_m = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{pmatrix}. \tag{7}$$

The formula (6) will become the LAD isotonic regression. When the $\lambda_j > 0$ for any j and the constraints matrix is same the matrix as before, it will be a monotonic order LADlasso. Indeed, there are many other options of the constraints for the monotonic order estimation. For example, C_m can also defined as follows.

$$C_m = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix}.$$

This C_m makes the estimation coefficient monotone decreasing. Other options can also be limited to some certain coefficients, such as

$$\beta_1 \leq \beta_3, \beta_2 \leq \beta_4, \beta_3 \leq \beta_6.$$

Furthermore, we can also obtain the LAD version of order lasso (Tibshirani and Suo 2016) by rewriting β into $\beta^+ - \beta^-$ and adding monotonoic order to β^+ and β^-.

2.3 Generalized LADlasso and LAD fused lasso

Gaines et al. (2018) and James et al. (2020) have proved that generalized lasso (Tibshirani and Taylor 2011) can be transformed into classo. We also consider the following

the generalized LADlasso problem:

$$\arg\min_{\beta} \|y - X\beta\|_1 + n\lambda \|D\beta\|_1 , \tag{8}$$

where $D \in R^{r\times p}$, $\text{rank}(D) = r$.

The following lemma will show the generalized LADlasso can also be transformed into pcLAD.

Lemma 1 *If $r \le p$, (8) can be converted to the classical LADlasso problem. If $r > p$ and $rank(D) = p$, then there exist matrix C, F, and $\tilde{X}$ such that, for all values of λ, the solution to (8) is equal to $\beta = F\theta$, where θ is given by:*

$$\arg\min_{\theta} \left\| y - \tilde{X}\theta \right\|_1 + n\lambda \|\theta\|_1 \ \textit{subject to}\ C\beta = 0. \tag{9}$$

The proof of Lemma 1 is provided in Appendix A. Hence, any problem that falls into the generalized LADlasso can be solved by pcLAD.

LAD fused lasso is the LAD version of fused lasso (Tibshirani et al. 2005), it is defined as the solution to

$$\arg\min_{\beta} \|y - X\beta\|_1 + n\sum_{j=1}^{p} \lambda_j \left|\beta_j\right| + n\sum_{j=2}^{p} \gamma_j \left|\beta_j - \beta_{j-1}\right|. \tag{10}$$

It's very easy to know LAD fused lasso is a special case of the generalized LADlasso (8) with the equality penalty matrix D as

$$\begin{pmatrix} -C_m \\ I_p \end{pmatrix} \in R^{(2p-1)\times p},$$

where I_p is the $p \times p$ identity matrix.

The fused LADlasso encourages blocks of adjacent estimated coefficients to all have the same value. This type of structure often makes sense in situations where there is a natural ordering in the coefficients. Similar to James et al. (2013), if the data have a two-dimensional ordering, such as for an image reconstruction, this idea can be extended to the 2d fused LADlasso

$$\arg\min_{\beta} \|y - X\beta\|_1 + n\left(\sum_{j,j'} \lambda_{j,j'} \left|\beta_{j,j'}\right| + \sum_{j\neq j'} \gamma_{j,j'} \left|\beta_{j,j'} - \beta_{j,j'-1}\right| + \sum_{j\neq j'} \eta_{j,j'} \left|\beta_{j,j'} - \beta_{j-1,j'}\right|\right)$$

2.4 Nonnegative sparse estimation

The most common non negative sparse estimation is nonnegative lasso. It appeared in a lot of literatures. First mentioned in the seminal work of Efron et al. (2004), the positive lasso requires the lasso coefficients to be nonnegative. This variant of the lasso has seen applications in areas such as vaccine design (Hu et al. 2015a), nuclear material detection (Kump et al. 2012), document classification (El-Arini et al. 2013), and portfolio management (Wu et al. 2014). Many other nonnegative sparse estimators have been proposed such as Yang and Wu (2016), Wu and Yang (2014), Mandal and Ma (2016), Li et al. (2019), Xie and Yang (2019), Li and Yang (2019), etc.. However, the LAD version of non negative lasso is not appeared in the literature. In the discussion of Wang (2013), we know that LAD method can process a wide range of non Gaussian observations, so it is necessary to propose some non negative sparse LAD methods.

The first non negative sparse LAD method proposed in this section is non negative LADlasso:

$$\arg\min_{\beta\geq 0} \|y - X\beta\|_1 + n\sum_{j=1}^{p} \lambda_j \left|\beta_j\right|. \tag{11}$$

Obviously, non negative LADlasso is pcLAD (2) with $C_1 = -I_p$ and $b_1 = 0_p$.

The other is non negative fused LADlasso:

$$\arg\min_{\beta\geq 0} \|y - X\beta\|_1 + n\sum_{j=1}^{p} \lambda_j \left|\beta_j\right| + n\sum_{j=2}^{p} \gamma_j \left|\beta_j - \beta_{j-1}\right|. \tag{12}$$

As discussed in Sect. 2.3, non negative fused LADlasso can be transformed into non negative generalized LADlasso, which is a special case of pcLAD.

Although four examples are listed, there are still many statistical problems that can be solved by pcLAD, such as sum to zero regression , sum to one regression, relaxed lasso, sign-constrained least square regression, etc.. One can find more details about these methods by referring to Shi et al. (2016), Meinshausen (2007), Meinshausen (2013).

3 Statistical properties

In this section, we firstly discuss the statistical properties of pcLAD with equality constraints when the dimension of estimation p is fixed and much larger than sample size n. As discussed in Sect. 1, we use adaptive L_1 penalty (Wang et al. 2007; Zou 2006), which is more general penalty than L_1 penalty (Tibshirani 1996). When the p is high dimensional setting, the calculation of adaptive L_1 penalty parameters are complicated and time-consuming, therefore we adopt ordinary L_1 penalty parameters suggested by Wang (2013). We assume that $y \in R^n$ is generated from:

$$y_i = x_i'\beta + \varepsilon_i, i = 1, 2, \ldots, n. \tag{13}$$

Where $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})'$, $\beta \in R^p$, and $\varepsilon = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)'$ are $i.i.d.$ median-zero random variables. Let $\hat{\beta}$ denote a solution of pcLAD defined by:

$$\hat{\beta} = \arg\min_{\beta} Q(\beta) \text{ subject to } C\beta = b, \tag{14}$$

where $Q(\beta) = \sum_{i=1}^{n} \left| y_i - x_i'\beta \right| + \sum_{j=1}^{p} \lambda_j \left| \beta_j \right|$. If the solution (14) is not unique, we can take $\hat{\beta}$ to be any optimal solution, our statistical properties hold for all such solutions.

3.1 *p* is fixed

For convenience, we decompose the true regression coefficient as $\beta_0 = (\beta_{0A}', \beta_{0B}')'$, where $\beta_{0A} = (\beta_{01}, \beta_{02}, \ldots, \beta_{0k})$ are k ture significant coefficients and $\beta_{0B} = (\beta_{0(k+1)}, \beta_{0(k+2)}, \ldots, \beta_{0p})$ are $p-k$ ture insignificant coefficients. Moreover, assume that $\beta_{0j} \neq 0$ for $1 \leq j \leq k$ and $\beta_{0j} = 0$ for $k < j \leq p$. Its corresponding pcLAD estimator is denoted $\hat{\beta} = (\hat{\beta}_A', \hat{\beta}_B')'$. We also decompose the covariate $x_i = (x_{iA}', x_{iB}')$ with $x_{iA} = (x_{i1}, x_{i2}, \ldots, x_{ik})'$ and $x_{iB} = (x_{i(k+1)}, x_{i(k+2)}, \ldots, x_{ip})'$. In addition, constraint matrix C can be rewritten as $C = (C_A, C_B)$. To study the theoretical properties of pcLAD in fixed dimension, the following technical assumptions are necessarily needed:

Assumption 1 The error ε_i has continuous and positive density at the origin, that is $f(0) > 0$.

Assumption 2 The design x_i, $i = 1, 2, \ldots, n$, satisfies the limit of $\sum_{i=1}^{n} x_i x_i'/n \to \Sigma$ as $n \to \infty$. Denote the top-left $k-by-k$ submatrix of Σ by Σ_{11}, and the right-bottom $(p-k)-by-(p-k)$ submatrix of Σ by Σ_{22}.

Assumption 3 There is a nonsingular submatrix in C_A, which is denoted as C_{A_1}. The size of index set A_1 should be equal to be the row rank of C, that is m.

Note that Assumptions 1 and 2 are both very typical technical assumptions used extensively in the sparse estimation in fixed dimension such as Fan and Li (2001), Wang et al. (2007), Wu and Liu (2009). Assumptions 3 is required in classo (James et al. 2013).

Furthermore, define $a_n = \max\{\lambda_j, 1 \leq j \leq k\}$ and $b_n = \min\{\lambda_j, k < j \leq p\}$, where λ_j is a function of n. Based on the foregoing notation, the consistency of pcLAD estimator can be first established.

Lemma 2 (Consistency) *Consider a sample* $\{(x_i, y_i), i = 1, 2, \ldots, n\}$ *from model (13) satisfying Assumption 1 and 2 with* $i.i.d.$ ε_i' *s. If* $\sqrt{n}a_n \to 0$, *as* $n \to \infty$, *there exists a pcLAD estimation* $\hat{\beta}$ *such that* $\left\| \hat{\beta} - \beta_0 \right\|_2 = O_P(n^{-\frac{1}{2}})$.

$\sqrt{n}$-consistency is a common property in constrained LAD estimation such as Wang (1995), Geyer (1994), Silvapulle and Sen (2005), and Parker (2019). Lemma 2

show that linear constrained LADlasso also enjoys this nice property. Under some further conditions, the sparsity property of the pcLAD estimator can be obtained as in Lemma 3.

Lemma 3 (Sparsity) *Consider a sample* $\{(x_i, y_i), i = 1, 2, \ldots, n\}$ *from model (13) satisfying Assumption 1 and 2 with i.i.d.* ε_i' *s. If* $\sqrt{n}b_n \to \infty$, *as* $n \to \infty$, *for any given* β, *satisfying* $\|\beta_A - \beta_{A0}\|_2 = O_P(n^{-\frac{1}{2}})$, $C(\beta - \beta_0) = 0$. *Then, with probability trending to 1, for any constant* $R > 0$, $Q((\beta_A', 0')') = \min_{|\beta_2| \le Rn^{-1/2}} Q((\beta_A', \beta_B')')$.

The Lemmas 2 and 3 are common results of many sparse estimations in fixed dimension. There is no doubt that they are very nice properties, but not reflect the influence of constraints. Our next theorem will show influence of constraints and illustrate the pcLAD estimator also enjoy the popular asymptotic Oracle property.

Theorem 1 (Oracle) *For a sample* $\{(x_i, y_i), i = 1, 2, \ldots, n\}$ *from model (13) satisfying Assumption 1, 2 and 3 with i.i.d.* ε_i' *s. if* $\sqrt{n}a_n \to 0$, $\sqrt{n}b_n \to \infty$, *as* $n \to \infty$, *then with probability trending to one, the consistent pcLAD estimation* $\hat{\beta}=(\hat{\beta}_A', \hat{\beta}_B')'$ *in Lemma 2 must be satisfy:*

$\langle a \rangle$ *Sparsity:*$\hat{\beta}_B=0$.

$\langle b \rangle$ *Asymptotic normality:* $\sqrt{n}(\hat{\beta}_A - \beta_{A0}) \xrightarrow{L} N(0, \frac{\sum_{11}^{-1}}{4f^2(0)}(I - V)'(I - V))$, *where* $\sum_{11}$ *is defined in Assumption 2,* $V = C_{A_1}'(C_{A_1}\sum_{11}^{-1}C_{A_1}')^{-1}C_{A_1}\sum_{11}^{-1}$ *and* $\xrightarrow{L}$ *represents convergence in distribution.*

The proof details of Lemma 2, 3 and Theorem 1 can be found in Appendix B.

Remark 1 Theorem 1 is a novel conclusion which reflects the influence of constraints on the asymptotic distribution of significant coefficients estimation. It is easy to see the variance of pcLAD estimation error is numerically smaller than LADlasso. This result is not surprising because the prior information to the model has been used.

3.2 *p* is high dimensional

Due to using the ordinary L_1 penalty, the high dimensional pcLAD can be rewritten as:

$$\hat{\beta} = \arg\min_{\beta} \|y - X\beta\|_1 + n\lambda \|\beta\|_1 \text{ subject to } C\beta = b, \tag{15}$$

where C has full row rank and $\text{rank}(C) = m$.

We adopt the Assumption 1, 3 and the notation in Sect. 3.1. We can also decompose the constrained matrix C_A as (C_{A_1}, C_{A_2}). In practical application, the equality constraint parameters before the insignificant coefficients do not work and are usually set to 0 that is $C_B = \mathbf{0}$. This setting is not only in line with the actual situation, but also necessary. If the corresponding penalty submatrix of the insignificant coefficients $C_B \neq \mathbf{0}$, then it must be found that an insignificant estimated coefficient can be linearly expressed by the significant estimated coefficients, which will cause the estimation value of this insignificant coefficient to become non-zero. In this way, the accuracy of

variable selection of pcLAD model will be greatly reduced. A naive method to avoid this is to increase the penalty parameter so that the insignificant coefficients (usually the estimated value is not too large) are completely shrunk to 0, but this will also lead to the bias of the significant estimated coefficients. Thus, if one wants to apply the λ suggested by Wang (2013) to pcLAD, $C_B = \mathbf{0}$ is indispensable.

It is worth noting that there is a special case that does not meet the above setting, that is relaxed lasso constraint (Meinshausen 2007). This constraint is defined as follows,

$$\beta_M = \mathbf{0}, \text{ where } M \in B. \tag{16}$$

Because in this setting, $C_A = \mathbf{0}$, then insignificant estimated coefficients are not linearly expressed by the significant estimated coefficients. In order to include this special case in pcLAD, when the constraint of $\beta_M = \mathbf{0}$ exists, the constraint of $C_B = \mathbf{0}$ can be transformed into the constraint of $C_{B/M} = \mathbf{0}$.

Decompose $\beta^{'} = (\beta^{'}_{A_1}, \beta^{'}_{A_2}, \beta^{'}_{M}, \beta^{'}_{B/M})$, and consider $\beta_M = \mathbf{0}$ and $C_{B/M} = \mathbf{0}$, then we get the equation:

$$C_{A_1}\beta_{A_1} + C_{A_2}\beta_{A_2} = b. \tag{17}$$

In order to do prove the near oracle property of pcLAD estimator $\hat{\beta}$, we need to present a lemma related to the estimation error $h = \beta_0 - \hat{\beta}$. In what follows, let the vector h_A be defined as: if the index i is in the index set A, the i th element of h_A is the same as that of h; otherwise, the i th element of h_A is 0. By (17) and Assumption 3, we will get $h_{A_1} = -\left(C_{A_1}\right)^{-1}(C_{A_2}h_{A_2})$, where C_{A_1} is $m \times m$ matrix and C_{A_2} is $m \times (k-m)$ matrix. In high dimensional statistics, as described by Bhlmann and van de Geer (2011), it is generally required to: $k\log(p) \ll n$. As Gu and Zou (2020) points out, in the lasso framework, the dimension of LAD estimation can reach the order $e^{n^{\pi}}$, where $0 < \pi < 1$. So, we need to assume $k < \infty$, that is $k = O(1)$ which will always satisfy $k\log(p) \ll n$. By synthesizing $m < k$, we can obtain that there is always a constant $\Phi > 0$, such that $\left\|h_{A_1}\right\|_1 \le \Phi\left\|h_{A_2}\right\|_1$ and $\left\|h_{A_1}\right\|_2 \le \Phi\left\|h_{A_2}\right\|_2$. Then we can get a lemma about the cone constraint of h.

Lemma 4 *Suppose* $\lambda = c\sqrt{2A(\alpha)\log(p)/n}$, *let* $\Delta_{\bar{c}} = \left\{\delta \in R^p : \left\|\delta_{A_2}\right\|_1 \ge \frac{\bar{c}}{1+\Phi}\|\delta_B\|_1\right\}$. *Then* $h \in \Delta_{\bar{c}}$, *where* $\bar{c} = \frac{(c-1)}{(c+1)}$.

The proof details of Lemma 4 can be found in the Appendix A. This cone constraint is extremely important for high dimensional estimation error bounds, one can see it in the classical lasso, square root lasso, LADlasso (quantile lasso) and constrained lasso, for example, Bickel et al. (2009), Wang (2013), Belloni et al. (2011), James et al. (2013), Gu and Zou (2020).

Now we introduce some restricted eigenvalue concepts on the design matrix X, based on L_2 norm to prepare for the analysis of near Oracle property of the pcLAD estimator. Let λ^u_k be the smallest number such that for any k sparse vector d:

$$\|Xd\|_2^2 \le n\lambda^u_k\|d\|_2^2; \tag{18}$$

also let λ_k^u be the largest number such that for any k sparse vector d:

$$\|Xd\|_2^2 \geq n\lambda_k^l \|d\|_2^2 . \tag{19}$$

Let θ_{k_1,k_2} be the smallest number such that for any k_1 and k_2 sparse vector c_1 and c_2 with disjoint support,

$$|(Xc_1, Xc_2)| \leq n\theta_{k_1}^{k_2} \|c_1\|_2 \|c_2\|_2. \tag{20}$$

The above concepts λ_k^u and $\theta_{k_1}^{k_2}$ are related to the sparse recovery conditions in the compressed sensing (CS). See Wang (2013) for more details. For the pcLAD model, we define a concept on restricted eigenvalues of design matrix X based on L_1 norm as:

$$k_k^l(\bar{c}) = \min_{h \in \Delta_{\bar{c}}} \frac{\|Xh\|_1}{n\|h_A\|_2}. \tag{21}$$

To simplify the notations, we will simply write $k_k^l(\bar{c})$ as k_k^l.

In order to formulate our main result, we also need the following condition:

$$\frac{3}{16}\sqrt{n}k_k^l > \lambda\,(1+\Phi)\sqrt{\frac{k-m}{n}} + c_1\sqrt{2\max(m,k-m)\log(p)}\left(\frac{5}{4} + \frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}\right), \tag{22}$$

for some constant c_1 such that $c_1 > 1 + 2\sqrt{\lambda_k^u}$. This condition is obviously true when $n \to \infty$. Then we have following theorem.

Theorem 2 *Consider pcLAD model, assume $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n$ are i.i.d. random variables satisfying Assumption 1 and 3, suppose $\lambda_k^l > \theta_{k,k}(\frac{1+\Phi}{\bar{c}})$ and (22) holds, then the pcLAD estimator $\hat{\beta}$ satisfies with probability at least $1 - 2p^{-4\min(k-m,m)(c_2^2-1)+1}$*

$$\left\|\hat{\beta} - \beta_0\right\|_2 \leq \sqrt{\frac{2\max(m,k-m)\log(p)}{n}} \frac{16\left\{\sqrt{2}c(1+\Phi) + c_1[\frac{5}{4} + \frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}]\right\}}{a\eta_k^l} \sqrt{1 + \frac{1}{\bar{c}} + \Phi},$$

where $c_1 = 1 + 2c_2\sqrt{\lambda_k^u}$ and $c_2 > 1$ is a constant, $\eta_k^l = \frac{[\lambda_k^l - \theta_k^k(\frac{1+\Phi}{\bar{c}})]^2}{\lambda_k^u(1+\Phi)}$, $\lambda = 2c\sqrt{n\log(p)}$.

From the theorem we can easily see that asymptotically, with high probability,

$$\left\|\hat{\beta} - \beta_0\right\|_2 = O\left(\sqrt{\max(k-m,m)\log(p)/n}\right). \tag{23}$$

By $k\log(p) \ll n$, we know the pcLAD estimator has near Oracle performance. Moreover, The bounds of pcLAD estimation error decay faster with the increase of n than (4), that is, equality constraints can improve the accuracy of estimation. This is also verified in section of simulation study in Sect. 5.

Unlike the L_2 bound of LADlasso in (4), which depends on $\sqrt{k}$, the L_2 bound of pcLAD depend on $\sqrt{m}$ and $\sqrt{k-m}$. The rate $\sqrt{(k-m)\log(p)/n}$ follows from the fact that (17) implies

$$\beta_{A_1} = (C_{A_1})^{-1}(b - C_{A_2}\beta_{A_2}). \tag{24}$$

Hence, the m coordinates of β_{A_1} are completely determined by the remaining $(p-m)$ coordinates. The problem of estimating k significant coefficients can be regarded as the problem of estimating $(k-m)$ significant coefficients. Note that the bound in Theorem 2 also depend on $\sqrt{m}$. In fact, this term reflects the error due to model selection. To see this, when $m = k$, it follows from (24) that we can exactly recover β_A, but only if we know the locations of the non-zero entries. There are $\binom{p}{m} \sim p^m$ possible locations of the non-zero entries. The number of possible locations of nonzero coefficients is a number related to m, so the bounds of estimation error are related to m.

Next, we will explain another reason that we need $k = O(1)$, there are p^k hypotheses for the determination of nonzero entries, and information theoretic arguments show that even if we have $m = k$ constraints, we still at least need n to be of order $\log\binom{p}{k} = k\log(p)$ to identify the correct hypothesis. In fact, the number of equality constraints m equals to the number of significance coefficients k is not satisfied in most cases. So we relax the requirement of $k\log(p) \ll n$, only require $k = O(1)$ to make sure correct hypothesis can be identified.

A simple consequence of Theorem 2 is that the pcLAD estimator will select most of the significant variables with high probability. We have the following theorem.

Theorem 3 *Suppose $\hat{T} = supp(\hat{\beta})$ be the estimated support of coefficients, in other words, $\hat{T}$ is the set of significant coefficient estimates. Then under the same conditions as in Theorem 2, with probability at least $1 - 2p^{-4\min(k-m,m)(c_2^2-1)+1}$*

$$\left\{ i : |\beta_i| \geq \sqrt{\frac{2\max(m, k-m)\log(p)}{n}} \frac{16\left\{\sqrt{2}c(1+\Phi) + c_1[\frac{5}{4} + \frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}]\right\}}{\alpha\eta_k^l} \right\} \in \hat{T}$$

where $c_1 = 1 + 2c_2\sqrt{\lambda_k^u}$ and $c_2 > 1$ is a constant, $\eta_k^l = \frac{[\lambda_k^l - \theta_k^k(\frac{1+\Phi}{\bar{c}})]^2}{\lambda_k^u(1+\Phi)}$, $\lambda = 2c\sqrt{n\log(p)}$.

This theorem shows that the pcLAD method will select a model that contains all the variables with large coefficients. If in model (15), all the nonzero coefficients are large enough in terms of absolute value, then the pcLAD method can select all of them into the model.

3.3 Discussion on inequality constraints

In the previous section we have concentrated on results for the equality constrained pcLAD. Next, we will briefly discuss the theoretical results of inequality constrained

pcLAD. When p is fixed dimension, results in the inequality setting are same to equality constraints'. It is easy to see that if β_0 lies inside the region, that is $C\beta_0 < b$, then the pcLAD and LADlasso should give same result because the constraints will play little role in the regression. However, if β_0 is on the constraint boundary, then the pcLAD should offer same improvements as equality LADlasso. This method of analyzing inequality constrained regression in fixed dimension has been used in many literatures, such as Liew (1976), Wang (1995), Wang (1996) and so on. Specifically, take nonnegative constraint $\beta \geq \mathbf{0}$ as an example. Under the assumptions and settings of this paper, the true significant coefficients and insignificant coefficients satisfy $\beta_{0A} > \mathbf{0}, \beta_{0B} = \mathbf{0}$. In the theoretical analysis of asymptotic properties, we only need to consider equality constraints $C_B = \mathbf{I}_{p-k}$. Thus the constrained matrix C is composed of 0 and C_B and the constrained vector b is $\mathbf{0}$, this constraint does not affect the proof of Lemma 1 and 2. In this paper, the result of Lemma 2 is the same as $\beta_B = 0$. So, $C_B = \mathbf{I}_{p-k}$ does not change the result of Theorem 1 and nonnegative constraint LAD also enjoys the Oracle property as unconstrained LADlasso.

Other examples that are highlighted in this paper are monotonic order LAD estimation, fused and general LADlasso. For monotonic order constraint in this paper, the insignificant coefficients $\beta_B = 0$ and the significant coefficients constrained matrix C_A is defined as (7). If there is no equality constraint in $C_A \leq b_A$, monotonic order constraint LADlasso will have the same result as unconstrained LADlasso. If there are some equality constraints, monotonic order constraint LADlasso will enjoy the asymptotic normality with equality constraints as (b) of Theorem 1. Nevertheless, for fused and general LADlasso, we can't assert this conclusion under the assumption of this paper. The reason is that in the proof of Lemma 1, $\tilde{X}$ will have different augmented forms under different dimensional settings. In fixed and high dimensions, $\tilde{X}$ may not satisfy the assumptions in this paper. How to make fused and general LADlasso also have Oracle property is a further research needing more technical assumptions and proof methods.

When p is high dimension, the L_2 bound of pcLAD in the inequality setting are more complicated. To our knowledge, there are two methods to analyze it. One is similar to the fixed dimension method, which has been used in He (2011). Following his ideas, for the inequality constraints $C_1\beta \leq b_1$, we partition C_1 and b_1 into block matrices as

$$C_1\beta_0 = \begin{pmatrix} C_{11} & C_{12} \\ C_{13} & C_{14} \end{pmatrix} \begin{pmatrix} \beta_{0A} \\ \beta_{0B} \end{pmatrix} \text{ and } b_1 = \begin{pmatrix} b_{11} \\ b_{12} \end{pmatrix},$$

such that $C_{11}\beta_{0A} = b_{11}$ and $C_{13}\beta_{0A} < b_{12}$. Note that β_0 satisfies the constraints $C_1\beta \leq b_1$ at the boundary (i.e., $C_{11}\beta_{0A} = b_{11}$) while satisfying the constraints $C_1\beta \leq b_1$ in the interior ($C_{13}\beta_{0A} < b_{12}$). Moreover, if the equality constraint $C_2\beta = b_2$ exists, partition it into block matrices as

$$C_2\beta = \begin{pmatrix} C_{21} & C_{22} \end{pmatrix} \begin{pmatrix} \beta_A \\ \beta_B \end{pmatrix} \text{ and } b_2 = b_2.$$

Then, we can reconstruct equality constraints

$$G = \begin{pmatrix} C_{11} \\ C_{21} \end{pmatrix} \text{ and } g = \begin{pmatrix} b_{11} \\ b_2 \end{pmatrix}.$$

Thus, $G\beta = g$ is new equality constraints which are brought into theoretical analysis. Here, we still take nonnegative constraint $\beta \geq \mathbf{0}$ as an example. The constraints of coefficients in high dimension is the same as in the fixed dimension. And the nonnegative constraints will be relax lasso like (16). Due to the $\beta_M = 0$ is not affect the proof process, the L_2 bound is $\sqrt{(k)\log(p)/n}$, which is the same as the result of Wang (2013). For monotonic order LAD estimation in high dimension, the L_2 bound is $\sqrt{\max(m, k-m)\log(p)/n}$, where m is the $\sum_{i \neq j} I(\beta_{0i} = \beta_{0j})$. The case of fused and general LADlasso in high dimension has been discussed briefly before, and we omit it here.

Another analysis method of inequality constrained regression is to add relaxed variables. As discussed by James et al. (2013), we can change inequality constraints into equality constraints by adding relaxed variables. However, when the added relaxed variable is close to 0 but not exactly 0, the assumption of coefficient sparsity may not be tenable. Although Negahban et al. (2010) have discussed lasso in this case, and they proved that another sparse vector can be used to approximate the not exactly sparse estimation, but extending it to pcLAD is far beyond the scope of our paper.

4 The implement of pcLAD

How to compute L1-penalized LAD regression is nontrivial task due to the nonsmoothness of LAD loss function and L1 penalty term. Fortunately, this nontrivial task has obtained a lot of attentions and many methods have been presented to solve penalized LAD regression such as the linear program using the interior point method (Koenker and Ng 2005), a solution path algorithm (Li and Zhu 2008), a greedy coordinate descent algorithm (Wu and Lange 2008; Peng and Wang 2015), pADMM and scdADMM (Gu et al. 2017), QPADMM (Yu and Lin 2017), QPADMM-slack (Fan et al. 2020) and so on.

However, linear constrained will makes all of the above methods can not be directly applied to pcLAD and even several methods fail completely since the optimal solution of pcLAD is limited to an affine set. In particular, a greedy coordinate descent algorithm for LADlasso can't work and all the ADMM algorithms mentioned above cannot be used directly. Recently, Inspired by Li and Zhu (2008) and Liu et al. (2020) proposed a solution path algorithm for solving generalized L1 penalized quantile regression with linear constraints. This algorithm utilizes the piecewise linear of the L1 penalized quantile regression solution path to get an entire solution path by solving a series of linear programming problems. It is worth noting that compared with the approach as in Li and Zhu (2008), it doesn't require that X has full column rank and allows more than one events occur at a transition point. These improvements make this algorithm possible to be used in high dimensional setting. If one want to get the entire solution

path, algorithm proposed by Liu et al. (2020) is a good choice. But this algorithm also has some limitations. One is it must calculate an entire solution for a sequence lambda in $(0, +\infty)$, then choose the best lambda by some criterions. For some optimization problems which can determine the specific value of λ, it is not cost-effective to use it. The other is that in solving the complete solution path, every λ in the sequence needs twice linear optimization with some constraints. Different from the path algorithm of classo (Gaines et al. 2018), it has no explicit solution only related to the active set. When p and n is large, it requires an expensive computational cost.

In the discussion of the Sect. 3, following Wang et al. (2007) and Wang (2013), we respectively determine the specific value λ of pcLAD in fixed and high dimension. Although, the algorithm proposed by Liu et al. (2020) can resolve the pcLAD, the computing is a large burden. In this section, we propose some efficient algorithms to solve pcLAD under specific λ.

4.1 Linear programing

A typical approach to solving LAD regression is to cast it as a linear program and then solve the linear program using the interior point method, so the first algorithm we present to solve pcLAD is a linear programing. Further, this method is also applied to penalized LAD in fixed and high dimensional regression such as Wang et al. (2007), Wu and Liu (2009), Gao and Huang (2010), Wang (2013) ,etc.. The popular **R** package quantreg is based on an interior point method which can solve the (penalized) LAD regression (Portnoy and Koenker 1997). LAD regression problem is equivalent to the linear program,

$$\begin{aligned} &\min_{\beta} \ 1_n^{\mathrm{T}} u + 1_n^{\mathrm{T}} v \\ &s.t. \quad \begin{cases} u - v + X\beta = y \\ u, v \in R_+^n, \beta \in R^p. \end{cases} \end{aligned} \tag{25}$$

Problem (25) is often solved with the interior method in its dual domain,

$$\begin{aligned} &\min_{d} \ -y^{\mathrm{T}} d \\ &s.t. \quad \begin{cases} X^{\mathrm{T}} d = 0 \\ d \in [-1/2, 1/2]^n. \end{cases} \end{aligned} \tag{26}$$

To apply this method to penalized LAD regression, just make a simple data augmentation for X and y. Then, penalized LAD regression can be computed with the follow dual domain,

$$\begin{aligned} &\min_{d} \ -\tilde{y}^{\mathrm{T}} d \\ &s.t. \quad \begin{cases} \tilde{X}^{\mathrm{T}} d = 0 \\ d \in [-1/2, 1/2]^{n+p}, \end{cases} \end{aligned} \tag{27}$$

where $\tilde{y} = (y^T, 0_p{}^T)^T$, $X = [X^T, \mathrm{diag}(\lambda)]^T$. Note that when $\lambda = 0_p$, (27) is ordinary LAD regression.

The main difficulty of solving pcLAD with linear programming algorithm is how to bring equality and inequality constraints into optimization. Inspired by Koenker

and Ng (2005), we can also following Berman (1973), consider the primal problem $\min_x\{c^{\mathrm{T}}x\,|Ax-b\in T, x\in S\}$,where the sets $\mathrm{T}=\{\mathrm{v}\in \mathrm{R}^n\}$ and $\mathrm{S}=\{\mathrm{v}\in \mathrm{R}^{2n}\times \mathrm{R}^p\}$ can be arbitrary closed convex cones. This canonical problem has the dual $\max_y\{b^T y\,\big|c-A^T y\in S^*, y\in T^*\}$, where $\mathrm{S}^*=\{v\in R^{2n}\times R^p\,\big|x^T y\geq 0\ if\ x\in S\}$ is the dual of S and $\mathrm{T}^*=\{\mathrm{v}\in \mathrm{R}^n\}$.

Thus, pcLAD is equivalent to the following linear program

$$\begin{aligned}&\min_{\beta}\ 1_n^{\mathrm{T}}u+1_n^{\mathrm{T}}v\\ &s.t.\quad\begin{cases}u-v+\tilde{X}\beta=\tilde{y}\\ C_1\beta=b_1\\ C_2\beta\leq b_2\\ u,v\in R_+^n,\beta\in R^p\end{cases}\end{aligned}\tag{28}$$

Then, for our purposes, it suffices to consider the following special case:

$$\begin{cases}c^T=(e_n^T,e_n^T,0_p^T)\\ x^T=(u^{\mathrm{T}},v^T,\beta^T)\\ T=\{v\in 0_{n+p+m_1}\times R_+^{m_2}\}\\ S=\{v\in R_+^{2n}\times R^p\}\end{cases}\text{ and }\begin{cases}\mathrm{b}^T=(\tilde{y}^T,{\mathrm{b}_1}^T,{b_2}^T)\\ y^T=(d_1^T,d_2^T,d_3^T)\\ T^*=\{v\in \mathrm{R}^{n+p}\times R^{m_1}\times R_-^{m_2}\}\\ S^*=\{v\in R_+^{2n}\times O_p\}\end{cases}\tag{29}$$

After some easy transformations, $z_1=\frac{d_1+e_n}{2}$, $z_2=d_2$, $z_3=-d_3$, $z=(z_1^T,z_2^T,z_3^T)^T$, the dual problem in (28) can be expressed more concisely as,

$$\begin{aligned}&\min_{z_1,z_2,z_3}\ -(\tilde{y}^T,b_1^T,b_2^T)z\\ &s.t.\quad\left[\tilde{X}^T,C_1^T,-C_2^T\right]\mathrm{z}=\tfrac{\tilde{X}^T}{2}e_{n+p},\\ &\qquad 0_{n+p}\leq z_1\leq e_{n+p},\\ &\qquad z_3\geq 0_{m_2}.\end{aligned}\tag{30}$$

It is noteworthy that we can also use two equality constraints $C_1\beta\geq b_1$ and $-C_1\beta\geq -b_1$ instead of the inequality constraint $C_1\beta=b_1$ to solve optimization problems (30),then the solution can be found using the **R** package quantreg, specifically, the estimator is implemented in *quantreg*'s functions **rq.fit.fnc** and **rq.fit.sfnc**.

In addition to transforming the augmented data into an optimized form of constrained LAD, pcLAD can also be equivalent to the following linear programming

$$\begin{aligned}&\min_{\beta}\ 1_n^{\mathrm{T}}u+1_n^{\mathrm{T}}v+n\lambda^T(\beta^+-\beta^-)\\ &s.t.\quad\begin{cases}u-v+X\beta=y\\ \beta=\beta^+-\beta^-\\ C_1\beta=b_1\\ C_2\beta\leq b_2\\ u,v,\beta^+,\beta^-\in R_+^n,\beta\in R^p\end{cases}\end{aligned}\tag{31}$$

Similar to the derivation of (30),the dual to (31) is

$$\begin{aligned} &\min_{z_1,z_2,z_3} -(y^T, 0_p^T, b_1^T, b_2^T)z \\ &s.t. \ \ \left[X^T, ndiag(\lambda), C_1^T, -C_2^T\right] \mathrm{z} = \tfrac{X^T}{2} e_p + \tfrac{n}{2}\lambda, \\ &\qquad 0_{n+p} \le \ z_1 \le e_{n+p}, \\ &\qquad z_3 \ge 0_{m_2}, \end{aligned} \tag{32}$$

where diag(λ) denotes the diagonal matrix with the components of λ on its diagonals. This result is consistent with the result in Gu et al. (2017). Furthermore, it's very easy to verify that (30) and (32) are equivalent. As Gu et al. (2017) points out, (32) involves p equality constraints and often is solved with the interior point method, but the interior point algorithm is the state-of-the-art method for fitting penalized LAD (0.5 quantile) regression in low to moderate dimensions. When the p is very large, the interior point method is less efficient. A lot of numerical evidence in Sect. 5.1 can demonstrate it. This phenomenon motivates us to consider another efficient alternative for fitting the high dimensional pcLAD regression.

4.2 Alternating direction algorithm

The ADMM is a general convex optimization algorithm first introduced by Gabay and Mercier (1976) and Glowinski and Marrocco (1975). It has become popular recently since its capability of solving high dimensional problems. In this subsection, we briefly review the ADMM and propose a nested scale form ADMM for pcLAD. A comprehensive overview of the ADMM can be found in Boyd et al. (2010).

In general ADMM is an algorithm to solve a problem that features a separable objective but connecting constraints.

$$\begin{aligned} &\min \ \ f(x) + g(z) \\ &s.t. \ \ Mx + Fz = c, \end{aligned} \tag{33}$$

where $f, g : R^p \mapsto R \cup \infty$ are closed proper convex functions. The ADMM solves problem (33) by writing it into the following equivalent form,

$$\begin{aligned} &\min \ \{f(x) + g(z)\} + \tfrac{\tau}{2} \|Mx + Fz - c\|_2^2 \\ &s.t. \ \ Mx + Fz = c, \end{aligned} \tag{34}$$

where the last term is called the augmentation, which is add for better convergence properties and the τ is a tunable augmentation parameter. Following standard convex optimization method, problem (45) has the following Lagrangian.

$$L_\tau(x, z, v) = f(x) + g(z) + v^T(Mx + Fz - c) + \frac{\tau}{2} \|Mx + Fz - c\|_2^2, \tag{35}$$

where v is the dual variable.

The basic idea of ADMM is to utilize block coordinate descent to the augmented Lagrangian function followed by an update of the dual variables v

$$\begin{aligned} x^{(t+1)} &\leftarrow \arg\min_{x} L_\tau(x, z^{(t)}, v^{(t)}); \\ z^{(t+1)} &\leftarrow \arg\min_{z} L_\tau(x^{(t+1)}, z^{(t)}, v^{(t)}); \\ v^{(t+1)} &\leftarrow v^{(t)} + \tau(Mx^{(t+1)} + Fz^{(t+1)} - c); \end{aligned} \tag{36}$$

where t is the iteration counter. Often it is more convenient to work with the equivalent scaled form of ADMM, which scales the dual variable and combines the linear and quadratic terms in the update step (36). The updates become

$$\begin{aligned} x^{(t+1)} &\leftarrow \arg\min_{x} f(x) + \tfrac{\tau}{2} \left\| Mx + Fz^{(t)} - c + u^{(t)} \right\|_2^2; \\ z^{(t+1)} &\leftarrow \arg\min_{z} g(z) + \tfrac{\tau}{2} \left\| Mx^{(t+1)} + Fz - c + u^{(t)} \right\|_2^2; \\ u^{(t+1)} &\leftarrow u^{(t)} + Mx^{(t+1)} + Fz^{(t+1)} - c; \end{aligned} \tag{37}$$

where $u = \frac{v}{\tau}$ is the scaled dual variable. As discussed in Gaines et al. (2018), the scaled form is especially useful in the case where $\mathrm{M} = -F = I$ and $c = 0$, as the updates can be rewritten as

$$\begin{aligned} x^{(t+1)} &\leftarrow prox_{\tau f}(z^{(t)} - u^{(t)}); \\ z^{(t+1)} &\leftarrow prox_{\tau g}(x^{(t+1)} + u^{(t)}); \\ u^{(t+1)} &\leftarrow u^{(t)} + x^{(t+1)} - z^{(t+1)}; \end{aligned} \tag{38}$$

where $prox_{\tau f}$ is the proximal mapping of a function f with parameter τ. Recall that the proximal mapping is defined as

$$prox_{\tau f}(v) = \arg\min_{x} \left(f(x) + \frac{\tau}{2} \|x - v\|_2^2 \right) \tag{39}$$

One benefit of using the scaled form for ADMM is that, in many situations, the proximal mappings have simple, closed form solutions, resulting in straightforward ADMM updates. And Gaines et al. (2018) has proved that the scaled form ADMM algorithm (hereinafter referred to as sADMM) is very effective in high dimensional constrained lasso estimation. Following this work, we will show that the scaled form ADMM can also be expanded to pcLAD.

Let $f(\beta) = \frac{1}{n} \sum_{i=1}^{n} \left| y_i - x_i^T \beta \right| + \sum_{j=1}^{p} \lambda_j \left| \beta_j \right|$ and $g(z) = \chi_{\mathbf{C}} = \begin{cases} +\infty & z \notin \mathbf{C} \\ 0 & z \in \mathbf{C} \end{cases}$, where set $\mathbf{C}$ is defined as $\{z \in R^p : C_1 z = b_1, C_2 z \le b_2\}$. For the first update of (38), $prox_{\tau f}$ in classo is regarded as a regular lasso problem, but in pcLAD needs a more technical method. Substitute $f(\beta)$ and $g(z)$ into $prox_{\tau f}$, we can get the following update

$$\beta^{(t+1)} = \arg\min_{\beta} \frac{1}{n}\sum_{i=1}^{n}\left|y_i - x_i^T\beta\right| + \sum_{j=1}^{p}\lambda_j\left|\beta_j\right| + \frac{\tau}{2}\left\|\beta + \mathrm{z}^{(t)} + u^{(t)}\right\|_2^2. \quad (40)$$

(40) is an unconstrained optimization problem of LAD loss function + penalty term, but the penalty is a lasso + a quadratic penalty. When $z^{(t)} + u^{(t)} = 0$, this combined penalty becomes an elastic net (Zou et al. 2005). Due to the combined penalty and nonsmoothness of the LAD loss function, as far as we know, there is no good method to resolve (40) directly when p is large scale. However, some recent literatures have proposed a number of algorithms to solve elastic net penalized quantile regression in high dimension such as Gu et al. (2017), Yu and Lin (2017). Inspired by these works, we can derive some algorithms to calculate (40).

Using the same steps as section 3.2 in Gu et al. (2017), (40) is equivalent to

$$\begin{aligned} &\min_{\beta,r}\ \frac{1}{n}\sum_{i=1}^{n}|r_i| + \sum_{j=1}^{p}\lambda_j\left|\beta_j\right| + \frac{\tau}{2}\left\|\beta + \mathrm{z}^{(t)} + u^{(t)}\right\|_2^2 \\ &s.t.\ \ X\beta + r = y \end{aligned} \quad (41)$$

Fix $\tilde{\tau} > 0$ and the augmented Lagrangian function of (41) is

$$\begin{aligned} L_{\tilde{\tau}}(\beta, \mathrm{r}, \theta) = &\frac{1}{n}\sum_{i=1}^{n}|r_i| + \sum_{j=1}^{p}\lambda_j\left|\beta_j\right| + \frac{\tau}{2}\left\|\beta + \mathrm{z}^{(t)} + u^{(t)}\right\|_2^2 - \theta^T(X\beta + r - y) \\ &+\frac{\tilde{\tau}}{2}\|X\beta + r - y\|_2^2. \end{aligned} \quad (42)$$

Denote $(\beta^{(t+1,k)}, r^{(k)}, \theta^{(k)})$ as the kth iteration of the algorithm for $k \geq 0$ and the next iteration is

$$\begin{aligned} \beta^{(t+1,k+1)} &\leftarrow \arg\min_{\beta} \sum_{j=1}^{p}\lambda_j\left|\beta_j\right| + \frac{\tau}{2}\left\|\beta + \mathrm{z}^{(t)} + u^{(t)}\right\|_2^2 - \beta^{\mathrm{T}}X^T\theta^{(k)} \\ &\quad + \frac{\tilde{\tau}}{2}\left\|X\beta + r^{(k)} - y\right\|_2^2 \\ r^{(k+1)} &\leftarrow \arg\min_{r} \frac{1}{n}\sum_{i=1}^{n}|r_i| - r^T\theta^{(k)} + \frac{\tilde{\tau}}{2}\left\|X\beta^{(t+1,k+1)} + r - y\right\|_2^2 \\ \theta^{(k+1)} &\leftarrow \theta^{(k)} - \tilde{\tau}(X\beta^{(t+1,k+1)} + r^{(k+1)} - y) \end{aligned} \quad (43)$$

It is noteworthy that although the β update of (43) is not the same as (40) in Gu et al. (2017), one can expand the quadratic penalty into a ridge penalty $\|\beta\|_2^2$ and a linear summation penalty $\beta^{\mathrm{T}}(\mathrm{z}^{(t)} + u^{(t)})$. Then,

$$\beta^{(t+1,k+1)} \leftarrow \arg\min_{\beta} \left(\sum_{j=1}^{p} \lambda_j |\beta_j| + \frac{\tau}{2} \|\beta\|_2^2 \right) - \beta^{\mathrm{T}}[X^T\theta^{(k)} - \tilde{\tau}(\mathrm{z}^{(t)} + u^{(t)})] + \frac{\tilde{\tau}}{2} \left\| X\beta + r^{(k)} - y \right\|_2^2 \tag{44}$$

It is the same as the update of elastic net penalized quantile regression.

Like algorithm 3 of Gu et al. (2017), when we use pADMM to solve (43), the update steps have the following closed formula.

$$\begin{aligned} \beta^{(t+1,k+1)} &\leftarrow ((\tilde{\tau}\eta + \tau)^{-1} shirnk[\tilde{\tau}\eta\beta^{(t+1k)} + X_j^T(\theta^{(k)} + \tilde{\tau}y - \tilde{\tau}X\beta^{(t+1,k)} - \tilde{\tau}r^{(k)}) \\ &\quad - \tau(z^{(t)} + u^{(t)}), \lambda_j])_{1\le j\le p}; \\ r^{(k+1)} &\leftarrow prox_{\tilde{\tau}\|\|_1}(y - x_i^T\beta^{(t+1,k+1)} + \tilde{\tau}^{-1}\theta^{(k)}); \\ \theta^{(k+1)} &\leftarrow \theta^{(k)} - \tilde{\tau}\gamma(X\beta^{(t+1,k+1)} + r^{(k+1)} - y); \end{aligned} \tag{45}$$

where $\eta \ge \Lambda_{\max}(X^TX)$, γ is a constant which is controlling the step length for θ. $\Lambda_{\max}(X^TX)$ denotes the largest eigenvalue of a real symmetric matrix and shrink$[x, y] = \mathrm{sgn}(x)\max(|x| - y, 0)$ denotes the soft shrinkage operator with sgn being the sign function. These definitions are the same as those in Gu et al. (2017). Consider $prox_{\tilde{\tau}\|\|_1}(v) = \arg\min_r(\frac{1}{n}\sum_{i=1}^n |r_i| + \frac{\tilde{\tau}}{2}\|r - v\|_2^2)$ and as Lemma 1 proved in Gu et al. (2017), it has an explicit solution.

$$prox_{\tilde{\tau}\|\|_1}(v) = v - \max\left(-\frac{1}{2\tilde{\tau}}, \min\left(v, \frac{1}{2\tilde{\tau}}\right)\right) \tag{46}$$

The main difference between (45) and algorithm is that the β update has one more constant offset term $\tau(z^{(t)} + u^{(t)})$ since the linear summation penalty $\beta^{\mathrm{T}}(\mathrm{z}^{(t)} + u^{(t)})$. Furthermore, we can also use scdADMM (algorithm 4 in Gu et al. (2017)) to solve (43) by adding an same constant offset term to the corresponding position of β update.

For the second update of (38), we use the same method as Gaines et al. (2018). $prox_{\tau g}$ is a projection onto the affine space $\mathbf{C}$. This projection onto convex sets is well-studied. In many applications, the projection can be solved analytically (see Section 15.2 of Lange (2013) for several examples). For situations where an explicit projection operator is not available, the projection can be found by using quadratic programming to solve the dual problem, which always has a smaller number of variables.

To sum up the above discussion, the nested scale form ADMM is described in Algorithm 1. Although Algorithm 1 contains a nested ADMM iteration, both outer and inner iteration have explicit expressions which makes it calculate very fast in high dimensional setting. In fact, if lasso problem of the β update of sADMM is solved by ADMM, sADMM also has a nested ADMM iteration. For numerical evidence, see Sect. 5.1.

Algorithm 1 Nested ADMM for solving the pcLAD

1.Initialize the algorithm with $\beta^{(0)} = z^{(0)} = \beta^0, u^{(0)} = 0, \tau > 0$

2.For $t = 0, 1, 2, \ldots,$ repeat steps 2.1-2.4 until the convergence criterion is met.

2.1.Initialize $\beta^{(t,0)} = \beta^{(t)}, r^{(0)} = y - X\beta^{(t)}, \theta^{(0)} = 0, \tilde{\tau} > 0$

2.1.1.For $k = 0, 1, 2, \ldots,$ repeat steps 2.1.2-2.1.4 until the convergence criterion is met.

2.1.2.update $\beta^{(tk+1)} \leftarrow ((\tilde{\tau}\eta + \tau)^{-1}\text{shrink}[\tilde{\tau}\eta\beta_j^{(t,k)} + X_j^T(\theta^{(k)} + \tilde{\tau}y - \tilde{\tau}X\beta^{(t,k)} - \tilde{\tau}r^{(k)}) - \tau(z^{(t)} + u^{(t)}), \lambda_j])_{1\le j\le p}$

2.1.3.update $r^{(k+1)} \leftarrow prox_{\tilde{\tau}\|\|_1}(y - x_i^T\beta^{(t,k+1)} + \tilde{\tau}^{-1}\theta^{(k)})$

2.1.4.update $\theta^{(k+1)} \leftarrow \theta^{(k)} - \tilde{\tau}\gamma(X\beta^{(t,k+1)} + r^{(k+1)} - y)$

2.2.update $\beta^{(t+1)} = \beta^{(t,k+1)}$

2.3.update $z^{(t+1)} \leftarrow proj_{\mathbf{C}}(\beta^{(t+1)} + u^{(t)})$

2.4.update $u^{(t+1)} \leftarrow u^{(t)} + \beta^{(t+1)} - z^{(t+1)}$

5 Simulation

In this section, we will show some numerical results when p is fixed and larger than n. All simulations were performed on the Inter E5-2650 2.0 GHz processor with 16 GB memory. In fixed dimension, we use the $\lambda_j = \frac{5\log(p)}{n|\tilde{\beta}_j|}$, where $\tilde{\beta}_j$ is the jth element in the ordinary LAD estimation vector. As discussed in Wang et al. (2007), these λ_j satisfy $\sqrt{n}a_n \to 0$ and $\sqrt{n}b_n \to \infty$. When p is high dimensional, we adopt $\lambda = \sqrt{1.1\log p/n}$, although it is smaller than $\lambda = \sqrt{2\log p/n}$ which is used most in Wang (2013), it will not affect the result of Theorem 2.

In the simulation, the model for the simulated data is $y_i = x_i'\beta + \varepsilon_i, i = 1, 2, \ldots, n$. Each experiment uses three different error terms ε, that is $N(0, 1), t(2)$, Cauchy$(0, 1)$. When the error term obeys the normal distribution, the λ of classo is selected according to James et al. (2013). When the error term does not obey the normal distribution, we use the 10 fold CV method to select λ. Moreover, each row of the design matrix X is generated by $N(0, \Omega)$ distribution with Toeplitz correlation matrix $\Omega_{ij} = 0.5^{|i-j|}$ and normalized such that each column has L_2 norm $\sqrt{n}$.

5.1 Comparison of algorithms

In this subsection, we introduce some implementation details of the several algorithms and compare theirs the time-consuming. In all numerical experiments, we will use four **R** packages, *quantreg*, *osqp*, *glmnet*, *FHDQP*. The first three packages can be found on **R** official website, https://www.r-project.org/, and the link of *FHDQP* package is https://users.stat.umn.edu/zouxx019/ftpdir/code/fhdqr/. LP and QP for constrained regression are implemented by *quantreg* and *osqp* respectively. More details about QP can be found in Gaines et al. (2018). In fixed dimensional constrained regression, the initial values of ADMM are unconstrained penalized estimates calculated by *quantreg*, and under the setting of high dimension, the corresponding initial value is calculated by *FHDQP*. For the first update of (36) in classo, we use *glmnet* package. Other iterative steps of sADMM and nADMM does not need to use **R** package since they have explicit solutions.

As noted in Algorithm 1, nADMM includes three additional tuning parameters, $\tau, \tilde{\tau}, \gamma$. We adopt $\tau = \frac{1}{n}$ the (suggested by Gaines et al. (2018)), $\tilde{\tau} = 0.05, \gamma =$

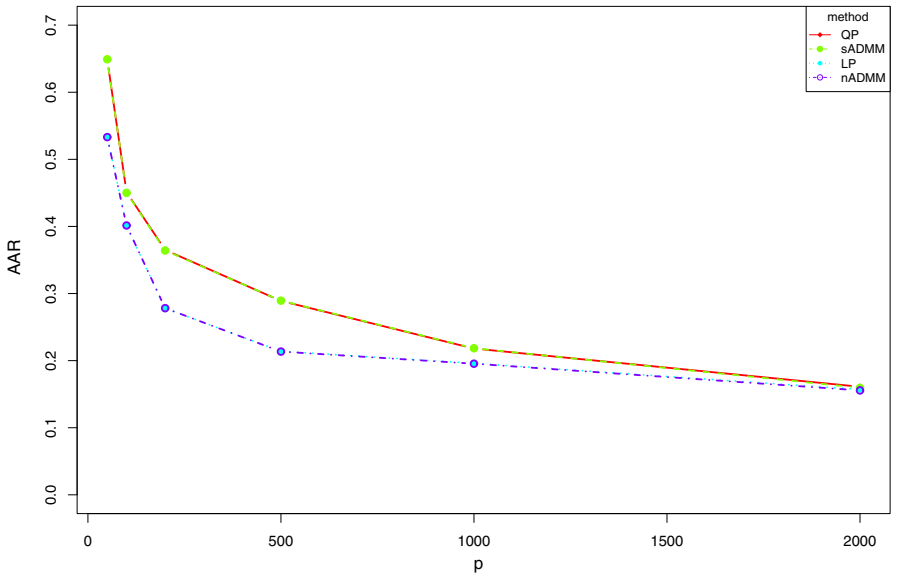

Fig. 1 Object function values computed by several algorithms

1(default value in $FHDQP$) in all numerical experiments. All ADMM algorithms are iterated until some stopping criterion is met. We adopt the stopping criterion from Boyd et al. (2010). Specifically, the outer iteration of nADMM is terminated either when sequence $\{(\beta^{(t)}, z^{(t)}, u^{(t)})\}$ meets the following criterion:

$$\begin{aligned} &\left\| X\beta^{(t)} + z^{(k)} - y \right\|_2 \leq \sqrt{n}\varepsilon_1 + \varepsilon_2 \max\{\left\| X\beta^{(t)} \right\|_2, \left\| z^{(t)} \right\|_2, \|y\|_2\}; \\ &\tau \left\| X\beta^{(t)} + z^{(k)} - y \right\|_2 \leq \sqrt{p}\varepsilon_1 + \varepsilon_2 \left\| X^T u^{(t)} \right\|_2; \end{aligned} \tag{47}$$

where typical choices are $\varepsilon_1 = 10^{-3}$ and $\varepsilon_2 = 10^{-3}$, or when the number of nADMM iterations exceeds a certain number, say 10^5. The conditions for termination of inner iteration is the same as outer iteration's. If one wants to get faster convergence rate, the termination condition of inner iteration can be more relaxed, such as $\varepsilon_1 = \varepsilon_2 = 10^{-2}$. In order to verify the efficiency of LP and nADMM algorithm in estimating pcLAD under different dimensions. Specifically, we set n is fixed at 100, but $p = (50, 200, 1000, 2000)$. The true coefficient vector is $\beta_0 = (-1, -2, -3, 1, 2, 3, 0_{p-6}^T)^T$ and $\varepsilon_i \sim N(0, 1)$. To make this experiment representative, we use mixed constraint set $\mathbf{C} = \{1_n^T \beta = 0, \beta_1, \beta_2, \beta_3 \leq 0\}$. We use QP (quadratic programming) and sADMM to fit classo regression, LP and nADMM to fit pcLAD regression. All simulations used 100 replicates and record the running time in the Table 1.

For fixed dimension, LP outperform nADMM in time consuming, while with the increase of p, the performance of LP is worse and worse. In all settings, although we have used a very efficient ADMM algorithm proposed by Stellato et al. (2018), QP needs longer computation time than LP since the optimization form of QP is

Table 1 Timings (in seconds) for running pcLAD and classo regression with specific λ

Method	p = 50	p = 200	p = 1000	p = 2000
LP	1.04	7.74	519	3368
nADMM	2.04	5.92	68	261
QP	1.27	8.32	736	4557
sADMM	1.33	3.05	38	120

Table 2 Sum to zero constraint in fixed p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	3.580 (5.534)	3.676 (5.637)	4.097 (13.16)
	Prediction	2.779 (3.403)	2.986 (4.480)	3.190 (4.949)
LADlasso	Estimation	4.747 (9.919)	4.894 (8.934)	5.197 (14.64)
	Prediction	4.406 (11.03)	4.535 (7.626)	5.015 (10.60)
classo	Estimation	3.399 (4.144)	5.158 (66.46)	$2747(10^9)$
	Prediction	2.681 (3.280)	4.787 (22.38)	$1321(3 \times 10^8)$

more complex. On the contrary, nADMM takes more time than sADMM, due to the nonsmoothness of LAD loss function. To be specific, the main difference between nADMM and sADMM is the iteration of the β step, the former is a variant of LADlasso, and the latter is a variant of lasso. One can also verify the different computation time by using *glmnet* (lasso) and $FHDQR$(LADlasso) packages to fit a same set of high dimensional regression data. Note also that to do a meaningful timing comparison, we need to check the objective function values of pcLAD and classo at the optimal solution computed by the different algorithm. To make sure different algorithms yield the same objective function values, it is sufficient to compare the optimal objective function value in (2) even though it's unfair to classo. The results are illustrated in Fig. 1. From Fig. 1, we know the objective functions of LP and nADMM are almost the same, QP and sADMM are the almost same too.

5.2 Sum to zero constraints

The first simulation involves a sum-to-zero constraint on the true parameter vector, $\sum_j \beta_j = 0$. Recently, this type of constraint on the lasso has seen increased interest as it has been used in the analysis of compositional data as well as analyses involving many biological measurement analyzed relative to a reference point (Lin et al. 2014; Shi et al. 2016; Altenbuchinger et al. 2017). Written in the pcLAD formulation (2), this corresponds to $C_1 = 1'_p$ and $b_1 = 0$. For this simulation, in order to distinguish the bounds of estimation error $||\hat{\beta} - \beta_0||_2^2$ and prediction error $||X\hat{\beta} - X\beta_0||_2^2/n$ in different cases, the true parameter vector β_0 , was defined as $\beta_0 = (10, 10, 10, -10, 10, -10, 0, \ldots, 0)$. The true parameter satisfies the sum to zero constraint, then the constraints can be imposed on the estimations. The main results of the simulation are given in the Tables 2 and 3.

Table 3 Sum to zero constraint in high dimensional p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	6.657 (27.57)	9.128 (55.45)	9.774 (107.2)
	Prediction	4.411 (9.605)	5.794 (16.59)	6.736 (70.80)
LADlasso	Estimation	6.548 (25.25)	8.997 (45.38)	9.755 (99.15)
	Prediction	4.958 (11.61)	6.457 (18.08)	7.442 (75.62)
Classo	Estimation	3.314 (2.375)	10.58 (300.3)	$10^5(10^{11})$
	Prediction	3.128 (3.757)	7.648 (469.0)	$9171(10^{10})$

In Tables 2 and 3, we set $n = 200$ and $p = 50, 400$, respectively. Each data in the table is the mean value after 100 repetitions and the numbers in brackets is its variance. We can see classo has the best performance when the ε obeys the normal distribution. When the ε obeys the $t(2)$, which does not have bounded variance, the classo will not perform as well as LADlasso and pcLAD. When the ε obeys the Cauchy(0, 1), classo will no longer make sense because the errors in estimation and prediction are intolerable. It is necessary to note that when the data in the table exceeds 10^5, in order to facilitate recording, we will only take the highest order. For example, 1234567 will be recorded as 10^7. By the way, when ε obeys Cauchy distribution, the median of prediction error and estimation error is not as large as the mean value, it is about between 10 and 20, which reflects that the classo estimation fluctuates greatly under Cauchy distribution.

In all dimensional settings, the unconstrained LADlasso can work normally under three kinds of error terms. Due to the existence of prior information, its estimation and prediction effects should be worse than that of equality constrained LADlasso. In Table 2, the obvious conclusion is true, However, the surprising results are appeared in Table 3, the estimation error of pcLAD is not as good as LADlasso in the three cases of ε, but the prediction error is better than it.

At first, it puzzled us, but the results of many experiments are still the same. Therefore, we notice that this equality constraint is for all coefficients, while in the setting of the high dimensional model in Sect. 3.2, the equality constraints are only imposed on the significance coefficients. In order to verify the conclusion of Sect. 3.2, we check the selection of non-zero coefficients of pcLAD, and the results confirm our idea. There are many zero coefficients being mistakenly selected as non-zero coefficients. At the same time, we have done another group of experiments. All the settings of this group of experiments are the same as the previous high-dimensional experiments. The only difference is that we only restrict the significance coefficients. The constraint matrix is as follows:

$$\begin{pmatrix} 1 & 0 & 0 & -1 & 0 & \cdots & & \\ 0 & 1 & 0 & 0 & -1 & 0 & \cdots & \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & \cdots \end{pmatrix},$$

and $b_1 = (0, 0, 0)'$.

Table 4 Three sum to zero constraints in high dimensional p

Method	Error	Three equality constraints	One equality constraint
pcLAD	Estimation	2.601 (6.080)	8.661 (54.94)
	Prediction	3.122 (13.39)	6.058 (32.00)
	Non-zero number	7.3 (2.677)	13.3 (47.78)
LADlasso	Estimation	8.471 (47.27)	
	Prediction	6.396 (34.43)	
	Non-zero number	6.400 (0.933)	

The main results of this experiment are shown in Table 4. From Table 4, we can see that LADlasso's result naturally does not change much because it has no constraints. However, the results of pcLAD are greatly improved. This result shows when p is high dimension, constraints should be placed on the significance coefficients, instead of every coefficients. Otherwise it will lead to excessive selection of non-zero coefficients. One may ask why this constraint is not set on the sparse model with fixed dimensions in this paper. Because in the fixed dimension pcLAD, we choose the adaptive L_1 penalty, which will impose a very large penalty on the insignificant coefficient. This forces the true zero coefficients to be estimated as 0.

5.3 Non negativity constraints

In this simulation, we choose the fixed dimension as $n = 100$, $p = 50$, the high dimension is $n = 100$, $p = 200$. This choice of n and p is different from the previous experiments and to verify whether the performance of the model is consistent under different n and p. In this experiment, we added type 1 error and type 2 error. The average type 1 error means the average number of significant variables that are unselected over 100 runs. The average type 2 error means the average number of insignificant variables that are selected over 100 runs. Because of the nonnegative constraint, the true coefficient we choose is $(1, 2, 3, 4, 5, 6, 0, \ldots)$.

The main results are shown in Tables 5 and 6. Just like the conclusion in Sect. 5.2, classo does better job in estimation and prediction errors under normal error. However, it also has a disadvantage, that is the type 2 error is relatively large. The reason for this is that the λ chosen by CV tends to choose more variables. For more details about this, one can refer to Leng et al. (2004) and Wang (2013). The estimation and prediction error of classo is worse than that of pcLAD in $t(2)$, but it can still be acceptable. However, under the Cauchy$(0, 1)$ distribution, the classo results are unreliable.

The above results show that inequality pcLAD has better estimation and prediction effect than inequality classo in non normal data. Moreover, in terms of variable selection, pcLAD is better than classo in any case.

5.4 Complex constraints

In the above two simulations, we have considered the case of equality constraint and inequality constraint respectively. In this subsection, we consider the case where both

Table 5 Non-negativity constraints in fixed p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	1.604 (2.812)	2.173 (6.969)	2.897 (4.758)
	Prediction	1.317 (1.140)	1.611 (2.396)	1.970 (2.001)
	Type1	0.18 (0.149)	0.25 (0.270)	0.42 (0.246)
	Type2	0.01 (0.01)	0.07 (0.065)	0.45 (0.734)
Classo	Estimation	0.956 (1.419)	3.402 (83.30)	$264.5(2 \times 10^7)$
	Prediction	0.960 (2.059)	3.863 (295.2)	$292(3 \times 10^7)$
	Type1	0.04 (0.038)	0.32 (0.219)	0.72 (0.931)
	Type2	2.7 (29.28)	8 (76.92)	13.34 (108.3)

Table 6 Non-negativity constraints in high dimensional p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	1.638 (3.024)	2.122 (3.693)	3.125 (10.18)
	Prediction	2.010 (4.841)	2.546 (6.098)	3.770 (21.20)
	Type1	0.09 (0.082)	0.12 (0.106)	0.16 (0.155)
	Type2	0.45 (0.674)	0.56 (0.531)	0.91 (1.032)
Classo	Estimation	0.602 (0.213)	3.145 (2.795)	$1421(4 \times 10^8)$
	Prediction	0.649 (0.317)	4.470 (5.821)	$805.7(10^8)$
	Type1	0 (0)	0.2 (0.161)	0.92 (1.145)
	Type2	2.53 (3.625)	18.22 (97.52)	36.18 (2629)

equality and inequality constraints are exist simultaneously. And two models will be considered.

The first model we consider is complex constrained ordinary LADlasso with $n = 200$, $p = 50, 400$. In fact, this complex constrained lasso has already appeared in Hu et al. (2015b). In order to compare the estimation error better, the coefficient of Hu et al. (2015b) is increased by 10 times. The true parameter vector is defined as $\beta_0 = (10, 5, -10, 0, \ldots, 0, 10, 5, -10, 0, \ldots, 0)'$, so only its 1st, 2nd, 3rd, 11th, 12th, and 13th elements are nonzero. The constrained pcLAD is estimated subject to the constraints:

$$\beta_1 + \beta_2 + \beta_3 \geq 0, \ \beta_1 + \beta_3 + \beta_{11} + \beta_{13} = 0,$$
$$\beta_2 + \beta_5 + \beta_{11} \geq 10, \ \beta_2 + \beta_8 + \beta_{12} = 10.$$

The main results are shown in Tables 7 and 8, the result of data presentation is the same as that in Sect. 5.3. We omit the data analysis in this section.

The second model we consider is the complex constrained LAD fused lasso and complex constrained fused lasso with $n = 200$, $p = 100, 1000$. Indeed, this complex constrained quantile lasso has already appeared in Liu et al. (2020). We adopt true parameter vector used in Liu et al. (2020), that is $\beta_0 = (-1, -1, 1, 1, 0, \ldots, 0)'$. The linear constraints are:

Table 7 Complex constraints in fixed dimensional p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	2.724 (3.310)	2.857 (4.151)	3.129 (4.105)
	Prediction	2.407 (2.818)	2.638 (4.780)	2.868 (3.803)
	Type1	0 (0)	0 (0)	0 (0)
	Type2	0 (0)	0.03 (0.029)	0.19 (0.216)
Classo	Estimation	2.699 (4.711)	4.800 (23.18)	$10^5(10^{11})$
	Prediction	2.376 (3.050)	3.348 (8.657)	$7489(3 \times 10^{10})$
	Type1	0 (0)	0 (0)	0 (0)
	Type2	19.10 (207.7)	20.19 (211.3)	25.36 (169.7)

Table 8 Complex constraints in high dimensional p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	2.997 (4.335)	3.970 (8.457)	5.083 (10.65)
	Prediction	2.646 (3.065)	3.225 (4.363)	4.070 (5.359)
	Type1	0 (0)	0 (0)	0 (0)
	Type2	2.18 (3.293)	2.44 (3.802)	2.82 (2.844)
Classo	Estimation	2.511 (1.984)	6.849 (62.32)	$10^6(10^{12})$
	Prediction	2.410 (2.125)	5.517 (41.01)	$10^6(10^{12})$
	Type1	0 (0)	0 (0)	0.2 (0.326)
	Type2	17.36 (174.6)	22.95 (139.94)	38.26 (8793)

Table 9 Complex constrained in fixed dimensional p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	1.117 (2.078)	1.206 (2.643)	1.415 (2.910)
	Prediction	0.987 (1.859)	1.132 (2.347)	1.347 (2.578)
	Type1	0 (0)	0 (0)	0 (0)
	Type2	0 (0)	0.020 (0.046)	0.187 (0.072)
Classo	Estimation	1.084 (1.971)	3.792 (19.51)	$5841(2 \times 10^8)$
	Prediction	0.973 (1.772)	3.542 (15.30)	$4920(1 \times 10^8)$
	Type1	0 (0)	0 (0)	0 (0)
	Type2	22.12 (115.7)	24.16 (157.7)	28.32 (191.4)

$$\beta_1 - 2\beta_2 - \beta_3 + 2\beta_4 \geq 1, \quad 3\beta_1 - 2\beta_2 + \beta_3 + \beta_4 \geq 0,$$
$$\beta_1 - \beta_2 + 2\beta_3 + 5\beta_4 = 7, \quad -3\beta_1 + \beta_2 - 6\beta_3 - \beta_4 = -5.$$

The main result of complex constrained LAD fused lasso in this synthetic data are shown in Tables 9 and 10. Note that our theoretical analysis is not suitable for fused LADlasso, so we utilize CV to select all penalty parameter λ.

Table 10 Complex constrained in high dimensional p

Method	Error	N(0,1)	t(2)	Cauchy
pcLAD	Estimation	1.216 (2.257)	1.377 (2.908)	1.507 (3.134)
	Prediction	1.107 (1.982)	1.224 (2.681)	1.431 (2.769)
	Type1	0 (0)	0 (0)	0 (0)
	Type2	4.45 (12.89)	5.09 (19.67)	7.15 (27.42)
Classo	Estimation	1.007 (1.528)	5.019 (37.49)	$4 \times 10^5 (10^{10})$
	Prediction	0.959 (1.597)	4.782 (31.67)	$2 \times 10^5 (10^{10})$
	Type1	0 (0)	0 (0)	0 (0)
	Type2	37.45 (254.7)	42.38 (312.5)	55.29 (502.3)

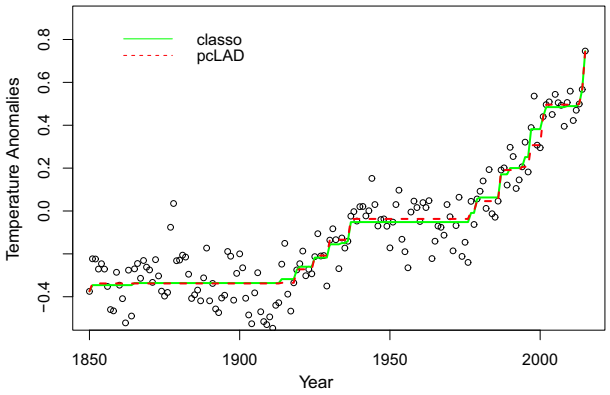

Fig. 2 Global warming data

In the Sect. 3.3, we have clarified that LAD fused lasso and constrained LAD fused lasso may not have the Oracle theoretical properties under the assumption in this paper. However, from Tables 9 and 10, constrained LAD fused lasso has good estimation and prediction performances. This numerical result also shows that theoretically constrained LAD fused lasso may also have Oracle property with new technical assumptions and methods.

An interesting phenomenon is that the estimation error and prediction error, are of the same order in all cases. We have not proved it in theory, but we believe that the theoretical results should be close to the numerical results.

6 Real data applications

In this section, we apply pcLAD to three different real data, and compare with classo.

6.1 Global warming data

For our first application of the pcLAD on a real data set, we revisit the global temperature data provided by Jones et al. (2016). The data set contains of annual temperature anomalies from 1850 to 2015. As mentioned, there appears to be a monotone trend to the data over time, so it is natural to want to incorporate this information when the trend is estimated.

Wu et al. (2001) and Gaines et al. (2018) achieved this by using isotonic regression. The LAD version of isotonic regression which has been described in Sect. 2.2, so we will not repeat it here. Because we want to get the temperature fitting data of each year, the penalty term is unnecessary, so $\lambda = 0$. In this experiment, the sample size n and dimension p are the same, and the value is not large, thus we use LP and QP to solve pcLAD and classo respectively. Significantly, the design matrix X is p-dimension identity matrix, so the optimal solution is the fitting value of y. Then we show the fitting effect of pcLAD and classo in Fig. 2. To be honest, we can't obviously see that the method fits better, so we calculate the $\|y - \hat{y}\|_1$ of classo and pcLAD. The values are 12.31 and 12.14 respectively, and show that the fitting of pcLAD is closer to the real value in this rule.

6.2 Brain tumor data

Our second application of the pcLAD uses a version of the comparative genomic hybridization (CGH) data from Bredel et al. (2005) which was modified and studied by Tibshirani and Wang (2008) and Gaines et al. (2018). This version of the dataset is available in the *cghFLasso* **R** package. The dataset includes CGH measurements from 2 glioblastoma multiforme (GBM) brain tumors. CGH array experiments are often used to estimate each gene's DNA copy number by obtaining the $\log_2$ ratio of the number of DNA copies of the gene in the tumor cells relative to the number of DNA copies in the reference cells. Mutations to cancerous cells result in amplifications or deletions of a gene from the chromosome, so the purpose of the analysis is to identify these gains or losses in the DNA copies of that gene (Michels et al. 2007). For a more detailed description of this data, one can see Bredel et al. (2005) and Michels et al. (2007). The form of pcLAD applied to this data set is as follows:

$$\underset{\beta}{\text{minimize}} \|y - \beta\|_1 + n\lambda_1 \|\beta\|_1 + n\lambda_2 \sum_{j=2}^{p} |\beta_j - \beta_{j-1}|. \tag{48}$$

From (48), we know the optimal solution is a sparse sequence fitting y, which is the log 2 ratio mentioned above. In this experiment, the sample size n and dimension p are the same, and the value is large. Thus, we use sADAMM and nADMM to solve pcLAD and classo respectively. For the penalty parameters λ_1 and λ_2, we use 10-fold CV to select them because our theoretical analysis is not suitable for fused LADlasso. Moreover, (48) can also be solved by *genlasso* **R** package. We don't show the results of *genlasso* package, because Gaines et al. (2018) has been used it in this real data applications, and proved that the experimental results of sADMM and *genlasso* are

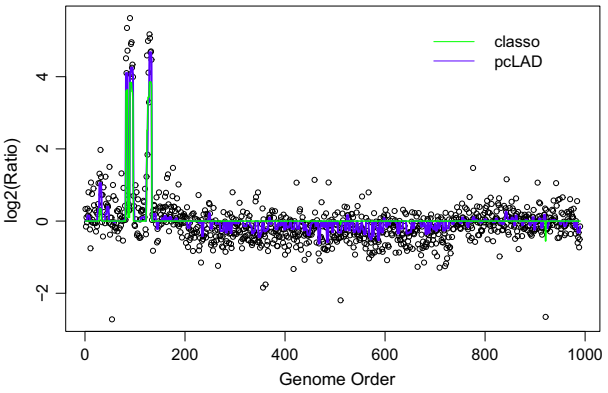

Fig. 3 Brain tumor data

the same. We compare the fitting results of this form of pcLAD and classo (sparse fused lasso Tibshirani et al. (2005)) in Fig. 2. From the Fig. 2, we can see that pcLAD fitting is better, especially for some large data. In numerical terms, the $\|y - \hat{y}\|_1$ of sparse fused lasso and pcLAD is 357.1 and 244.7 respectively. The difference between the two fitting values shows that pcLAD is better than sparse fused lasso to fit this dataset.

6.3 Stock index data

The last application of the real data is the Shanghai Stock Exchange 50 stock index (SSE 50) and Shanghai Shenzhen 300 stock index (CSI 300). SSE 50 Index is composed of 50 representative stocks with large scale and good liquidity in Shanghai securities market. CSI 300 Index is made up of 300 A-shares from Shanghai and Shenzhen stock markets.

Firstly, we give a brief introduction about stock index and index tracking. Stock index is a method for fitting and predicting the trend of the stock market by choosing some representative stocks. For example, SSE 50 contains 50 component stocks. There are many other famous stock index named by their Exchanges such as S&P 500, FTSE 100 and so on. Index tracking is a hot issue, the main idea is to select a few representative stocks to predict the whole stock index. The contribution of each component stock to the stock index must be positive, so we need to add nonnegative constraints on the regression coefficients. Because stock index tracking requires both sparse and nonnegative constraints, many nonnegative constrained penalty estimates have been proposed recently, include Wu et al. (2014), Yang and Wu (2016), Wu and Yang (2014), Li et al. (2019), etc.. Following the above work, in this section, the form of pcLAD applied to the stock index traching is the nonnegative LADlasso (nLADlasso) mentioned in Sect. 2.4, which is the LAD version of nonnegative lasso (nlasso) Wu et al. (2014). It is worth noting that the contribution of all component stocks to the index is positive, that is, the true model does not have sparseness and every true coefficient is positive. Then the assumption of sparsity is not tenable, and the selection

criterion of λ is meaningless. Following Wu et al. (2014), we can choose some penalty parameters from 0 to a sufficiently large positive number which shrinks all coefficients to 0, and the interval between the two parameters is equal.

Because the latest two component stock adjustments of CSI 300 Index and SSE 50 Index are on December 16, 2019 and June 15, 2020, we selected SSE 50 Index and CSI 300 Index data from January 2 to June 12, 2020. It is necessary to note that some component stocks of SSE50 and CSI 300 have been closed for a short period of time. We use the average stock price of the constituent stock during the non-closing period to fill the stock price at these closing times. The data is divided into time windows: the first 80 days' data used for modeling and the next 20 days' data used for forecasting. In the process of tracking SSE 50 Index, the sample size n is 80 and the dimension of variable p is 50, which is a fixed dimension problem. But in CSI 300 Index, $n = 80$, $p = 300$, which is a high dimensional problem. Therefore, we use fixed pcLAD and classo in SSE 50 tracking and high dimensional pcLAD and classo in CSI 300. For these data, $\lambda = 100$ is enough large to shrink all coefficients to 0, so we choose 1000 penalty parameters from 0 to 100 and the interval between the two parameters is 0.1.

Let $x_{t,j}$ and y_t represent the returns of the jth constituent stock and the index respectively, $j = 1, 2, \ldots, 50(300)$. Then we can describe the relationship between $x_{t,j}$ and y_t by a linear regression model:

$$y_t = \sum_t \beta_j x_{t,j} + \varepsilon_t, t = 1, 2, \ldots T, \tag{49}$$

where β_j is the weight of the ith chosen stock, ε_t is the error term. In practical application, the optimal estimate of β means the proportion of each stock. For example, if $\hat{\beta}_1 = 1$, $\hat{\beta}_2 = 2$, then when tracking the stock index, for each unit of labeled 1 stock held, it is necessary to hold 2 units of labeled 2 stock.

The bias measure for tracking, called Annual Tracking Error (ATE), is defined by

$$Tracking\ Error_{Year} = \sqrt{252} \times \sqrt{\frac{\sum (err_t - mean(err_t)^2}{T-1}}, \tag{50}$$

where $err_t = \hat{y}_t - y_t$ and $\hat{y}_t$ is the fitted or predicted value of y_t ,for $t = 1, 2, \ldots, T$.

The results of pcLAD (nLADlasso) and classo (nlasso) are shown in Table 11. In SSE 50 index tracking, we select 5, 10 and 20 component stocks respectively, and in CSI 300 index, we select 25, 30 and 40 component stocks. For the same number of non-zero coefficients, we only record the model with smallest training ATE value. From Table 11, in all the tracking experiments, the ATE of pcLAD method is better than that of classo, whether it is 80 days of modeling or 20 days of forecasting.

We are not surprised to see such a tracking result. Firstly, the data in financial market rarely satisfy Gauss's hypothesis. Secondly, this year's outbreak of novel coronavirus has led to more turbulence and uncertainty in stock index. Hence the reliability of the usual OLS-based estimation and model selection method is severely challenged, whereas the LAD-based methods become more attractive. In addition, with the increase of non-zero coefficients, the values of both ATE are decreasing. The reason for this

Table 11 SSE 50 and CSI 300 index tracking data

Number	SSE 50			Number	CSI 300		
	Method	ATE_{train}	ATE_{test}		Method	ATE_{train}	ATE_{test}
5	pcLAD	141.1	406.7	25	pcLAD	230.5	593.6
	Classo	289.3	409.1		Classo	950.2	604.0
10	pcLAD	111.6	406.5	30	pcLAD	211.2	593.4
	Classo	244.3	408.4		Classo	924.2	603.6
20	pcLAD	61.71	406.4	40	pcLAD	116.4	592.6
	Classo	214.7	408.1		Classo	868.6	602.9

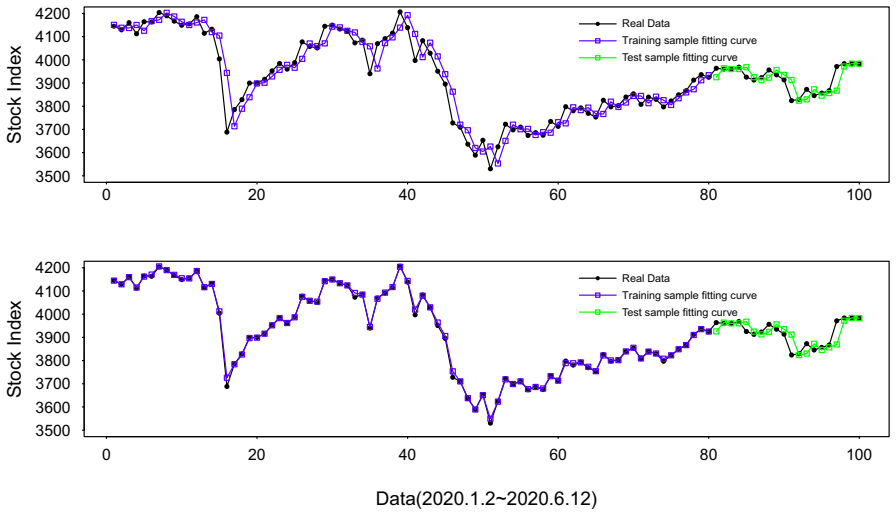

Fig. 4 The fitted and predicted results about tracking CSI 300 index

phenomenon is that the true coefficients are all positive. More non-zero coefficients are selected, smaller ATE will be got. However, in the financial market, holding more stocks means more costs, so sparsity is indispensable in stock index tracking. Finally, we show the fitted and predicted results of classo and pcLAD selecting 30 stocks to track CSI 300 in Fig. 4.

7 Discussion

When the noise does not obey Gaussian or near Gaussian, pcLAD is an effective alternative to classo method. In this paper, we prove that effective constraints can improve the accuracy of pcLAD estimation. In the fixed dimension, the constraints will reduce the variance of the estimation, and in the high dimension, the constraints will reduce the upper bound of the estimation bias. Furthermore, two algorithms named

as linear programming and nested ADMM are proposed to solve pLAD effectively in fixed and high dimension respectively.

However, there are still many further works to be studied. In this paper, we assume that k is less than infinity when p is of order $e^{n^{\pi}}$, where $0 < \pi < 1$. This assumption is more strict than $p < n$ and limits the number of equality constraints m. How to generalize the theoretical results to the case that both k and m tend to infinity with the growth of n is a challenging work. The upper bound (23) in high dimension can also be improved, because when the equality constraints m increases, the upper bound will not continue to decrease, especially when $m = k$, the upper bound is equal to the unconstrained case.

The derivation method in theory and algorithm of this paper can be well applied to more general model such as constrained Huber's estimation, quantile and composite quantile estimation (Gu and Zou 2020). For other penalty terms constrained regression such as elastic net, SCAD, MCP (Zhang 2010), the idea of nested ADMM is also available, but the theoretical analysis needs more technical methods. In fixed and high dimension, although the two proposed algorithms can be applied to generalized lasso and constrained generalized lasso, their theoretical analysis cannot be included in the framework of pcLAD. It's also a challenge to get Oracle or near Oracle property of generalized lasso and constrained generalized lasso with other assumptions.

Recently, parallel algorithms have been applied to large scale penalized regresion, such as Liqun et al. (2017) and Fan et al. (2020), and achieved good performance in numerical experiments. Extending nested ADMM to parallel algorithms' framework is a potentially valuable work for big data.

Appendix A

Proof of Lemma 1 When $r = p$, assume $\theta = D\beta$, then (8) can be rewrite as

$$\left\| y - XD^{-1}\theta \right\|_1 + n\lambda \left\| \theta \right\|_1 . \tag{51}$$

When $r < p$, we construct a $p \times p$ matrix $\tilde{D} = \begin{pmatrix} D \\ E \end{pmatrix}$ with $rank(\tilde{D}) = p$, by finding a $(p - r) \times p$ matrix E, whose rows are orthogonal to those in D. Then we change variables to $\theta = \tilde{D}\beta = (\theta_1', \theta_2')'$, so that the generalized LADlasso (8) becomes

$$\hat{\theta} = \underset{\theta \in R^p}{\arg\min} \left\{ \left\| y - X\tilde{D}^{-1}\theta \right\|_1 + n\lambda \left\| \theta \right\|_1 \right\}. \tag{52}$$

This is almost a regular LADlasso, except that L_1 penalty only convers part of the coefficient vector. we write $X\tilde{D}^{-1}\theta = X_1\theta_1 + X_2\theta_2$, then it is clear that at the solution the second block of the coefficients can be given by linear LAD regression:

$$\hat{\theta}_2 = \underset{\theta_2 \in R^{p-r}}{\arg\min} \left\{ \left\| y - X_1\theta_1 - X_2\theta_2 \right\|_1 \right\}. \tag{53}$$

Therefore, we can rewrite (52) as

$$\hat{\theta}_1 = \underset{\theta_1 \in R^r}{\arg\min} \left\{ \left\| y - X_1\theta_1 - X_2\hat{\theta}_2 \right\|_1 + n\lambda \left\| \theta_1 \right\|_1 \right\}. \tag{54}$$

we use (53) and (54) get θ, then $\beta = \tilde{D}^{-1}\theta$, so, when $r \leq p$, (8) can be seen as LADlasso problem.

When $r > p$, since D has full column rank, we can write D as $D = \begin{pmatrix} D_1 \\ D_2 \end{pmatrix}$, where $D_1 \in R^{p\times p}$ is an invertible matrix and $D_2 \in R^{(r-p)\times p}$. Then,

$$\begin{aligned} \|y - X\beta\|_1 + n\lambda \|D\beta\|_1 &= \left\| y - XD_1^{-1}D_1\beta \right\|_1 + n\lambda \|D_1\beta\|_1 + n\lambda \|D_2\beta\|_1 \\ &= \left\| y - (XD_1^{-1})D_1\beta \right\|_1 + n\lambda \|D_1\beta\|_1 \\ &\quad + n\lambda \left\| D_2D_1^{-1}D_1\beta \right\|_1. \end{aligned} \tag{55}$$

Using the change of variables, $\theta_1 = D_1\beta, \theta_2 = D_2D_1^{-1}D_1\beta = D_2D_1^{-1}\theta_1$, and $\theta = (\theta_1', \theta_2')'$.

we can rewrite the generalized LADlassso problem as follows:

$$\begin{aligned} \underset{\beta\in R^p}{\arg\min} \|y - X\beta\|_1 + n\lambda\|D\beta\|_1 &= \underset{\theta\in R^r}{\arg\min} \left\{ \left\| y - (XD_1^{-1})\theta_1 \right\|_1 \right. \\ &\quad \left. + n\lambda\|\theta\|_1;\, D_2D_1^{-1}\theta_1 - \theta_2 = 0 \right\} \\ &= \min_{\theta\in R^r} \left\{ \frac{1}{2} \left\| y - \tilde{X}\theta \right\|_1 + n\lambda\|D\theta\|_1;\, C\beta = 0 \right\}, \end{aligned}$$

where $\tilde{X} = (XD^{-1}, 0)$ and $C = (D_2D_1^{-1}, -I)$. Note that $\beta = \left(D^{-1}, 0\right)\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}$. Thus, this generalized LADlasso is a special case of the constrained LADlasso. □

Proof of Lemma 4 From Section 3.1 of Wang (2013), we know $\|h_A\|_1 \geq \bar{c}\,\|h_B\|_1$, consider $\left\|h_{A_1}\right\|_1 \leq \Phi \left\|h_{A_2}\right\|_1$, $\|h_A\|_1 = \left\|h_{A_1}\right\|_1 + \left\|h_{A_2}\right\|_1$, we arrive at a conclusion that $\left\|h_{A_2}\right\|_1 \geq \frac{\bar{c}}{1+\Phi}\|h_B\|_1$.

We give a technical Lemma 5 to prove Theorem 1. Define a linear approximation to $|\varepsilon_i - t|$ by $D_i = sign(\varepsilon_i) = I(\varepsilon_i > 0) - I(\varepsilon_i < 0)$. One intuitive interpretation of D_i is that D_i can be thought of as the first derivative of $|\varepsilon_i - t|$, at $t = 0$ (Pollard 1991). Moreover, the condition that ε_i has the median zero, implies $E(D_i) = 0$. Then, define $W_n = \sum_{i=1}^{n} D_i x_i'/\sqrt{n}$, and $W_{n,11} = \sum_{i=1}^{n} D_i x_{i1}'/\sqrt{n}$. We draw a conclusion $W_n \xrightarrow{L} N(0, \sum)$, $W_{n,11} \xrightarrow{L} N(0, \sum_{11})$ as Wang et al. (2007), Wu and Liu (2009). □

Lemma 5 *For model (13) with true parameter β_0, denote $G_n(u) = \sum_{j=1}^{n}\left(\left|\varepsilon_i - x_i'u/\sqrt{n}\right| - |\varepsilon_i|\right)$, where $\varepsilon_i = y_i - x_i'\beta_0$, under condition Assumption 1*

and 2, we have for any fixed u, satisfying $Cu = 0$,

$$G_n(u) = f(0)u' \sum_{j=1}^{n} x_i x_i' u + W_n' u + o_P(1).$$

Detailed proof of this Lemma can be found at Lemma 1 of Wang et al. (2007) and Lemma 3 of Wu and Liu (2009). The equality constraint $Cu = 0$ *will not affect the proof.* □

Appendix B

Proof of Lemma 2 For any given $\delta > 0$, there exists a large constant R such that

$$P\left\{ \inf_{\|u\|_2 = R, Cu=0} Q(\beta_0 + u/\sqrt{n}) > Q(\beta_0) \right\} \geq 1 - \varepsilon \tag{56}$$

where $u = (u_1, u_2, \ldots, u_p)'$. Due to $Cu = 0$ and $C\beta_0 = 0$, then $C(\beta_0 + u/\sqrt{n}) = 0$. Therefore, $\beta_0 + u/\sqrt{n}$ satisfy the $C(\beta_0 + u/\sqrt{n}) = 0$. From the fact $Q(\beta) = \sum_{i=1}^{n} |y_i - x_i^T \beta| + \sum_{j=1}^{p} \lambda_j |\beta_j|$ is convex and piecewise liner, the inequality (56) implies, with probability at least $1 - \delta$, the pcLAD estimator lies in the ring $\{\beta_0 + u/\sqrt{n} : \|u\|_2 \leq R, Cu = 0\}$. This in trun implies that there exists a local minimizer such that $\left\| \hat{\beta} - \beta_0 \right\|_2 = O_P(n^{-\frac{1}{2}})$, which is exactly what we want to show. Therefore, to prove Lemma 2, we only need to verify that (56) holds.

Note that

$$\begin{aligned} Q(\beta_0 + u/\sqrt{n}) - Q(\beta_0) &= \sum_{i=1}^{n} \left[\left| y_i - x_i'(\beta_0 + u/\sqrt{n}) \right| - \left| y_i - x_i' \beta_0 \right| \right] \\ &\quad + n \sum_{j=1}^{p} \lambda_j \left(\left| \beta_{0j} + u_j/\sqrt{n} \right| - \left| \beta_{0j} \right| \right) \\ &\geq G_n(u) - \sqrt{n} a_n \sum_{j=1}^{k} \left| u_j \right| \\ &= \frac{1}{n} f(0) u' \sum_{i=1}^{n} x_i x_i' u + W_n' u + o_P(1) \end{aligned}$$

As Wu and Liu (2009), we can point out that $W_n' u = E(W_n' u) + O_P(\sqrt{Var(W_n' u)})$, together with $Var(W_n' u) = E(\sum_{i=1}^{n} D_i x_i' u/\sqrt{n})^2 = \frac{1}{n} u' \sum_{i=1}^{n} x_i x_i' u$, implies $W_n' u = O_P(\sqrt{u' \sum_{i=1}^{n} x_i x_i' u/n})$.

The last equality follows from the Lemma 3 and $\sqrt{n}a_n \to 0$. By applying the Lemma 2 of Wu and Liu (2009), then $Q(\beta_0 + u/\sqrt{n}) - Q(\beta_0) \xrightarrow{L} \frac{1}{n} f(0) u' \Sigma u + W_n' u + o_P(1)$, where Σ is a positive definite matrix in Assumption 2.

Based on all the above, $Q(\beta_0 + u/\sqrt{n}) - Q(\beta_0)$ is dominated by the quadratic term $\frac{1}{n} f(0) u^T \Sigma u$ when $\|u\|_2$ is enough large. This completes the proof of Lemma 2. □

Proof of Lemma 3 For any $\beta_A - \beta_{A0} = O_P(n^{-1/2}), 0 \le \|\beta_B\|_2 \le Rn^{-1/2}$, and $C(\beta - \beta_0) = 0$,

$$\begin{aligned} & Q\left[(\beta_A', 0')'\right] - Q\left[(\beta_A', \beta_B')'\right] \\ &= \left\{Q\left[(\beta_A', 0')'\right] - Q\left[(\beta_{A0}', 0')'\right]\right\} - \left\{Q\left[(\beta_A', \beta_B')'\right] - Q\left[(\beta_{A0}', 0')'\right]\right\} \\ &= G_n\left[\sqrt{n}((\beta_A - \beta_{A0})', 0')'\right] - G_n\left[\sqrt{n}((\beta_A - \beta_{A0})', \beta_B')'\right] - n\sum_{j=k+1}^{p} \lambda_j |\beta_j|. \end{aligned}$$

The conditions $\beta_A - \beta_{A0} = O_P(n^{-1/2}), 0 \le \|\beta_B\|_2 \le Rn^{-1/2}$, and $\frac{1}{n}\sum x_i x_i' = \text{trace}(\Sigma)$ imply that

$$\begin{aligned} G_n\left[\sqrt{n}((\beta_A - \beta_{A0})', 0')'\right] &= f(0)\sqrt{n}((\beta_A - \beta_{A0})', 0')' \frac{1}{n}\sum_{i=1}^{n} x_i x_i' \\ &\quad \sqrt{n}((\beta_A - \beta_{A0})', 0')' = O_P(1), \\ G_n\left[\sqrt{n}((\beta_A - \beta_{A0})', \beta_B')'\right] &= f(0)\sqrt{n}((\beta_A - \beta_{A0})', \beta_B')' \frac{1}{n}\sum_{i=1}^{n} x_i x_i' \\ &\quad \sqrt{n}((\beta_A - \beta_{A0})', \beta_B')' = O_P(1). \end{aligned}$$

Moreover, $n\sum_{j=k+1}^{p} \lambda_j |\beta_j| = \sqrt{n}(\sqrt{n}b_n)\sum_{j=1}^{p} |\beta_j|$. Hecen the condition that $\sqrt{n}b_n \to +\infty$, implies that $n\sum_{j=k+1}^{p} \lambda_j |\beta_j|$ is of higher order than any other terms and as a result. This in turn implies that $Q\left[(\beta_A', 0')'\right] - Q\left[(\beta_A', \beta_B')'\right] < 0$ for large n. This proves the consistency of model selection of pcLAD. □

Proof of Theorem 1 Similarly as in Fan and Li (2001) and Wang et al. (2007), part $\langle a \rangle$ holds simply due to Lemma 3. Next we prove part $\langle b \rangle$. From theorem of Wang et al. (2007) and Theorem 3 of Wu and Liu (2009), we can obtain the result

$$\begin{aligned} \min_u \sum_{i=1}^{n} \Big\{ & \left| y_i - x_i'\beta_{A0} - n^{-1/2} x_{Ai}' u_A \right| \\ & - \left| y_i - x_{Ai}'\beta_{A0} \right| \Big\} \xrightarrow{L} \min_u \left\{ f(0) u_A' \Sigma_{11} u_A + u_A' w_0 \right\}, \end{aligned}$$

where w_0 is a k dimension normal random vector with mean 0 and variance matrix Σ_{11}, $\sqrt{n}(\hat{\beta_A} - \beta_{A0}) \xrightarrow{L} N(0, \frac{1}{4f^2(0)}\Sigma_{11}^{-1})$.

For the constraints $C_A\beta = b, C_A u_A = 0$, we can use the method proposed by Wang (1995) and Wang (1996), we can get the minimizer $\hat{u}_A$ of $\min_{u_A, C_A u_A = 0} \sum_{i=1}^{n} \left\{ \left| y_i - x_i'\beta_{A0} - n^{-1/2} x_{Ai}' u_A \right| - \left| y_i - x_{Ai}'\beta_{A0} \right| \right\}$ will convergence to

$$\min_{u_A, C_A u_A = 0} f(0) u_A' \Sigma_{11} u_A + u_A' w_0 \tag{57}$$

in distribution.

Following the Assumption 2 is a positive definite matrix. According the KKT condition, $\hat{u}_A$ is the minimizer of (52), if and only if

$$\begin{cases} 2f(0)\Sigma_{11}\hat{u}_A + w_0 + C_A' v = 0 \\ C_A \hat{u}_A = 0 \end{cases} \tag{58}$$

where v is a $m \times 1$ dimesional Lagrange multiplier, we can transform (58) to the following formula:

$$\begin{pmatrix} 2f(0)\Sigma_{11} & C_A' \\ C_A & 0 \end{pmatrix} \begin{pmatrix} \hat{u}_A \\ v \end{pmatrix} = 0$$

Let

$$B = \begin{pmatrix} 2f(0)\Sigma_{11} & C_A' \\ C_A & 0 \end{pmatrix},$$

B is invertibble, because of C_A is full of row rank. By routine calculation, we get

$$B^{-1} = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}, \begin{cases} B_{11} = \frac{\Sigma_{11}^{-1}}{2f(0)} \left[I - C_A' \left(C_A \Sigma_{11}^{-1} C_A' \right)^{-1} C_A \Sigma_{11}^{-1} \right] \\ B_{12} = \Sigma_{11}^{-1} C_A' \left(C_A \Sigma_{11}^{-1} C_A' \right)^{-1} \\ B_{21} = B_{12} \\ B_{22} = -2f(0) \left(C_A \Sigma_{11}^{-1} C_A' \right)^{-1} \end{cases}$$

Hence, we have

$$\begin{cases} \hat{u}_A = B_{11}(-w_0) \\ v = B_{21}(-w_0) \end{cases}$$

Therefore, $\hat{u}_A = N(0, \frac{\Sigma_{11}^{-1}}{4f^2(0)}(I - V_A)'(I - V_A))$, $V_A = C_A' \left(C_A \Sigma_{11}^{-1} C_A' \right)^{-1} C_A \Sigma_{11}^{-1}$. □

Appendix C

Proof of Theorem 2 Without loss of generality, we assume $|h_1| \geq |h_2| \geq \cdots \geq |h_p|$ as Wang (2013) and Wang et al. (2019). Let $S_0 = \{1, 2, \ldots, m\}$,$S_1 = \{m + 1, m + 2, \ldots, k\}$,$S_2 = \{k + 1, k + 2, \ldots, 2k - m\}$, $S_0 = \{2k - m + 1, 2k - m + 2, \ldots, 3k - m\}$,.... Due to $h \in \Delta_{\bar{c}} = \left\{ \delta \in R^p : \left\| \delta_{A_2} \right\|_1 \geq \frac{\bar{c}}{1+\Phi} \| \delta_B \|_1 \right\}$, we have

$\|h_{s_1}\|_1 \geq \frac{\bar{c}}{1+\Phi}\|h_B\|_1 = \frac{\bar{c}}{1+\Phi}\sum_{i\geq 2}\|h_{s_i}\|_1$. Then it follows from Lemma 8 of Wang (2013) that

$$\sum_{i\geq 2}\|h_{s_i}\|_2 \leq \frac{\sum_{i\geq 1}\|h_{s_i}\|_1}{\sqrt{k_0}} + \frac{\sqrt{k_0}}{4}\|h_{k_0+1}\|_1 \leq \frac{\sum_{i\geq 1}\|h_{s_i}\|_1}{\sqrt{k_0}} + \frac{1}{4\sqrt{k_0}}\|h_{s_1}\|_1$$
$$\leq \left(\frac{1+\Phi+\bar{c}}{\bar{c}\sqrt{k_0}} + \frac{1}{4\sqrt{k_0}}\right)\|h_{s_1}\|_1 \leq \left(\frac{1+\Phi+\bar{c}}{\bar{c}} + \frac{1}{4}\right)\|h_{s_1}\|_2, \quad (59)$$

where $k_0 = k - m$.

It is easy to see that

$$\frac{1}{\sqrt{n}}\left(\|Xh+\varepsilon\|_1 - \|\varepsilon\|_1\right) \geq \sum_{i\geq 1}\frac{1}{\sqrt{n}}\left(\left\|X\sum_{j=0}^{i}h_{s_j}+\varepsilon\right\|_1 - \left\|X\sum_{j=0}^{i-1}h_{s_j}+\varepsilon\right\|_1\right)$$
$$+\frac{1}{\sqrt{n}}\left(\|Xh_{s_0}+\varepsilon\|_1 - \|\varepsilon\|_1\right).$$

Now for any fixed vector d, let

$$M(d) = \frac{1}{\sqrt{n}}E\left(\|Xh_{s_0}+\varepsilon\|_1 - \|\varepsilon\|_1\right).$$

By Lemma 3 of Wang (2013), for $p > n$ and $p > 3\sqrt{\max(m,k_0)}$, we know that with probability at least $1-2p^{-4m(c_2^2-1)+1}$,

$$\frac{1}{\sqrt{n}}\left(\|Xh_{s_0}+\varepsilon\|_1 - \|\varepsilon\|_1\right) \geq M\left(h_{s_0}\right) - c_1\sqrt{2m\log(p)}\|h_{s_0}\|_2, \quad (60)$$

and for any $i \geq 1$ with probability at least $1-2p^{-4k_0(c_2^2-1)+1}$,

$$\frac{1}{\sqrt{n}}\left(\left\|X\sum_{j=0}^{i}h_{s_j}+\varepsilon\right\|_1 - \left\|X\sum_{j=0}^{i-1}h_{s_j}+\varepsilon\right\|_1\right) \geq M\left(h_{s_i}\right) - c_1\sqrt{2k_0\log(p)}\|h_{s_i}\|_2,$$

where $c_1 = 1+2c_2\sqrt{\lambda_k^u}$ and $c_2 > 1$ is a constant. Put the above inequalities together, we know that with the probability at least $1-2p^{-4\min(m,k_0)(c_2^2-1)+1}$,

$$\frac{1}{\sqrt{n}}\left(\|Xh+\varepsilon\|_1 - \|\varepsilon\|_1\right) \geq M(h) - c_1\sqrt{2\max(m,k_0)\log(p)}\|h_{s_i}\|_2.$$

Due to $\|Xh+\varepsilon\|_1+n\lambda\left\|\hat{\beta}\right\|_1 \le \|\varepsilon\|_1+n\lambda\|\beta_0\|_1$, $\|Xh+\varepsilon\|_1-\|\varepsilon\|_1 \le n\lambda\|h_A\|_1$. By (59), we have

$$\begin{aligned}\frac{1}{\sqrt{n}}\left(\|Xh+\varepsilon\|_1-\|\varepsilon\|_1\right) &\le \sqrt{n}\lambda\|h_A\|_1 \le \sqrt{n}\lambda\left(\left\|h_{s_0}\right\|_1+\left\|h_{s_1}\right\|_1\right)\\ &\le \sqrt{n}\lambda\left(1+\Phi\right)\left\|h_{s_1}\right\|_1 \le \sqrt{n}\lambda\left(1+\Phi\right)\sqrt{k_0}\left\|h_{s_1}\right\|_2,\\ \sum_{i\ge 0}\left\|h_{s_i}\right\|_2 &\le \Phi\left\|h_{s_1}\right\|_2+\left\|h_{s_1}\right\|_2+\left(\frac{1}{4}+\frac{\Phi+\bar{c}+1}{\bar{c}}\right)\left\|h_{s_1}\right\|_2\\ &= \left(\frac{5}{4}+\frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}\right)\left\|h_{s_1}\right\|_2.\end{aligned}$$

Put the above inequalities together, we have that with the probability at least $1-2p^{-4\min(m,k_0)(c_2^2-1)+1}$,

$$\begin{aligned}M(h) &\le \sqrt{n}\lambda\left(1+\Phi\right)\sqrt{k_0}\left\|h_{s_1}\right\|_2\\ &+c_1\sqrt{2\max(m,k_0)log(p)}\left(\frac{5}{4}+\frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}\right)\left\|h_{s_1}\right\|_2.\end{aligned}\tag{61}$$

From Lemma 5 and Lemma 7 of Wang (2013), we have

$$\begin{aligned}M(h)=\frac{1}{\sqrt{n}}E\left(\|Xh+\varepsilon\|_1-\|\varepsilon\|_1\right)&=\frac{1}{\sqrt{n}}E\left[\sum_{i=1}^{n}|(Xh)_i+\varepsilon_i|-|\varepsilon_i|\right]\\ \ge\frac{1}{\sqrt{n}}\frac{a}{16}\left[\sum_{i=1}^{n}|(Xh)_i|\left(|(Xh)_i|\wedge\frac{b}{a}\right)\right]&\ge\begin{cases}\frac{3}{16\sqrt{n}}\frac{\|Xh\|_1}{2}, \|Xh\|_1\ge\frac{3n}{a}\\ \frac{a}{16\sqrt{n}}\|Xh\|_2^2, \|Xh\|_1<\frac{3n}{a}.\end{cases}\end{aligned}\tag{62}$$

Therefore, if $\|Xh\|_1 \ge \frac{3n}{a}$, from inequality (62), we have

$$\begin{aligned}M(h)=\frac{1}{\sqrt{n}}E\left(\|Xh+\varepsilon\|_1-\|\varepsilon\|_1\right)&\ge\frac{3}{16\sqrt{n}}\|Xh\|_1\\ &\ge\frac{3}{16}\sqrt{n}k_k^l\|h_A\|_2\ge\frac{3\sqrt{n}}{16}k_k^l\left\|h_{s_1}\right\|_2.\end{aligned}$$

Then, from condition (22) and inequality (62), we can show $\left\|h_{s_0}\right\|_2=0$ with probability at least $1-2p^{-4\min(m,k_0)(c_2^2-1)+1}$. Since $|h_1|\ge|h_2|\ge\cdots|h_p|$, we have $\|h\|_2=0$, i.e., $\hat{\beta}=\beta^*$ holds with probability at least $1-2p^{-4\min(m,k_0)(c_2^2-1)}$. Then, if $\|Xh\|_1<\frac{3n}{a}$, from inequality (62)

$$M(h)=\frac{1}{\sqrt{n}}E\left(\|Xh+\varepsilon\|_1-\|\varepsilon\|_1\right)\ge\frac{a}{16\sqrt{n}}\|Xh\|_2^2.\tag{63}$$

By the same argument as in the proof of Theorem 3.1 and 3.2 in Cai et al. (2010), we know that

$$|\langle Xh_A, Xh\rangle| \geq n\lambda_k^l \|h_A\|_2^2 - n\theta_k^k \|h_A\|_2 \sum_{i\geq 2} \|h_{s_i}\|_2 \geq n\left(\lambda_k^l - \theta_k^k\left(\frac{1+\Phi}{\bar{c}}\right)\right)\|h_A\|_2^2$$
$$\geq n\left(\lambda_k^l - \theta_k^k\left(\frac{1+\Phi}{\bar{c}}\right)\right)\|h_{s_1}\|_2^2.$$

and

$$|\langle Xh_A, Xh\rangle| \leq \|Xh_A\|_2\|Xh\|_2 \leq \|Xh\|_2\sqrt{n\lambda_k^u}\|h_A\|_2$$
$$\leq \|Xh\|_2\sqrt{n\lambda_k^u}\left(\sqrt{1+\Phi}\|h_{s_1}\|_2\right).$$

Therefore, $\|Xh\|_2^2 \geq \frac{n[\lambda_k^l - \theta_k^k(\frac{1+\Phi}{\bar{c}})]^2}{\lambda_k^u(1+\Phi)}\|h_{s_1}\|_2^2$.

Hence by (62) and (63), let $\eta_k^l = \frac{[\lambda_k^l - \theta_k^k(\frac{1+\Phi}{\bar{c}})]^2}{\lambda_k^u(1+\Phi)}$, $\lambda = 2c\sqrt{\log(p)/n}$, we have that with probability at least $1 - 2p^{-4\min(m,k_0)(c_2^2-1)+1}$

$$\|h_{s_1}\|_2 \leq \frac{32c(1+\Phi)}{a\eta_k^l}\sqrt{\frac{k_0\log(p)}{n}}$$
$$+ \frac{16c_1\sqrt{2\max(m,k_0)\log(p)}}{a\eta_k^l\sqrt{n}}\left[\frac{5}{4} + \frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}\right]$$
$$\leq \left\{\frac{16[\sqrt{2}c(1+\Phi) + c_1(\frac{5}{4} + \frac{(\bar{c}+1)(\Phi+1)}{\bar{c}})]}{a\eta_k^l}\right\} \times \sqrt{\frac{2\max(m,k_0)\log(p)}{n}},$$

Furthermore, $\sum_{i\geq 2}\|h_{s_i}\|_2^2 \leq |h_{k+1}|\sum_{i\geq 1}\|h_{s_i}\|_1 \leq \frac{1}{\bar{c}}\|h_{s_1}\|_2^2$ and $\|h_0\|_2^2 \leq \Phi\|h_{s_1}\|_2^2$, we know that with probability at least $1 - 2p^{-4\min(m,k_0)(c_2^2-1)+1}$

$$\left\|\hat{\beta} - \beta\right\|_2 \leq \sqrt{1+\frac{1}{\bar{c}}+\Phi}\,\frac{16\left\{\sqrt{2}c(1+\Phi) + c_1[\frac{5}{4} + \frac{(\bar{c}+1)(\Phi+1)}{\bar{c}}]\right\}}{a\eta_k^l}$$
$$\times\sqrt{\frac{2\max(m,k_0)log(p)}{n}}.$$

□

References

Altenbuchinger M, Rehberg T, Zacharias HU et al (2017) Reference point insensitive molecular data analysis. Bioinformatics 2:2

Bassett GW, Koenker R (1978) Asymptotic theory of least absolute error regression. J Am Stat Assoc 73(363):618–622
Belloni A, Chernozhukov V (2011) L_1-penalized quantile regression in high-dimensional sparse models. Ann Stat 39(1):82–130
Belloni A, Chernozhukov V, Wang L (2011) Square-root lasso: pivotal recovery of sparse signals via conic programming. Biometrika 4(4)
Bhlmann P, van de Geer S (2011) Statistics for high-dimensional data. Springer, Berlin
Bickel PJ, Ritov Y, Tsybakov AB (2009) Simultaneous analysis of lasso and Dantzig selector. Ann Stat 37(4):1705–1732
Bredel M, Bredel C, Juric D, Harsh GR, Vogel H, Recht LD, Sikic BI (2005) High-resolution genome-wide mapping of genetic alterations in human glial brain tumors. Can Res 65:4088–4096
Berman A (1973) Cones, matrices and mathematical programming. Springer, Berlin
Boyd S, Parikh N, Chu E et al (2010) Distributed optimization and statistical learning via the alternating direction method of multipliers
Cai TT, Wang L, Xu G (2010) New bounds for restricted isometry constants. IEEE Trans Inf Theory 56(9):4388–4394
Efron B, Hastie T, Johnstone I, Tibshirani R (2004) Least angle regression. Ann Stat 32:407–499
El-Arini K, Xu M, Fox EB, Guestrin C (2013) Representing documents through their readers. In: Proceedings of the 19th association for computing machinery international conference on knowledge discovery and data mining, pp 14–22
Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its Oracle properties. J Am Stat Assoc 96:1348–1360
Fan Y, Lin N, Yin X (2020) Penalized quantile regression for distributed big data using the slack variable representation. J Comput Graph Stat 1–22
Gaines BR, Kim J, Zhou H et al (2018) Algorithms for Fitting the constrained lasso. J Comput Graph Stat 27(4):861–871
Gabay D, Mercier B (1976) A dual algorithm for the solution of nonlinear variational problems via finite element approximation[J]. Comput Math Appl 2(1):17–40
Gao X, Huang J (2010) Asymptotic analysis of high-dimensional LAD regression with lasso smoother. Stat Sin 20(4):187–193
Geyer CJ (1994) On the asymptotics of constrained M-estimation. Ann Stat 22:1993–2010
Glowinski R, Marrocco A (1975) Sur l'approximation par éléments finis d'ordre un et la résolution par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires[J]. Revue franaise d automatique informatique recherche opérationnelle Mathématique 2(R–2):41–76
Gu Y, Zou H (2020) Sparse composite quantile regression in ultrahigh dimensions with tuning parameter calibration. IEEE Trans Inf Theory PP(99):1–1
Gu Y, Fan J, Kong L et al (2017) ADMM for high-dimensional sparse penalized quantile regression. Technometrics
He T (2011) Lasso and general L1-regularized regression under linear equality and inequality constraints (Ph.D. thesis), Purdue University, West Lafayette, IN
Hu Z, Follmann DA, Miura K (2015a) Vaccine design via nonnegative lasso based variable selection. Stat Med 34:1791–1798
Hu Q, Zeng P, Lin L (2015b) The dual and degrees of freedom of linearly constrained generalized lasso. Comput Stat Data Anal 86:13–26
Huber P (1981) Robust statistics. Wiley, New York
James GM, Paulson C, Rusmevichientong P (2013) Penalized and constrained regression, Unpublished Manuscript, University of Southern California
James GM, Paulson C, Rusmevichientong P et al (2020) Penalized and constrained optimization: an application to high-dimensional website advertising. J Am Stat Assoc 115(529):107–122
Jones P, Parker D, Osborn T, Briffa K (2016) Global and hemispheric temperature anomalies—land and marine instrumental records, trends: a compendium of data on global change
Koenker R, Ng P (2005) Inequality constrained quantile regression. Sankhyā: The Indian Journal of Statistics (2003-2007) 67(2):418–440
Kump P, Bai EW, Chan KS, Eichinger B, Li K (2012) Variable selection via RIVAL (removing irrelevant variables amidst lasso iterations) and its application to nuclear material detection. Automatica 48:2107–2115

Lambert-Lacroix S, Zwald L (2011) Robust regression through the Huber's criterion and adaptive lasso penalty. Electron J Stat 5:1015–1053
Lange K (2013) Optimization, 2nd edn. Springer, New York
Leng C, Lin Y, Wahba G (2004) A note on the lasso and related procedures in model selection. Stat Sin 16(4)
Li N, Yang H (2019) Nonnegative estimation and variable selection under minimax concave penalty for sparse high-dimensional linear regression models. Stat Pap
Li Y, Zhu J (2008) L1-norm quantile regression. J Comput Graph Stat 17:163–185
Li N, Yang H, Yang J et al (2019) Nonnegative estimation and variable selection via adaptive elastic-net for high-dimensional data. Commun Stat 1–17
Liew CK (1976) Inequality constraint least-squares estimation. J Am Stat Assoc 71(355):746–751
Lin W, Shi P, Feng R, Li H (2014) Variable selection in regression with compositional covariates. Biometrika 101:785–797
Liqun Yu, Lin N, Wang L (2017) A parallel algorithm for large-scale nonconvex penalized quantile regression. J Comput Graph Stat 26(4):935–939
Liu Y, Zeng P, Lin L (2020) Generalized 1-penalized quantile regression with linear constraints. Comput Stat Data Anal 142
Mandal BN, Ma J (2016) l1 regularized multiplicative iterative path algorithm for non-negative generalized linear models. Comput Stat Data Anal 101:289–299
Meinshausen N (2007) Relaxed lasso. Comput Stat Data Anal 52(1):374–393
Meinshausen N (2013) Sign-constrained least squares estimation for high-dimensional regression. Electron J Stat 7(2):1607–1631
Michels E, De Preter K, Van Roy N, Speleman F (2007) Detection of DNA copy number alterations in cancer by array comparative genomic hybridization. Genet Med 9:574–584
Negahban SN, Ravikumar P, Wainwright MJ et al (2010) A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. Stat Sci 27(4):538–557
Parker T (2019) Asymptotic inference for the constrained quantile regression process. J Econ
Peng B, Wang L (2015) An iterative coordinate descent algorithm for high-dimensional nonconvex penalized quantile regression. J Comput Graph Stat
Pollard D (1991) Asymptotics for least absolute deviation regression estimators. Econ Theory 7(2):186–199
Portnoy S, Koenker R (1997) The Gaussian hare and the Laplacian tortoise: computability of squared-error versus absolute-error estimators. Stat Sci 12(4):279–300
Shi P, Zhang A, Li H (2016) Regression analysis for microbiome compositional data. Ann Appl Stat 10:1019–1040
Silvapulle MJ, Sen PK (2005) Constrained statistical inference. Wiley, New York
Stellato B, Banjac G, Goulart P et al (2018) OSQP: an operator splitting solver for quadratic programs. Math Program Comput
Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc B 58:267–288
Tibshirani R, Suo X (2016) An ordered lasso and sparse time-lagged regression. Technometrics 58(4):415–423
Tibshirani RJ, Taylor J (2011) The solution path of the generalized Lasso. Ann Stat 39(3):1335–1371
Tibshirani R, Wang P (2008) Spatial smoothing and hot spot detection for CGH data using the fused lasso. Biostatistics 9:18–29
Tibshirani R, Saunders M, Rosset S, Zhu J (2005) Sparsity and smoothness via the fused lasso. J Roy Stat Soc B 67(1):91–108
Tibshirani RJ, Hoefling H, Tibshirani R (2011) Nearly-isotonic regression. Technometrics 53:54–61
Wang JD (1995) Asymptotic normality of L1-estimators in nonlinear regression. J Multivar Anal
Wang J (1996) Asymptotics of least-squares estimators for constrained nonlinear regression. Ann Stat 24(3):1316–1326
Wang L (2013) The penalized LAD estimator for high dimensional linear regression. J Multivar Anal
Wang H, Li G, Jiang G (2007) Robust regression shrinkage and consistent variable selection through the LAD-lasso. J Bus Econ Stat 25(3):347–355
Wang H, Kong L, Tao J et al (2019) The linearized alternating direction method of multipliers for sparse group LAD model. Optim Lett 13(3):505–525
Wu TT, Lange K (2008) Coordinate descent algorithms for lasso penalized regression. Ann Appl Stat 2:224–244
Wu Y, Liu Y (2009) Variable selection in quantile regression. Stata Sin 19(2):801–817

Wu L, Yang Y (2014) Nonnegative elastic net and application in index tracking. Appl Math Comput 227:541–552
Wu WB, Woodroofe M, Mentz G (2001) Isotonic regression: another look at the changepoint problem. Biometrika 88:793–804
Wu L, Yang Y, Liu H (2014) Nonnegative-lasso and application in index tracking. Comput Stat Data Anal 70:116–126
Xie WL, Yang H (2019) Nonnegative hierarchical lasso with a mixed $(1, \frac{1}{2})$ penalty and a fast solver. Stat Interface 12(4):599–615
Yang YH, Wu L (2016) Nonnegative adaptive lasso for ultra-high dimensional regression models and a two-stage method applied in financial modeling. J Stat Plan Inference
Yang J, Meng X, Mahoney MW (2013) Quantile regression for large-scale applications
Yen Y, Yen T (2014) Solving norm constrained portfolio optimization via coordinate-wise descent algorithms. Comput Stat Data Anal 76(76):737–759
Yu L, Lin N (2017) ADMM for penalized quantile regression in big data. Int Stat Rev
Zhang CH (2010) Nearly unbiased variable selection under minimax concave penalty. Ann Stat 38(2):894–942
Zhou H, Lange K (2013) A path algorithm for constrained estimation. J Comput Graph Stat 22(2):261–283
Zou H (2006) The adaptive LASSO and its Oracle properties. J Am Stat Assoc 101(476):1418–1429
Zou H, Hastie T, Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J R Stat Soc B 67(2):301–320