ARTICLE TEMPLATE

Solution Path Algorithm for Distributionally Robust Regression

Guangrui Tang[a] and Neng Fan[a]

[a]Department of Systems and Industrial Engineering, The University of Arizona, 1127 E. James E. Rogers Way, Tucson, Arizona, USA

ARTICLE HISTORY
Compiled April 5, 2024

ABSTRACT
In this paper, we propose a general distributionally robust regression model based on distributionally robust optimization theory. The proposed model has piecewise linear loss function and elastic net penalty term, and it generalizes many other regression models. We prove the piecewise linear property of the optimal solutions to this model, which enables us to develop a solution path algorithm for the hyperparameter tuning. Doubly regularized Least Absolute Deviations (DrLAD) regression model is proposed based on this framework, and a solution path algorithm is developed to speed up the tuning of two hyperparameters in this model. Numerical experiments are implemented to validate the performance of this model and the computational efficiency of the solution path algorithm.

KEYWORDS
Solution path algorithm; Distributionally robust optimization; Regression; Regularization; Hyperparameter tuning

1. Introduction

Regression is a technique to estimate the relationships between between input $\boldsymbol{x} \in \mathbb{R}^d$ and output $y \in \mathbb{R}$. The regression coefficients $\boldsymbol{\beta}$ and β_0, representing the relationships between input $\boldsymbol{x}$ and output y, can be estimated by solving expected risk minimization problem

$$\min_{\boldsymbol{\beta},\beta_0} \mathbb{E}^{\mathbb{P}}[\ell(z)], \tag{1}$$

where $\ell(z)$ is the loss function, and $z = y - (\boldsymbol{\beta}^\intercal \boldsymbol{x} + \beta_0)$ is the residual. Here, $\mathbb{P}$ is the input-output distribution.

Different loss functions will lead to different regression models. The Ordinary Least Squares (OLS) takes the squared error as the loss function, but it is very sensitive to outlier points. The Least Absolute Deviation (LAD) [1] model minimizes the absolute error loss, which is more robust to outliers. Huber regression [2,3] models combine the squared error loss and absolute loss, where the loss function is squared error for residuals around zero and is absolute error for residuals far away from zero, but the Huber loss introduces another hyperparameter to define the range for squared error and absolute error. Support vector regression [4] adopts the ϵ-insensitive loss to find

CONTACT Guangrui Tang Email: guangruitang@arizona.edu

the appropriate regression hyperplane, which is robust to outliers. The pinball loss is used in quantile regression [5], which can produce accurate quantile forecast.

In practical applications, data will contain noise that makes it deviate from original values; this is called data uncertainty. With data uncertainty, the model will suffer from overfitting and have bad generalization performance in test data. Regularization is often used to handle overfitting. Hoerl et al. [6] added the ℓ_2-norm regularization to OLS model, which can achieve better prediction performance. The ℓ_1-norm regularization [7] can shrink the coefficient and perform feature selection. As a combination of ℓ_1-norm and ℓ_2-norm, the elastic net regularization [8] has the benefits of both norms and also encourages a grouping effect for correlated input features.

Instead of directly using the uniform empirical distribution over the training data, distributionally robust optimization (DRO) [9,10] considers a family of distributions in an ambiguity set. The Wasserstein ambiguity set, constructed through Wasserstein distance, contains all distributions around the uniform empirical distribution of the training data samples within a certain probability distance [9,11,12]. This Wasserstein distance can be interpreted as the minimum transportation cost for moving unit mass of one distribution to the other distribution, and the corresponding ambiguity set guarantees that the unknown data-generating distribution belongs to the set with high confidence for a sufficiently large sample size. The DRO based on a Wasserstein ambiguity set is a popular technique to improve the performance of machine learning models, and its tractable reformulations will introduce a regularization term [13]. The coefficient of the regularization term, controlling the bias-variance trade-off between loss function and regularization, is a hyperparameter. The choice of this hyperparameter will highly influence the generalization performance of the model.

In some machine learning models, the optimal solutions to $\boldsymbol{\beta}, \beta_0$ will change piecewise linearly with respect to hyperparameters. Once the break points can be found sequentially, all the optimal solutions can be obtained efficiently. These kinds of algorithms are called solution path algorithms [14]. They can be used to speed up the tuning of hyperparameters, and have been studied in literature for some machine learning models. For example, the Least Angle Regression (LARS) algorithm was proposed by Efron et al. [15] to fit the Lasso model. Inspired by LARS, Zou et al. [8] proposed LARS-EN algorithm for elastic net regression model. Tibshirani et al. [16] further extended the solution path algorithm to generalized Lasso problems. Wang et al. [17] proposed the solution path algorithm for regularized least absolute deviations regression model. For support vector regression problem, λ-path [18], ϵ-path [19] and two-dimensional path algorithms [20] were developed to tune the hyperparameter. Takeuchi et al. [21] proposed a solution path algorithm for kernel quantile regression.

However, there is no well developed algorithm for the hyperparameter tuning of Wasserstein DRO regression model. This paper proposes a general distributionally robust regression model with piecewise linear loss function and elastic net regularization term. The optimal solutions to $\boldsymbol{\beta}, \beta_0$ in this model are piecewise linear with respect to the hyperparameters. As a special case, the Doubly Regularized Least Absolute Deviations (DrLAD) regression model, combining both the LAD loss and elastic net regularization, is developed, and a solution path algorithm is designed for tuning the two hyperparameters in this model.

The remainder of this paper is organized as follows. In Section 2, a general framework for regression problems is proposed. In section 3, the piecewise linear property of the optimal solutions is studied. In Section 4, a solution path algorithm for hyperparameter tuning in DrLAD model is designed. In Section 5, numerical experiments are implemented to test the performance of the model and the solution path algorithm.

Finally, Section 6 concludes this paper.

2. Distributionally robust optimization for regression problems

2.1. Distributionally robust regression models with Wasserstein ambiguity set

In practice, the distribution $\mathbb{P}$ in (1) is uncertain and only indirectly observable through the training data. Based on distributionally robust optimization theory, we can construct an ambiguity set which contains the unknown distribution $\mathbb{P}$ with high confidence. Let $\Xi = \mathbb{R}^d \times \mathbb{R}$, the Wasserstein distance between two distributions $\mathbb{Q}$ and $\mathbb{Q}'$ supported on Ξ is defined as:

$$W(\mathbb{Q}, \mathbb{Q}') = \inf_{\Pi} \Big\{ \int_{\Xi^2} d((\boldsymbol{x}, y), (\boldsymbol{x}', y'))\Pi((\boldsymbol{x}, y), (\boldsymbol{x}', y')) : \\ \Pi \text{ is a joint distribution of } \xi \text{ and } \xi' \text{ with marginals } \mathbb{Q} \text{ and } \mathbb{Q}' \Big\},$$

where $d((\boldsymbol{x}, y), (\boldsymbol{x}', y')) = \|\boldsymbol{x} - \boldsymbol{x}'\|_p + \kappa|y - y'|$ is a metric on Ξ, indicating the minimum cost of moving one distribution to another, and here p is the norm of feature space. The parameter $\kappa > 0$ reflects the relative emphasis between feature uncertainty and label mismatch. The larger the κ is, the more confidence in labels. Given the training data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$, the empirical distribution is defined as $\hat{\mathbb{P}} = \frac{1}{n}\sum_{i=1}^n \delta_{(\boldsymbol{x}_i, y_i)}$. Then an ambiguity set centered around $\hat{\mathbb{P}}$ can be constructed as follows:

$$\mathcal{B}_\lambda(\hat{\mathbb{P}}) = \{\mathbb{Q} : W(\mathbb{Q}, \hat{\mathbb{P}}) \le \lambda\},$$

where λ is radius of the Wasserstein ambiguity set. Thus, the following distributionally robust optimization model for regression problems can be formulated as

$$\min_{\boldsymbol{\beta}, \beta_0} h(\boldsymbol{\beta}) + \sup_{\mathbb{Q} \in \mathcal{B}_\lambda(\hat{\mathbb{P}})} \mathbb{E}^{\mathbb{Q}}[\ell(z)], \tag{2}$$

where $h(\boldsymbol{\beta})$ is a function of regression coefficients $\boldsymbol{\beta}$, and the sup is to identify the worst-case expected risk within the Wasserstein ambiguity set. This model (2) constitutes an infinite-dimensional optimization problem over probability distributions and thus appears to be intractable [9]. If $\Xi = \mathbb{R}^{d+1}$ and $\ell(z)$ is Lipschitz continuous with respect to z, according to Theorem 4 and Remark 8 in [13], if $\kappa = \infty$, the model (2) can be reformulated as

$$\min_{\boldsymbol{\beta}, \beta_0} h(\boldsymbol{\beta}) + \frac{1}{n}\sum_{i=1}^n \ell(z_i) + \lambda \cdot Lip(\ell)\|\boldsymbol{\beta}\|_q, \tag{3}$$

where q means the dual norm (i.e., $\frac{1}{p} + \frac{1}{q} = 1$), and $Lip(\ell)$ represents the Lipschitz modulus of ℓ.

2.2. Distribution robust regression models with piecewise linear loss function

The piecewise linear loss functions are very common in regression models, such as ϵ-insensitive loss, pinball loss and LAD loss, and they can be expressed as

$$\ell(z) = \max_{k=1,\cdots,K} \{a_k z + b_k\}.$$

where K is the number of pieces. In the general distributionally robust regression model (3), set without loss of generality $\max_k |a_k| = 1$ (thus $Lip(\ell) = 1$), $\lambda = \lambda_1$, $q = 1$, $h(\boldsymbol{\beta}) = \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2$, we can obtain the following regression model with piecewise linear loss functions:

$$\min_{\boldsymbol{\beta},\beta_0} \frac{1}{n}\sum_{i=1}^{n} \ell(z_i) + \lambda_1\|\boldsymbol{\beta}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2. \tag{4}$$

Although this model is a special case of the general distributionally robust regression, with different choice of piecewise linear loss, it still generalize many well-known regression models. Here we list two cases of them:

- *Support Vector Regression (SVR).* Taking $\ell(z) = \max\{0, |z| - \epsilon\}, \epsilon \geq 0$, model (4) reduces to the following SVR model with elastic net regularization term:

$$\min_{\boldsymbol{\beta},\beta_0} \frac{1}{n}\sum_{i=1}^{n} \max\{0, |z_i| - \epsilon\} + \lambda_1\|\boldsymbol{\beta}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2.$$

  For this regression model with only with ridge regularization term, Gunter et al. [18] and Wang et al. [20] developed solution path algorithms.
- *Quantile regression.* Taking $\ell(z) = \max\{-\tau z, (1-\tau)z\}$, $\tau \in [0, 1]$, model (4) reduces to the quantile regression model [22,23]:

$$\min_{\boldsymbol{\beta},\beta_0} \frac{1}{n}\sum_{i=1}^{n} \max\{-\tau z_i, (1-\tau)z_i\} + \lambda_1\|\boldsymbol{\beta}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2.$$

  For the hyperparameter τ, Takeuchi et al. [21] proposed a solution path algorithm to efficiently find the optimal solutions with respect to different values of τ. A regularization path algorithm for ℓ_1-norm regularized quantile regression was developed by Li and Zhu [24]. Li et al. [25] developed a solution path algorithm for the ℓ_2-norm regularized quantile regression.

Additionally, taking $\ell(z) = |z|$, model (4) reduces to the following model,

$$\min_{\boldsymbol{\beta},\beta_0} \frac{1}{n}\sum_{i=1}^{n} |z_i| + \lambda_1\|\boldsymbol{\beta}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2, \tag{5}$$

with the LAD loss. This *Doubly regularized Least Absolute Deviations regression (Dr-LAD)* model is robust to noise, and the elastic net regularization can perform feature selection and encourage grouping effect for correlated input features. For this newly developed regression model, it has two hyperparameters λ_1 and λ_2.

In the following Section 3, we will first prove that the optimal solutions of model (4) change piecewise linearly with respect to the hyperparameters. As a special case, the DrLAD model (5) will be further studied and a solution path algorithm is designed in Section 4.

3. Piecewise linear properties of the optimal solutions for distributionally robust regression model

For the distributionally robust regression model (4) with piecewise linear loss functions, it can be equivalently reformulated as follows,

$$\min_{\boldsymbol{\beta},\beta_0,\xi_i} \ \frac{1}{n}\sum_{i=1}^{n}\xi_i + \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2 \tag{6a}$$

$$s.t.\ a_k z_i + b_k \le \xi_i, \quad i=1,\dots,n, \quad k=1,\dots,K \tag{6b}$$

$$\sum_{j=1}^{d}\boldsymbol{\beta}_j \cdot sign(\boldsymbol{\beta}_j) \le s, \tag{6c}$$

$$\xi_i \ge 0, \quad i=1,\dots,n \tag{6d}$$

where the term $\lambda_1\|\boldsymbol{\beta}\|_1$ in the objective function is replaced by constraint (6c). Thus, for every λ_1, there exists a s such that model (4) and model (6) are equivalent. This model (6) is a convex quadratic program with linear constraints. There are two hyperparameters s and λ_2 in the model, and they control the trade-off between the loss function and regularization terms. The performance of the models is highly influenced by the choice the hyperparameters. If we directly perform cross-validation to identify proper values for these hyperparameters, this convex quadratic program should be solved repeatedly, leading to high computational cost.

To design the solution path algorithm, we first explore the piecewise linear properties of the optimal solutions to $\boldsymbol{\beta}, \beta_0$ in the equivalent model (6).

3.1. Piecewise linear solution path with respect to s

We first fix λ_2, and find the optimal solutions with respect to s.

Theorem 3.1. *If λ_2 is fixed, the optimal solutions $(\boldsymbol{\beta}^*, \beta_0^*)$ of distributionally robust regression model* (6) *are piecewise linear with respect to s.*

Proof. The Lagrangian function of (6) is

$$\begin{aligned} L(\boldsymbol{\beta},\beta_0,\xi;s,\lambda_2) =& \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i + \sum_{k=1}^{K}\sum_{i=1}^{n}\eta_i^k(a_k z_i + b_k - \xi_i) \\ &+ \theta(\sum_{j=1}^{d}\boldsymbol{\beta}_j \cdot sign(\boldsymbol{\beta}_j) - s) - \sum_{i=1}^{n}\gamma_i\xi_i, \end{aligned}$$

where η_i^k, γ_i $(i=1,\dots,n)$, θ are Lagrangian multipliers, and $z_i = y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0)$.

Define the following sets

$$\mathcal{R}_k = \{i : a_k z_i + b_k \neq 0\}, \quad \mathcal{E}_k = \{i : a_k z_i + b_k = 0\}, \quad \mathcal{V} = \{j : \boldsymbol{\beta}_j \neq 0\}.$$

Based on the generalized Karush–Kuhn–Tucker (KKT) theory [26], we can obtain

$$\frac{\partial L}{\partial \boldsymbol{\beta}_j} = \lambda_2 \boldsymbol{\beta}_j - \sum_{k=1}^{K} \sum_{i=1}^{n} a_k \eta_i^k \boldsymbol{x}_{ij} + \theta \cdot sign(\boldsymbol{\beta}_j) = 0, \quad j \in \mathcal{V} \tag{8a}$$

$$\frac{\partial L}{\partial \beta_0} = -\sum_{k=1}^{K} \sum_{i=1}^{n} a_k \eta_i^k = 0 \tag{8b}$$

$$\frac{\partial L}{\partial \xi_i} = \frac{1}{n} - \sum_{k=1}^{K} \eta_i^k - \gamma_i = 0, \quad i = 1, \ldots, n \tag{8c}$$

$$\eta_i^k (a_k z_i + b_k - \xi_i) = 0, \quad i = 1, \ldots, n, \quad k = 1, \ldots, K \tag{8d}$$

$$\theta(\sum_{j=1}^{d} \boldsymbol{\beta}_j \cdot sign(\boldsymbol{\beta}_j) - s) = 0 \tag{8e}$$

$$\gamma_i \xi_i = 0, \quad i = 1, \ldots, n \tag{8f}$$

For the points in $\mathcal{R}_k$ and $\mathcal{E}_k$, we can obtain

- For $i \in \mathcal{R}_k$: $\xi_i \neq 0$, $\gamma_i = 0$, $\sum_{k=1}^{K} \eta_i^k = \frac{1}{n}$, and
- For $i \in \mathcal{E}_k$: $\xi_i = 0$, $\sum_{k=1}^{K} \eta_i^k \leq \frac{1}{n}$.

For the points in the sets $\mathcal{V}, \mathcal{E}_k$, the corresponding constraints are active. Then we can obtain the following system of equations

$$\lambda_2 \boldsymbol{\beta}_j - \sum_{k=1}^{K} \sum_{i=1}^{n} a_k \eta_i^k \boldsymbol{x}_{ij} + \theta \cdot sign(\boldsymbol{\beta}_j) = 0, \quad j \in \mathcal{V} \tag{9a}$$

$$\sum_{k=1}^{K} \sum_{i=1}^{n} a_k \eta_i^k = 0, \tag{9b}$$

$$a_k(y_i - (\sum_{j \in \mathcal{V}} \boldsymbol{\beta}_j \boldsymbol{x}_{ij} + \beta_0)) + b_k = 0, \quad i \in \mathcal{E}_k, \quad k = 1, \ldots, K \tag{9c}$$

$$\sum_{j \in \mathcal{V}} \boldsymbol{\beta}_j \cdot sign(\boldsymbol{\beta}_j) = s, \text{ if } \theta > 0 \tag{9d}$$

In the above linear system, the number of equations and number of unknowns are equal. When s increases by a small enough amount, the sets $\mathcal{R}_k$, $\mathcal{E}_k$, $\mathcal{V}$ will not change, and the structure of the above system will not change. Taking derivatives with respect to

s, we have

$$\lambda_2 \frac{\Delta \boldsymbol{\beta}_j}{\Delta s} - \sum_{k=1}^{K} \sum_{i \in \mathcal{E}_k} \frac{\Delta \eta_i^k}{\Delta s} a_k \boldsymbol{x}_{ij} + \frac{\Delta \theta}{\Delta s} \cdot sign(\boldsymbol{\beta}_j) = 0, \quad j \in \mathcal{V} \tag{10a}$$

$$\sum_{k=1}^{K} \sum_{i \in \mathcal{E}_k} a_k \frac{\Delta \eta_i^k}{\Delta s} = 0, \tag{10b}$$

$$a_k (\sum_{j \in \mathcal{V}} \frac{\Delta \boldsymbol{\beta}_j}{\Delta s} \boldsymbol{x}_{ij} + \frac{\Delta \beta_0}{\Delta s}) = 0, \quad i \in \mathcal{E}_k, \quad k = 1, \dots, K \tag{10c}$$

$$\sum_{j \in \mathcal{V}} \frac{\Delta \boldsymbol{\beta}_j}{\Delta s} \cdot sign(\boldsymbol{\beta}_j) = 1, \text{ if } \theta > 0 \tag{10d}$$

The solutions of the above system of linear equations are the gradients of $\boldsymbol{\beta}_j, \beta_0, \eta_i^k, \theta$. If the active sets $\mathcal{V}$ and $\mathcal{E}_k$ do not change, the gradients will not change. So $\boldsymbol{\beta}_j, \beta_0, \eta_i^k, \theta$ will change linearly in s. When the s increases large enough, one of the active set $\mathcal{V}$ and $\mathcal{E}$ will change, and the structure of the above system will change. We will obtain new gradients, so the optimal solutions will change piecewise linear with respect to s. □

3.2. Piecewise linear solution path with respect to λ_2

Next, we fix s, and find the optimal solutions of $\boldsymbol{\beta}$ and β_0 with respect to λ_2.

Theorem 3.2. *If s is fixed, the optimal solutions $(\boldsymbol{\beta}^*, \beta_0^*)$ of distributionally robust regression model* (6) *are piecewise linear with respect to $1/\lambda_2$.*

Proof. Similar to (9), let $\nu = 1/\lambda_2$, $\zeta_i^k = \nu \eta_i^k$, we have

$$\boldsymbol{\beta}_j - \sum_{k=1}^{K} \sum_{i=1}^{n} a_k \zeta_i^k \boldsymbol{x}_{ij} + \theta \nu \cdot sign(\boldsymbol{\beta}_j) = 0, \quad j \in \mathcal{V} \tag{11a}$$

$$\sum_{k=1}^{K} \sum_{i=1}^{n} \zeta_i^k = 0, \tag{11b}$$

$$a_k (y_i - (\sum_{j \in \mathcal{V}} \boldsymbol{\beta}_j \boldsymbol{x}_{ij} + \beta_0)) + b_k = 0, \quad i \in \mathcal{E}_k, \quad k = 1, \dots, K \tag{11c}$$

In the above linear system, the number of equations and unknowns ($\theta = \lambda_1$ is known) are equal. When ν changes by a small enough amount, the sets $\mathcal{E}_k$, $\mathcal{V}$ will not change, and the structure of the above system will not change. The system of linear equations (11) can be written as

$$\boldsymbol{\beta}_j - \sum_{k=1}^{K} \sum_{i \in \mathcal{E}}^{n} a_k \zeta_i^k \boldsymbol{x}_{ij} - \sum_{k=1}^{K} \sum_{i \notin \mathcal{E}}^{n} \nu a_k \eta_i^k \boldsymbol{x}_{ij} + \theta \nu \cdot sign(\boldsymbol{\beta}_j) = 0, \quad j \in \mathcal{V} \tag{12a}$$

$$\sum_{k=1}^{K} \sum_{i \in \mathcal{E}_k}^{n} a_k \zeta_i^k + \sum_{k=1}^{K} \sum_{i \notin \mathcal{E}}^{n} a_k \zeta_i^k = 0, \tag{12b}$$

$$a_k(y_i - (\sum_{j\in\mathcal{V}} \boldsymbol{\beta}_j \boldsymbol{x}_{ij} + \beta_0)) + b_k = 0, \quad i \in \mathcal{E}_k, k = 1, \ldots, K \tag{12c}$$

Taking derivatives with respect to ν, we can obtain

$$\frac{\Delta\boldsymbol{\beta}_j}{\Delta\nu} - \sum_{k=1}^{K}\sum_{i\in\mathcal{E}_k} \frac{\Delta\zeta_i^k}{\Delta\nu} a_k \boldsymbol{x}_{ij} = \sum_{k=1}^{K}\sum_{i\notin\mathcal{E}_k} a_k\eta_i^k \boldsymbol{x}_{ij} - \theta\cdot sign(\boldsymbol{\beta}_j), \quad j \in \mathcal{V} \tag{13a}$$

$$\sum_{k=1}^{K}\sum_{i\in\mathcal{E}_k} a_k \frac{\Delta\zeta_i^k}{\Delta\nu} = -\sum_{k=1}^{K}\sum_{i\notin\mathcal{E}_k} a_k\eta_i^k = \sum_{k=1}^{K}\sum_{i\in\mathcal{E}_k} a_k\eta_i^k \tag{13b}$$

$$a_k(\sum_{j\in\mathcal{V}} \frac{\Delta\boldsymbol{\beta}_j}{\Delta\nu}\boldsymbol{x}_{ij} + \frac{\Delta\beta_0}{\Delta\nu}) = 0, \quad i \in \mathcal{E}_k, \quad k = 1, \ldots, K \tag{13c}$$

Similar as the proof for Theorem 3.1, we can find $\boldsymbol{\beta}$, β_0 are piecewise linear in ν. In equation (11), if we plugin $\boldsymbol{\beta}_j = \frac{1}{\lambda_2}\sum_{k=1}^{K}\sum_{i=1}^{n} a_k\eta_i^k\boldsymbol{x}_{ij} - \frac{\theta}{\lambda_2}\cdot sign(\boldsymbol{\beta}_j)$,

$$\sum_{k=1}^{K}\sum_{i=1}^{n} a_k\eta_i^k = 0,$$

$$a_k(y_i\lambda_2 - (\sum_{j\in\mathcal{V}}\sum_{k=1}^{K}(\sum_{i=1}^{n} a_k\eta_i^k\boldsymbol{x}_{ij} - \theta\cdot sign(\boldsymbol{\beta}_j))\boldsymbol{x}_{ij} + \eta_0)) + \lambda_2 b_k = 0, \quad i \in \mathcal{E}_k, k = 1, \ldots, K$$

where $\eta_0 = \lambda_2\beta_0$. Then taking derivative with respect to λ_2, we can obtain

$$\sum_{k=1}^{K}\sum_{i\in\mathcal{E}_k} a_k \frac{\Delta\eta_i^k}{\Delta\lambda_2} = 0, \tag{15a}$$

$$a_k(\sum_{j\in\mathcal{V}}(\sum_{k=1}^{K}\sum_{i\in\mathcal{E}_k} \frac{\Delta\eta_i^k}{\Delta\lambda_2} a_k\boldsymbol{x}_{ij})\boldsymbol{x}_{ij} + \frac{\Delta\eta_0}{\Delta\lambda_2}) = y_i + b_k, \quad i \in \mathcal{E}_k, k = 1, \ldots, K \tag{15b}$$

Similar as the proof for Theorem 3.1, we can find that η_i^k is piecewise linear with respect to λ_2. □

4. Solution path algorithm for the Doubly Regularized Least Absolute Deviations Regression

In Section 3, we have proved that the optimal solutions of model (4) change piecewise linearly with respect to hyperparameters $s, 1/\lambda_2$. Next, we need to identify the breakpoints between different pieces, where the structure of the linear systems (10), (13), (15) will change. For different loss functions, there are different ways to identify breakpoints.

In this section, we consider a special case of the distributionally robust regression model, the DrLAD model (shown in (5)). A solution path algorithm is proposed to track the optimal solution path of DrLAD with respect hyperparameters. In a similar way to the reformulation of model (6), model (5) can be equivalently written as

$$\min_{\boldsymbol{\beta},\beta_0,\xi} \ \frac{1}{n}\sum_{i=1}^{n}\xi_i + \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2 \tag{16a}$$

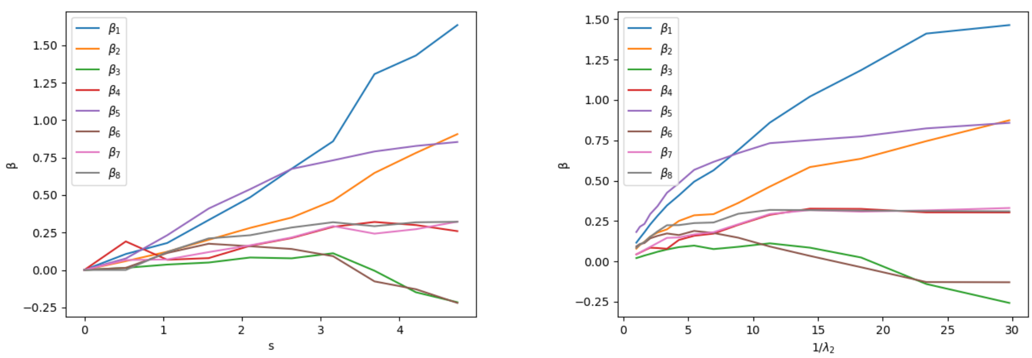
$$s.t. \ y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) \le \xi_i, \quad i = 1,\dots,n \tag{16b}$$

$$(\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) - y_i \le \xi_i, \quad i = 1,\dots,n \tag{16c}$$

$$\sum_{j=1}^{d} \boldsymbol{\beta}_j \cdot sign(\boldsymbol{\beta}_j) \le s \tag{16d}$$

$$\xi_i \ge 0, \quad i = 1,\dots,n \tag{16e}$$

With Theorem 3.1 and Theorem 3.2, the optimal solutions $\boldsymbol{\beta}, \beta_0$ are piecewise linear with respect to $s, 1/\lambda_2$. For example, Figure 1 is the piecewise linear solution path of DrLAD model in prostate dataset shown in Section 5.



(a) (b)

Figure 1. The piecewise linear solution path of coefficients with respect to different hyperparameters

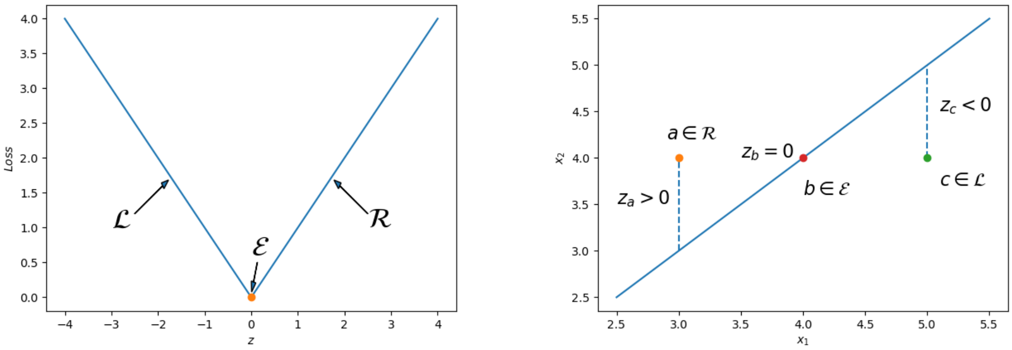
4.1. Piecewise linear solution path with respect to s

The Lagrangian function of (16) is

$$\begin{aligned} L(\boldsymbol{\beta}, \beta_0, \xi; s, \lambda_2) =& \frac{\lambda_2}{2}\|\boldsymbol{\beta}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\eta_i^+(y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) - \xi_i) \\ & - \sum_{i=1}^{n}\eta_i^-(y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) + \xi_i) + \theta(\sum_{j=1}^{d}\boldsymbol{\beta}_j \cdot sign(\boldsymbol{\beta}_j) - s) - \sum_{i=1}^{n}\gamma_i\xi_i, \end{aligned}$$

where η_i^+, η_i^-, γ_i $(i = 1,\dots,n)$ and θ are Lagrangian multipliers. Then, as shown in Figure 2, we can split the data points into different groups based on the residuals

$$\mathcal{E} = \{i : y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) = 0\}, \quad \mathcal{L} = \{i : y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) < 0\},$$
$$\mathcal{R} = \{i : y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0) > 0\}, \quad \mathcal{V} = \{j : \boldsymbol{\beta}_j \ne 0\},$$



(a) (b)

Figure 2. Partition of data points. $z = y - (\boldsymbol{\beta}^\intercal \boldsymbol{x} - \beta_0)$ is the residual. (a) If the residual $z < 0$, the point belongs to $\mathcal{L}$; If the residual $z = 0$, the point belongs to $\mathcal{E}$; If the residual $z > 0$, the point belongs to $\mathcal{R}$. (b) An example of the partition of data points.

where $\mathcal{E}$ contains points in the elbow of the LAD loss; $\mathcal{L}$ corresponds to points in the left of the elbow; $\mathcal{R}$ means points lie in the right of the elbow; and $\mathcal{V}$ contains the index of the nonzero feature. Then we have

- For $i \in \mathcal{R}$: $\xi_i > 0$, $\gamma_i = 0$, $\eta_i^+ = \frac{1}{n}$, $\eta_i^- = 0$;
- For $i \in \mathcal{L}$: $\xi_i > 0$, $\gamma_i = 0$, $\eta_i^+ = 0$, $\eta_i^- = \frac{1}{n}$;
- For $i \in \mathcal{E}$: $\xi_i = 0$, $\eta_i^+ + \eta_i^- \leq \frac{1}{n}$.

Let $\eta_i = \eta_i^+ - \eta_i^-$, according to equations (10), we have the following equations

$$\begin{aligned}
\lambda_2 \frac{\Delta \boldsymbol{\beta}_j}{\Delta s} - \sum_{i \in \mathcal{E}} \frac{\Delta \eta_i}{\Delta s} \boldsymbol{x}_{ij} + \frac{\Delta \theta}{\Delta s} \cdot sign(\boldsymbol{\beta}_j) &= 0, \quad j \in \mathcal{V} \\
\sum_{i \in \mathcal{E}} \frac{\Delta \eta_i}{\Delta s} &= 0, \\
\sum_{j \in \mathcal{V}} \frac{\Delta \boldsymbol{\beta}_j}{\Delta s} \boldsymbol{x}_{ij} + \frac{\Delta \beta_0}{\Delta s} &= 0, \quad i \in \mathcal{E} \\
\sum_{j \in \mathcal{V}} \frac{\Delta \boldsymbol{\beta}_j}{\Delta s} \cdot sign(\boldsymbol{\beta}_j) &= 1, \text{ if } \theta > 0
\end{aligned}$$

We define the following events to identify changes in $\mathcal{V}$, $\mathcal{L}$, $\mathcal{R}$ and $\mathcal{E}$:

- An active feature in $\mathcal{V}$ becomes inactive. This event happens when $|\boldsymbol{\beta}_j|$ becomes 0.
- An inactive feature joins active set $\mathcal{V}$. This event happens when the absolute value of the generalized correlation $c_j = \lambda_2 \boldsymbol{\beta}_j - \sum_{i=1}^n \eta_i \boldsymbol{x}_{ij}$ reaches θ.
- A point moves from $\mathcal{E}$ to $\mathcal{L}$ or $\mathcal{R}$. This event happens when η_i becomes $-\frac{1}{n}$ or $\frac{1}{n}$.
- A point moves from $\mathcal{L}$ or $\mathcal{R}$ to $\mathcal{E}$. This event happens when the residual $r_i = y_i - (\boldsymbol{\beta}^\intercal \boldsymbol{x}_i + \beta_0)$ reaches 0.

At the initial state, $s=0$, $\boldsymbol{\beta}=0$, and the objective function of (5) becomes

$$\min_{\beta_0} \frac{1}{n}\sum_{i=1}^{n}|y_i-\beta_0|$$

We have $\mathcal{V}=\emptyset$. Let $y_{(1)}, y_{(2)}, \ldots, y_{(n)}$ be the response variable in ascending order. If n is odd, let t_0 be the index corresponding to $y_{([n/2])}$, then we have $b=y_{t_0}$ and $\eta_{t_0}=0$. If n is even, let t_1 and t_2 be the corresponding indices of $y_{(n/2)}$ and $y_{(n/2+1)}$, then either one of them can be added to $\mathcal{E}$. If we take $b=y_{t_1}$, then we have $\eta_{t_1}=-\frac{1}{n}$.

Let δ^i be the smallest decrease for the first event i to happen. Next, we will detail exactly how to find each δ^i.

(1) An active feature in $\mathcal{V}$ becomes inactive:

$$\delta_s^1=\min_{j\in\mathcal{V}_+}\left\{\frac{0-\boldsymbol{\beta}_j}{\Delta\boldsymbol{\beta}_j/\Delta s}\right\}, \tag{19}$$

where $\mathcal{V}_+=\{j\in\mathcal{V}:\frac{0-\boldsymbol{\beta}_j}{\Delta\boldsymbol{\beta}_j/\Delta s}>0\}$.

(2) An inactive feature joins active set $\mathcal{V}$. Let $f_j^+=\frac{-\theta-c_j}{\Delta c_j/\Delta s+\Delta\theta/\Delta s}$, $f_j^-=\frac{\theta-c_j}{\Delta c_j/\Delta s-\Delta\theta/\Delta s}$, then we have

$$g_j=\begin{cases}\min\{f_j^+,f_j^-\}, \text{ if } f_j^+>0 \text{ and } f_j^->0\\ \max\{f_j^+,f_j^-\}, \text{ if either } f_j^+\le 0 \text{ or } f_j^-\le 0 \text{ but not both}\\ \infty, \text{ if } f_j^+\le 0 \text{ and } f_j^-\le 0\end{cases}$$

$$\delta_s^2=\min_{j\notin\mathcal{V}}\{g_j\} \tag{20}$$

where $\frac{\Delta c_j}{\Delta s}=-\sum_{i\in\mathcal{E}}\frac{\Delta\eta_i}{\Delta s}\boldsymbol{x}_{ij}$

(3) A point moves from $\mathcal{E}$ to $\mathcal{L}$ or $\mathcal{R}$:

$$\delta_s^3=\min_{i\in\mathcal{E}}\max\left\{\frac{-\frac{1}{n}-\eta_i}{\Delta\eta_i/\Delta s},\frac{\frac{1}{n}-\eta_i}{\Delta\eta_i/\Delta s}\right\} \tag{21}$$

(4) A point moves from $\mathcal{L}$ or $\mathcal{R}$ to $\mathcal{E}$:

$$\delta_s^4=\min_{i\in\mathcal{E}_+^c}\left\{\frac{0-r_i}{\Delta r_i/\Delta s}\right\}, \tag{22}$$

where $\mathcal{E}_+^c=\{i\notin\mathcal{E}:\frac{0-r_i}{\Delta r_i/\Delta s}>0\}$, $\frac{\Delta r_i}{\Delta s}=-\sum_{j\in\mathcal{V}}\frac{\Delta\boldsymbol{\beta}_j}{\Delta s}\boldsymbol{x}_{ij}-\frac{\Delta\beta_0}{\Delta s}$.

(5) the generalized correlation of active features reaches to 0:

$$\delta_s^5=\frac{0-\theta}{\Delta\theta/\Delta s} \tag{23}$$

The algorithm for the s-path is shown in Algorithm 1.

Algorithm 1 s-path algorithm

Input: Training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$

1: Initialize s, λ_2, β_0, $\boldsymbol{\beta}$, η_i, θ, $\mathcal{R}$, $\mathcal{L}$, $\mathcal{E}$, $\mathcal{V}$;
2: **while** $s \leq s_{max}$ **do**
3: Compute $\delta_s^1, \delta_s^2, \delta_s^3, \delta_s^4, \delta_s^5$ by equations (19)-(23);
4: Find $\delta_s = \min(\delta_s^1, \delta_s^2, \delta_s^3, \delta_s^4, \delta_s^5)$;
5: Update $\boldsymbol{\beta}_j, \beta_0, \eta_i, \theta$ by

$$\boldsymbol{\beta}_j \leftarrow \boldsymbol{\beta}_j + \delta_s \frac{\Delta \boldsymbol{\beta}_j}{\Delta s}, \quad j \in \mathcal{V}, \qquad \beta_0 \leftarrow \beta_0 + \delta_s \frac{\Delta \beta_0}{\Delta s},$$
$$\eta_i \leftarrow \eta_i + \delta_s \frac{\Delta \eta_i}{\Delta s}, \quad i \in \mathcal{E}, \qquad \theta \leftarrow \theta + \delta_s \frac{\Delta \theta}{\Delta s};$$

6: Update $\mathcal{L}, \mathcal{R}, \mathcal{E}$, and $\mathcal{V}$;
7: $s \leftarrow s + \delta_s$;
8: **end while**

Output: Solution path of $\boldsymbol{\beta}, \beta_0, \eta, \theta$ with respect to s

4.2. Piecewise linear solution path with respect to λ_2

As a special case, according to equations (13), we can obtain

$$\frac{\Delta \boldsymbol{\beta}_j}{\Delta \nu} - \sum_{i \in \mathcal{E}} \frac{\Delta \zeta_i}{\Delta \nu} \boldsymbol{x}_{ij} = \sum_{i \notin \mathcal{E}} \eta_i \boldsymbol{x}_{ij} - \theta \cdot sign(\boldsymbol{\beta}_j), \quad j \in \mathcal{V}$$
$$\sum_{i \in \mathcal{E}} \frac{\Delta \zeta_i}{\Delta \nu} = -\sum_{i \notin \mathcal{E}} \eta_i = \sum_{i \in \mathcal{E}} \eta_i$$
$$\sum_{j \in \mathcal{V}} \frac{\Delta \boldsymbol{\beta}_j}{\Delta \nu} \boldsymbol{x}_{ij} + \frac{\Delta \beta_0}{\Delta \nu} = 0, \quad i \in \mathcal{E}$$

We can find $\boldsymbol{\beta}$, β_0 and ζ_i are piecewise linear in ν. Then the following equations can be obtained based on equations (15)

$$\sum_{i \in \mathcal{E}} \frac{\Delta \eta_i}{\Delta \lambda_2} = 0,$$
$$\sum_{j \in \mathcal{V}} (\sum_{i \in \mathcal{E}} \frac{\Delta \eta_i}{\Delta \lambda_2} \boldsymbol{x}_{ij}) \boldsymbol{x}_{ij} + \frac{\Delta \eta_0}{\Delta \lambda_2} = y_i, \quad i \in \mathcal{E}$$

This linear system consists of $|\mathcal{E}|+1$ equations and $|\mathcal{E}|+1$ unknowns. We can find that η_i is piecewise linear in λ_2. Then we need to compute the increase of ν (or decrease of λ_2) needed to reach the next event:

(1) An active feature in $\mathcal{V}$ becomes inactive:

$$\delta_\nu^1 = \min_{j \in \mathcal{V}_+} \left\{ \frac{0 - \boldsymbol{\beta}_j}{\Delta \boldsymbol{\beta}_j / \Delta \nu} \right\}, \tag{26}$$

where $\mathcal{V}_+ = \{j \in \mathcal{V} : \frac{0-\boldsymbol{\beta}_j}{\Delta\boldsymbol{\beta}_j/\Delta\nu} > 0\}$.

(2) An inactive feature joins active set $\mathcal{V}$. Let $f_j^+ = \frac{-\theta - c_j}{\Delta c_j/\Delta\lambda_2}$, $f_j^- = \frac{\theta - c_j}{\Delta c_j/\Delta\lambda_2}$, then we have

$$g_j = \begin{cases} \max\{f_j^+, f_j^-\}, \text{ if } f_j^+ < 0 \text{ and } f_j^- < 0, \\ \min\{f_j^+, f_j^-\}, \text{ if either } f_j^+ \geq 0 \text{ or } f_j^- \geq 0 \text{ but not both}, \\ -\infty, \text{ if } f_j^+ \geq 0 \text{ and } f_j^- \geq 0, \end{cases}$$

$$\delta_{\lambda_2}^2 = \max_{j \notin \mathcal{V}} \{g_j\}, \tag{27}$$

where $\frac{\Delta c_j}{\Delta\lambda_2} = -\sum_{i=1}^n \frac{\Delta\eta_i}{\Delta\lambda_2}\boldsymbol{x}_{ij}$.

(3) A point moves from $\mathcal{E}$ to $\mathcal{L}$ or $\mathcal{R}$:

$$\delta_{\lambda_2}^3 = \max_{i \in \mathcal{E}} \min\left\{\frac{-\frac{1}{n} - \eta_i}{\Delta\eta_i/\Delta\lambda_2}, \frac{\frac{1}{n} - \eta_i}{\Delta\eta_i/\Delta\lambda_2}\right\}. \tag{28}$$

(4) A point moves from $\mathcal{L}$ or $\mathcal{R}$ to $\mathcal{E}$:

$$\delta_\nu^4 = \min_{i \in \mathcal{E}_+^c} \left\{\frac{0 - r_i}{\Delta r_i/\Delta\nu}\right\}, \tag{29}$$

where $\mathcal{E}_+^c = \{i \notin \mathcal{E} : \frac{0-r_i}{\Delta r_i/\Delta\nu} > 0\}$.

(5) λ_2 reaches to 0:

$$\delta_{\lambda_2}^5 = -\lambda_2. \tag{30}$$

Then find the event happens first by

$$\delta_{\lambda_2} = \max\left\{\frac{-\lambda_2^2\delta_\nu^1}{1+\lambda_2\delta_\nu^1}, \delta_{\lambda_2}^2, \delta_{\lambda_2}^3, \frac{-\lambda_2^2\delta_\nu^4}{1+\lambda_2\delta_\nu^4}, \delta_{\lambda_2}^5\right\} \tag{31}$$

The algorithm for the λ_2-path is shown in Algorithm 2.

Based on Algorithm 1 and Algorithm 2, the solution path algorithm for the DrLAD can be designed and is shown in Algorithm 3.

5. Numerical experiment

We implement multiple experiments to test the computational efficiency of solution path algorithms and the generalization performance of DrLAD model on different datasets. The datasets used in the experiments are from UCI machine learning repository [27], and the information of them are summarized in Table 1. The experiments are performed on a Linux server with AMD Zen2 16 core processors (2.3GHz) and 64GB RAM, with code by Python 3.8. The computational time is reported in second (s). In the experiments, we implement 5-fold cross-validation to select the appropriate hyperparameters, and repeat ten times.

Algorithm 2 λ_2-path algorithm

Input: Training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$, s

Initialize λ_2, β_0, $\boldsymbol{\beta}$, η_i, θ, $\mathcal{R}$, $\mathcal{L}$, $\mathcal{E}$, $\mathcal{V}$;
while $\lambda_2 \geq \lambda_2^{min}$ **do**
  Compute $\delta_\nu^1, \delta_{\lambda_2}^2, \delta_{\lambda_2}^3, \delta_\nu^4, \delta_{\lambda_2}^5$ by equations (26)-(30);
  Find δ_{λ_2} by equation (31);
  Update $\boldsymbol{\beta}_j, \beta_0, \eta_i$ by

$$\boldsymbol{\beta}_j \leftarrow \boldsymbol{\beta}_j + \delta_\nu \frac{\Delta \boldsymbol{\beta}_j}{\Delta \nu}, \quad j \in \mathcal{V}, \qquad \beta_0 \leftarrow \beta_0 + \delta_\nu \frac{\Delta \beta_0}{\Delta \nu}, \qquad \eta_i \leftarrow \eta_i + \delta_{\lambda_2} \frac{\Delta \eta_i}{\Delta \lambda_2}, i \in \mathcal{E};$$

  Update $\mathcal{L}, \mathcal{R}, \mathcal{E}$, and $\mathcal{V}$;
  Set $\lambda_2 \leftarrow \lambda_2 + \delta_{\lambda_2}$;
end while

Output: Solution path of $\boldsymbol{\beta}, \beta_0, \eta$ with respect to λ_2

Algorithm 3 Solution path algorithm for DrLAD

Input: Training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$

Initialize s, λ_2, β_0, $\boldsymbol{\beta}$, η_i, θ, $\mathcal{R}$, $\mathcal{L}$, $\mathcal{E}$, $\mathcal{V}$;
Set $\lambda_2 = 0$, Compute the solution path for s using Algorithm 1;
Set $s = 0$;
while $s \leq s^{max}$ **do**
  For given s, Compute the solution path with respect to λ_2 using Algorithm 2;
  Set $s \leftarrow s + \delta_s$;
end while

Output: Solution path of $\boldsymbol{\beta}, \beta_0, \eta, \theta$ with respect to s and λ_2

5.1. Computational efficiency

In this experiment, we first linearly sample the hyperparameter values from a given interval. The number of sampled hyperparameter values are 5, 10 and 20 (see Figure 3). Then, we implement 5-fold cross-validation on Prostate data and Triazines data to select the appropriate hyperparameters. The computational time and test error are recorded for solution path (SP) algorithm (shown in Algorithm 3), and also for CVXPY solver [28] as a comparison. The mean results and standard deviations are shown in Table 2 and Table 3.

In Table 2, as the number of hyperparameter values increases, the computational time will increase. The computational time of CVXPY method scales in total number of hyperparameter values. For example, when the number of values for s is 5, and the number of λ_2 is 5, the average computational time is 1.709s. When the number of values for λ_2 doubles to 10, the computational time also doubles to 3.884s. When the number of values for s is 10, and the number of values for λ_2 is 5, the computational time is 3.462s. The computational time in the fourth row is very close to the time in second row, because they have same total number of hyperparameter values.

The computational time of solution path algorithm mainly scales in number of s. When the number of values for λ_2 fixed 5, and the number of values for s increases from 5 to 10, the computational time increases from 0.482s to 0.781s. The time nearly doubles as the number of values for s doubles. But when the number of values for s

Table 1. Summary of datasets

Dataset	Number of samples	Dimensionality
Prostate	97	8
Wine quality	4898	12
Forest fire	517	12
Auto mpg	398	7
Triazines	186	60

Table 2. Experiments of DrLAD on Prostate data.

Hyperparameter		Computational time		Test Error	
# of s	# of λ_2	SP algorithm (s)	CVXPY(s)	SP algorithm	CVXPY
5	5	$\mathbf{0.482 \pm 0.078}$	1.709 ± 0.011	0.554 ± 0.204	$\mathbf{0.548 \pm 0.193}$
5	10	$\mathbf{0.497 \pm 0.104}$	3.884 ± 0.041	$\mathbf{0.537 \pm 0.198}$	0.547 ± 0.194
5	20	$\mathbf{0.522 \pm 0.116}$	8.129 ± 0.060	$\mathbf{0.531 \pm 0.195}$	0.550 ± 0.189
10	5	$\mathbf{0.781 \pm 0.185}$	3.462 ± 0.042	$\mathbf{0.547 \pm 0.204}$	0.556 ± 0.204
10	10	$\mathbf{0.873 \pm 0.227}$	7.774 ± 0.065	0.532 ± 0.187	$\mathbf{0.529 \pm 0.170}$
10	20	$\mathbf{0.976 \pm 0.269}$	16.371 ± 0.179	$\mathbf{0.526 \pm 0.188}$	0.538 ± 0.188
20	5	$\mathbf{1.452 \pm 0.316}$	6.905 ± 0.063	$\mathbf{0.538 \pm 0.185}$	0.550 ± 0.200
20	10	$\mathbf{1.666 \pm 0.414}$	15.611 ± 0.144	0.533 ± 0.186	$\mathbf{0.533 \pm 0.185}$
20	20	$\mathbf{1.873 \pm 0.495}$	32.884 ± 0.308	$\mathbf{0.530 \pm 0.185}$	0.545 ± 0.184

fixed 5, and the number of values for λ_2 increases from 5 to 10, the computational time only increases from 0.482s to 0.497s.

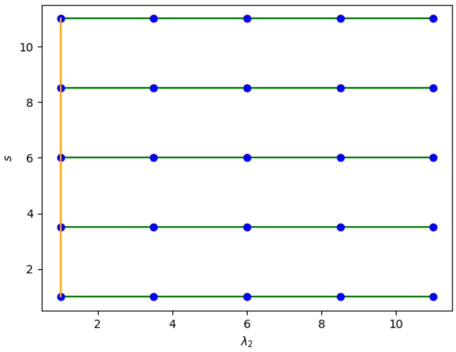
In Figure 3, the CVXPY method only computes the optimal solutions with respect to the blue points. So the computational time is scale in the total number of blue points. The solution path algorithm calculates all the optimal solutions along the solution path, corresponding to the green lines and orange lines. From Figure 3a to 3b, as the number of values for s increases, we need to calculate the new path for λ_2, which will lead to more computational cost. But from Figure 3a to 3c, although the number of values for λ_2 increases, the path remains the same as Figure 3a. The increased time is mainly caused by more iterations.

Compared with CVXPY method, solution path algorithm also achieves good test error, which means that it can select appropriate hyperparameters.

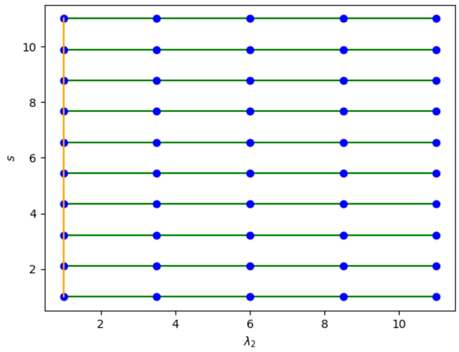
Table 3. Experiments of DrLAD on Triazines data.

Hyperparameter		Computational time		Test Error	
# of s	# of λ_2	SP algorithm (s)	CVXPY(s)	SP algorithm	CVXPY
5	5	$\mathbf{3.774 \pm 0.226}$	9.266 ± 0.166	$\mathbf{0.0226 \pm 0.0082}$	0.0229 ± 0.0078
5	10	$\mathbf{4.219 \pm 0.208}$	18.067 ± 0.345	$\mathbf{0.0212 \pm 0.0066}$	$\mathbf{0.0212 \pm 0.0066}$
5	20	$\mathbf{4.583 \pm 0.355}$	35.604 ± 0.674	$\mathbf{0.0214 \pm 0.0069}$	$\mathbf{0.0214 \pm 0.0069}$
10	5	$\mathbf{7.147 \pm 0.391}$	18.796 ± 0.343	$\mathbf{0.0200 \pm 0.0061}$	$\mathbf{0.0200 \pm 0.0061}$
10	10	$\mathbf{8.193 \pm 0.532}$	36.446 ± 0.638	$\mathbf{0.0202 \pm 0.0059}$	0.0204 ± 0.0057
10	20	$\mathbf{9.252 \pm 0.622}$	72.497 ± 1.184	$\mathbf{0.0197 \pm 0.0053}$	0.0202 ± 0.0058
20	5	$\mathbf{14.217 \pm 0.911}$	37.670 ± 0.554	$\mathbf{0.0202 \pm 0.0060}$	0.0203 ± 0.0060
20	10	$\mathbf{16.253 \pm 0.954}$	74.473 ± 1.543	$\mathbf{0.0201 \pm 0.0062}$	0.0203 ± 0.0060
20	20	$\mathbf{18.470 \pm 1.444}$	145.494 ± 2.045	$\mathbf{0.0196 \pm 0.0056}$	0.0201 ± 0.0061

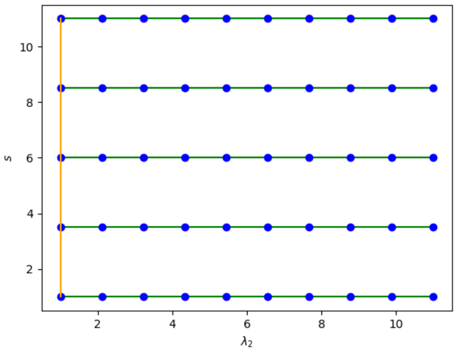
Table 3 shows the results for Triazines data. It also validates the efficiency of solution path algorithm. The triazines data has higher dimensions (dim = 60) than the prostate data (dim = 8). Regularization technique is widely used in high dimensional data analysis. The regularization term can perform feature selection and deal with correlated features. The solution path algorithm highly improves the efficiency



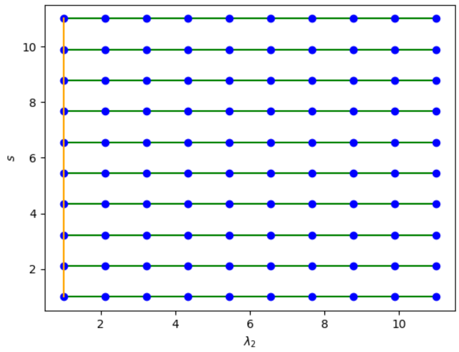
(a) Number of values for s is 5, Number of values for λ_2 is 5



(b) Number of values for s is 10, Number of values for λ_2 is 5



(c) Number of values for s is 5, Number of values for λ_2 is 10



(d) Number of values for s is 10, Number of values for λ_2 is 10

Figure 3. The grid of hyperparameters

of cross-validation process and selects the appropriate hyperparameters.

5.2. Test data performance

5.2.1. Simulation Experiments

In this experiment, we generate simulated data with group correlated coefficients under different correlations. The data are generated as follows:

$$\boldsymbol{x}_i \sim N_m(\mathbf{0}, \boldsymbol{\Sigma}),\ y_i \sim N_m(\boldsymbol{x}_i^\intercal \boldsymbol{\beta}^*, \sigma^2),$$

where the covariance matrix $\boldsymbol{\Sigma}$ is defined as

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}^1_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \boldsymbol{\Sigma}^2_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \boldsymbol{\Sigma}^3_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{I}_{21\times 21} \end{pmatrix}, \quad \boldsymbol{\Sigma}^1_{3\times 3} = \boldsymbol{\Sigma}^2_{3\times 3} = \boldsymbol{\Sigma}^3_{3\times 3} = \begin{pmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{pmatrix},$$

and ρ is the correlation of two variables. The Signal to Noise Ratio (SNR) is set to be 2, and the value of σ is calculated by $SNR = \frac{(\boldsymbol{\beta}^*)^\intercal \Sigma \boldsymbol{\beta}^*}{\sigma^2}$. The coefficients are defined as

$$\boldsymbol{\beta}^* = [1,1,1,0.5,0.5,0.5,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0].$$

We use the following evaluation metrics [29]:

- Mean Squared Error (MSE): $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i^\intercal \hat{\boldsymbol{\beta}})^2$.
- Relative Risk (RR): $RR(\hat{\boldsymbol{\beta}}) = \frac{\mathbb{E}(\boldsymbol{x}^\intercal\hat{\boldsymbol{\beta}} - \boldsymbol{x}^\intercal\boldsymbol{\beta}^*)^2}{\mathbb{E}(\boldsymbol{x}^\intercal\boldsymbol{\beta}^*)^2} = \frac{(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}^*)^\intercal\boldsymbol{\Sigma}(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}^*)}{(\boldsymbol{\beta}^*)^\intercal\boldsymbol{\Sigma}\boldsymbol{\beta}^*}$.
- Relative Test Error (RTE): $RTE(\hat{\boldsymbol{\beta}}) = \frac{\mathbb{E}(\mathbf{y}-\boldsymbol{x}^\intercal\hat{\boldsymbol{\beta}})^2}{\sigma^2} = \frac{(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}^*)^\intercal\boldsymbol{\Sigma}(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}^*)+\sigma^2}{\sigma^2}$.
- Proportion of Variance Explained (PVE): $PVE(\hat{\boldsymbol{\beta}}) = 1 - \frac{(\mathbf{y}-\boldsymbol{x}^\intercal\hat{\boldsymbol{\beta}})^2}{Var(\mathbf{y})} = 1 - \frac{(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}^*)^\intercal\boldsymbol{\Sigma}(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}^*)+\sigma^2}{(\boldsymbol{\beta}^*)^\intercal\boldsymbol{\Sigma}\boldsymbol{\beta}^*+\sigma^2}$.

Figure 4 shows that DrLAD achieves best performance in RR, RTE and PVE, which means that it provides the best explanation of the estimated coefficients. In Figure 4a, as the correlation increases, the estimation of coefficients becomes harder, and hence MSE has an increase trend for all the models. The LAD model has no regularization terms, and the generalization performance is not good. It has the highest MSE among the four models. DrLAD model achieves the lowest MSE or second lowest MSE for varied correlations. The elastic net regularization term is a combination of ℓ_1-norm regularization and ℓ_2-norm regularization. Thus it can perform feature selection like ℓ_1-norm regularization and deal with correlation like ℓ_2-norm regularization. It also encourages grouping effect to select a group of coefficients.

Figure 4b shows the result of RR, which is the ratio of expectation of squared error and expectation of true response. The score is 0 if the estimated coefficients equal true coefficients ($\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}^*$). It can be found that the RR score of LAD model is very high while DrLAD model has the lowest RR in varied correlations.

The RTE shown in Figure 4c is the ratio of expected squared error and Bayes error rate. If $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}^*$, the RTE score is 1. The DrLAD model obtains the lowest RTE score compared with other models.

The PVE is used to measure the proportion to which the model can explain the variation of data. Figure 4d shows that the DrLAD achieves highest PVE scores, which means that the coefficients estimated by DrLAD model account for the data better than the remaining models.

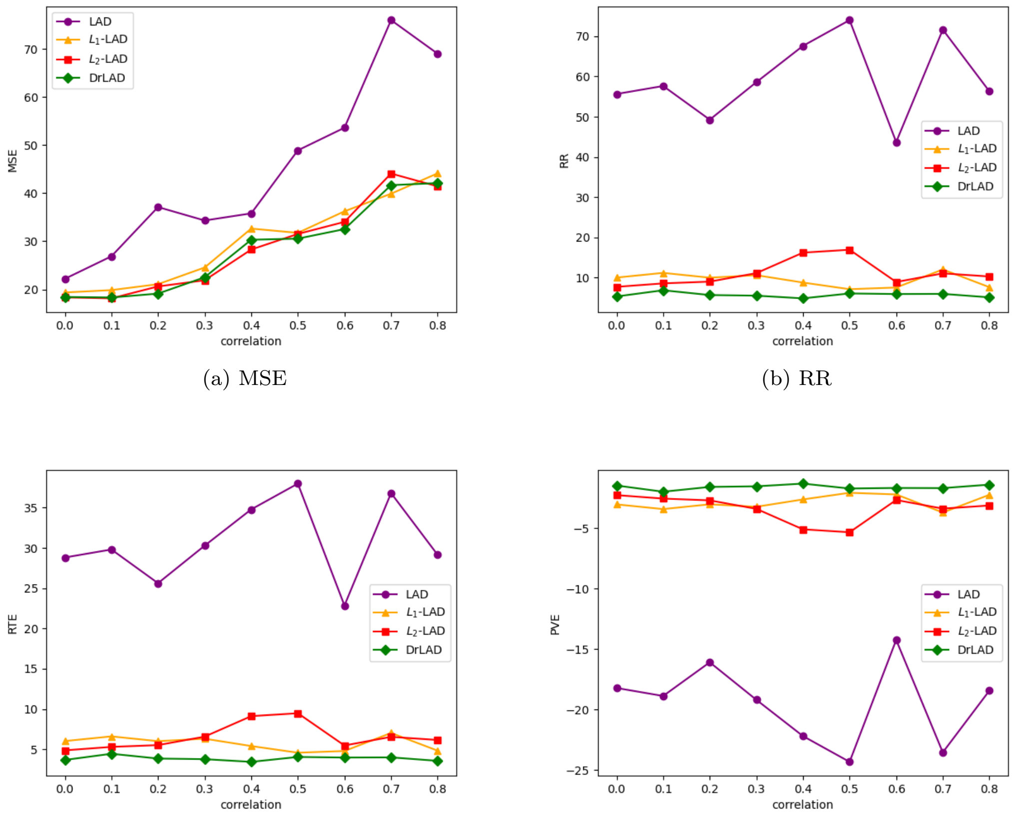
5.2.2. Real data experiments

In this experiment, we compare LAD, ℓ_1-LAD, ℓ_2-LAD and DrLAD models in different datasets. Each dataset is split into 80% for training and 20% for test. The 5-fold cross-validation is used for hyperparameter tuning.

Table 4. Mean Squred Error (MSE) of LAD, ℓ_1-LAD, ℓ_2-LAD and DrLAD Models

Dataset	LAD	ℓ_1-LAD	ℓ_2-LAD	DrLAD
Prostate	0.571 ± 0.210	0.568 ± 0.203	0.539 ± 0.189	$\mathbf{0.526 \pm 0.188}$
Wine quality	0.573 ± 0.025	0.570 ± 0.022	$\mathbf{0.569 \pm 0.022}$	$\mathbf{0.569 \pm 0.022}$
Forest fire	2.563 ± 0.363	2.469 ± 0.427	2.544 ± 0.358	$\mathbf{2.422 \pm 0.457}$
Auto MPG	12.560 ± 1.480	$\mathbf{12.481 \pm 1.460}$	12.548 ± 1.680	12.539 ± 1.680
Triazines	0.0294 ± 0.0084	0.0202 ± 0.0062	$\mathbf{0.0194 \pm 0.0045}$	0.0196 ± 0.0056

In Table 4, we can find that the regularized the LAD models have better general-



(a) MSE (b) RR

(c) RTE (d) PVE

Figure 4. Results of simulation experiments

ization performance compared with LAD model. It validates that the regularization term can improve the generalization performance. The ℓ_1-LAD model has the lowest MSE in Auto MPG data. The ℓ_2-LAD model has the best performance in Triazines data. The DrLAD model has the lowest MSE in Prostate, Wine quality, Forest fire and second lowest MSE in Auto MPG and Triazines. As the combination of ℓ_1 and ℓ_2 regularization, the elastic net regularization term enjoys the benefits from both ℓ_1 and ℓ_2 regularization, and has the better generalization performance in different datasets.

6. Conclusions

In this paper, we derive a general formulation for distributionally robust regression model based on distributionally robust optimization theory. This model generalizes many regression models, such as support vector regression, quantile regression. We also derive a new model DrLAD from this general framework. The optimal solutions of these models are piecewise linear with respect to hyperparameters, and a solution path algorithm is proposed to efficiently obtain the optimal solutions and speed up the hyperparameter tuning.

The experiments of computational time show that the solution path algorithm highly improves the cross-validation process compared to directly applying CVXPY solver. In the simulation experiments, we can observe that the regularization terms can deal with correlation, and the elastic net regularization term has the overall best performance. The real data experiments also show that the regularization terms can decrease the generalization error, and the DrLAD model has good performance in different datasets.

For future research, we can extend the solution path algorithm to distributionally robust regression models with more complex loss functions and regularization terms and to distributionally robust classification models.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

[1] Bloomfield P, Steiger WL. Least absolute deviations: theory, applications, and algorithms. Springer; 1983.

[2] Huber PJ. Robust Estimation of a Location Parameter. Ann Math Stat.1964;35(1):73 – 101.

[3] Huber PJ. Robust regression: asymptotics, conjectures and monte carlo. Ann Stat. 1973;1(5):799–821.

[4] Drucker H, Burges CJ, Kaufman L, et al. Support vector regression machines. Proceedings of Advances in neural information processing systems. 1996;9.

[5] Koenker R, Bassett Jr G. Regression quantiles. Econometrica. 1978;46(1):33–50.

[6] Hoerl AE, Kennard RW. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics. 1970;12(1):55–67.

[7] Tibshirani R. Regression shrinkage and selection via the lasso. J R Stat Soc Series B Stat Methodol. 1996;58(1):267–288.

[8] Zou H, Hastie T. Regularization and variable selection via the elastic net. J R Stat Soc Series B Stat Methodol. 2005;67(2):301–320.

[9] Mohajerin Esfahani P, Kuhn D. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. Math Program. 2018;171(1):115–166.

[10] Rahimian H, Mehrotra S. Distributionally robust optimization: A review. arXiv preprint arXiv:190805659. 2019.

[11] Shafieezadeh Abadeh S, Mohajerin Esfahani PM, Kuhn D. Distributionally robust logistic regression. Proceedings of Advances in Neural Information Processing Systems. 2015;28.

[12] Chen R, Paschalidis IC. A robust learning approach for regression models based on distributionally robust optimization. J Mach Learn Res. 2018;19(13):1–48.

[13] Shafieezadeh-Abadeh S, Kuhn D, Esfahani PM. Regularization via mass transportation. J Mach Learn Res. 2019;20(103):1–68.

[14] Tang G, Fan N. A survey of solution path algorithms for regression and classification models. Ann Data Sci. 2022;9(4):749–789.

[15] Efron B, Hastie T, Johnstone I, et al. Least angle regression. Ann Stat. 2004;32(2):407–499.

[16] Tibshirani RJ, Taylor J. The solution path of the generalized lasso. Ann Stat. 2011;39(3):1335–1371.

[17] Wang L, Gordon MD, Zhu J. Regularized least absolute deviations regression and an efficient algorithm for parameter tuning. Proceedings of 6th International Conference on Data Mining; IEEE; 2006. p. 690–700.

[18] Gunter L, Zhu J. Efficient computation and model selection for the support vector regression. Neural Comput. 2007;19(6):1633–1655.

[19] Wang G, Yeung DY, Lochovsky FH. A new solution path algorithm in support vector regression. IEEE Trans Neural Netw. 2008;19(10):1753–1767.
[20] Wang G, Yeung DY, Lochovsky FH. Two-dimensional solution path for support vector regression. Proceedings of the 23rd international conference on Machine learning; 2006. p. 993–1000.
[21] Takeuchi I, Nomura K, Kanamori T. Nonparametric conditional density estimation using piecewise-linear solution path of kernel quantile regression. Neural Comput. 2009;21(2):533–559.
[22] Takeuchi I, Le Q, Sears T, et al. Nonparametric quantile estimation. J Mach Learn Res. 2006. p. 1231–1264.
[23] Su M, Wang W. Elastic net penalized quantile regression model. J Comput Appl Math. 2021;392:113462.
[24] Li Y, Zhu J. L_1-norm quantile regression. J Comput Graph Stat. 2008; 17(1):163–185.
[25] Li Y, Liu Y, Zhu J. Quantile regression in reproducing kernel Hilbert spaces. J Am Stat Assoc. 2007;102(477):255–268.
[26] Hastie T, Tibshirani R, Wainwright M. Statistical learning with sparsity: the lasso and generalizations. CRC press; 2015.
[27] Kelly M, Longjohn R, Nottingham K. The UCI Machine Learning Repository, https://archive.ics.uci.edu.
[28] Diamond S, Boyd S. CVXPY: A Python-embedded modeling language for convex optimization. J Mach Learn Res. 2016;17(83):1–5.
[29] Hastie T, Tibshirani R, Tibshirani RJ. Extended comparisons of best subset selection, forward stepwise selection, and the lasso. arXiv preprint arXiv:170708692; 2017.