A User-Friendly Computational Framework for Robust Structured Regression with the L_2 Criterion

Jocelyn T. Chi and Eric C. Chi*

Abstract

We introduce a user-friendly computational framework for implementing robust versions of a wide variety of structured regression methods with the L_2 criterion. In addition to introducing an algorithm for performing L_2E regression, our framework enables robust regression with the L_2 criterion for additional structural constraints, works without requiring complex tuning procedures on the precision parameter, can be used to identify heterogeneous subpopulations, and can incorporate readily available non-robust structured regression solvers. We provide convergence guarantees for the framework and demonstrate its flexibility with some examples. Supplementary materials for this article are available online.

Keywords: block-relaxation, convex optimization, minimum distance estimation, regularization

*The work was supported in part by NSF grants DMS-2201136 and DMS-2103093, and NIH grant R01GM135928.

1 Introduction

Linear multiple regression is a classic method that is ubiquitous across numerous domains. Its ability to accurately quantify a linear relationship between a response vector $\mathbf{y} \in \mathbb{R}$ and a set of predictor variables $\mathbf{X} \in \mathbb{R}^{n \times p}$, however, is diminished in the presence of outliers. The L_2E method (Hjort, 1994; Scott, 2001, 2009; Terrell, 1990) presents an approach to robust linear regression that optimizes the well-known L_2 criterion from nonparametric density estimation in lieu of the maximum likelihood. Usage of the L_2E method for structured regression problems, however, has been limited by the lack of a simple computational framework. We introduce a general computational framework for performing a wide variety of robust structured regression methods with the L_2 criterion. Our work offers the following novel contributions.

1) Our framework extends the L_2E method from Scott (2001, 2009) to a wide variety of robust structured regression methods with the L_2 criterion.
2) Our framework enables simultaneous estimation of the regression coefficients and precision parameter (Section 3). We accomplish this via a block-coordinate descent algorithm. Therefore, our simultaneous estimation simplifies the process of choosing a parameter that tunes the robustness of the estimation procedure.
3) Our framework can "robustify" existing implementations of non-robust structured regression methods in a "plug-and-play" manner (Sections 3.3 and 4).
4) Our framework comes with convergence guarantees for the iterate sequence (Proposition 2).

Section 2 presents motivation for L_2 robust linear regression. Section 3 introduces our computational framework with convergence guarantees. Section 4 demonstrates the

simplicity and flexibility of our framework with robust implementations of several MLE-based methods via existing structured regression solvers. Section 5 provides a discussion.

1.1 Related Work

The L_2 minimization criterion has been employed in histogram bandwidth selection and kernel density estimators (Scott, 1992). Applying this well-known criterion from nonparametric density estimation to parametric estimation for regression problems enables a trade-off between efficiency and robustness. In fact, Basu et al. (1998) introduced a family of divergences that includes the L_2E as a special case and the MLE as a limiting case. The members of this family of divergences are indexed by a parameter that explicitly trades off efficiency for robustness. The MLE is the most efficient but also the least robust. Meanwhile, the L_2E offers a reasonable trade-off between efficiency and robustness (Warwick and Jones, 2005). The robustness of the L_2E can also be anticipated since it is a minimum distance estimator, which is known for robustness (Donoho et al., 1988).

Minimization of the L_2 criterion has been employed in developing robust statistical models including quantile regression (Lane, 2012), mixture models (Lee, 2010), classification (Chi and Scott, 2014), forecast aggregation (Ramos, 2014), and survival analysis (Yang and Scott, 2013). It also has uses in engineering applications including signal processing tasks such as wavelet-based image denoising (Scott, 2006) and image registration (Ma et al., 2015, 2013; Yang et al., 2017).

Some of the example methods we use to demonstrate our framework in Section 4 have robust implementations. These include robust multiple linear regression (Andrews, 1974; Audibert et al., 2011; Davies, 1993; Holland and Welsch, 1977; Meng and Mahoney, 2013), robust convex regression (Blanchet et al., 2019), robust isotonic regression (Álvarez

and Yohai, 2012; Lim, 2018), and robust sparse regression (Alfons et al., 2013; Chang et al., 2018; Ma et al., 2015; Nguyen and Tran, 2013; She and Owen, 2011; Yang et al., 2018). The purpose of our experiments is not to compare the L_2E to each of these robust methods. Rather, it is to demonstrate the flexibility and wide applicability of this computational framework and to show how it can obtain robust versions of existing non-robust implementations in lieu of case-by-case development of robust implementations.

Our framework's ability to simultaneously optimize over both the precision parameter and regression coefficients is a unique contribution to the literature. To highlight this, we briefly discuss two lines of prior work that are closely related to our proposed framework.

1.1.1 Minimum distance estimators for sparse regression and image registration

In the context of sparse regression, Wang et al. (2013) and Lozano et al. (2016) propose minimum distance estimators that coincide with our formulation when utilizing an ℓ_1-norm sparsity promoting regularizer. Lozano et al. (2016) employ a modification that applies a log transform on the empirical minimum distance criterion. The key difference between these prior approaches and our framework lies in obtaining the precision parameter. Wang et al. (2013) propose a hybrid block alternating scheme that estimates the regression coefficients by minimizing the L_2E criterion with the precision parameter fixed, and then the precision parameter is chosen to maximize efficiency subject to satisfying an asymptotic breakdown point of $\frac{1}{2}$. Their procedure alternates between these two steps and we refer to this approach as "hybrid" since the algorithm iterates are not minimizing a single objective function. Based on their simulation experiments, they conclude that their algorithm appears to converge within 1 to 3 steps but they lack a convergence proof. Lozano et al. (2016) treat

the precision parameter as a hyper-parameter that can be selected via cross-validation. For a fixed precision parameter, the Lozano et al. (2016) algorithm has algorithmic guarantees.

Both Wang et al. (2013) and Lozano et al. (2016) require pre-specifying a grid of values for the precision parameter. A fine grid enables finding a better precision parameter at the cost of more computational effort. In our work, we estimate the regression coefficients *and* precision parameter by solving an optimization problem. Like Wang et al. (2013), we also employ a block alternating algorithm but unlike their approach, our approach is not hybrid and is kept completely within an optimization framework, enabling algorithmic convergence guarantees (Proposition 2). Our strategy can also lead to better statistical performance in our simulation studies (Section 4). We anticipate this since our strategy enables exploring the joint space of regression coefficients and precision parameter more comprehensively. Our improved empirical performance comes without huge additional computational cost since the precision update involves solving a univariate optimization problem. This is a modest computational trade-off compared with searching over a grid of precision parameters.

In the context of image registration, Ma et al. (2015, 2013) and Yang et al. (2017) employ minimum distance estimation to robustly fit a linear model. The primary difference between their work and ours also lies in obtaining the precision parameter. They employ a deterministic annealing approach for choosing the precision parameter. Their algorithm solves an optimization problem to minimize the L_2E criterion with respect to the regression coefficients for a fixed precision parameter, and then decreases the precision parameter by a user-defined amount. Finally, they re-estimate the regression coefficients and alternate between updating the regression coefficient estimates and the precision parameter. Once again, a key question is whether the algorithm iterate sequence is guaranteed to converge.

1.1.2 Trimmed estimators for high dimensional regression

An alternative approach to obtaining robustness is to maximize a trimmed likelihood. Alfons et al. (2013) employ this for sparse robust multiple linear regression and estimate a sparse regression coefficient vector $\boldsymbol{\beta}$ by solving

$$\min_{\boldsymbol{\beta}} \frac{1}{2} \sum_{i=1}^{h} r_{[i]}(\boldsymbol{\beta})^2 + \lambda \|\boldsymbol{\beta}\|_1, \tag{1}$$

where $\mathbf{r}(\boldsymbol{\beta}) = \mathbf{y} - \mathbf{X}\boldsymbol{\beta}$ is a vector of residuals and $r_{[i]}(\boldsymbol{\beta})$ is the ith order statistic of $\mathbf{r}(\boldsymbol{\beta})$. The nonnegative parameter λ trades off model fit with sparsity in $\boldsymbol{\beta}$. The trimming hyper-parameter h imparts robustness to the standard residual sum of squares term by "trimming away" observations with large residuals. Yang et al. (2018) extend Alfons et al. (2013) to a general framework for robust penalized estimation similar to ours in the sense that they introduce a single framework for computing structured robust regression problems. The robustness of the estimator hinges on an appropriate h. Alfons et al. (2013) recommend employing prior knowledge to set h while Yang et al. (2018) utilize cross-validation.

The hyper-parameter h plays the same role as the precision parameter in the L_2E formulation. A first key difference between the approach in Yang et al. (2018) and ours is that we jointly estimate both the structured model and amount of trimming. This has three benefits. First, we reduce the potential for cross-validation to regularization parameters associated with the structure-incentivizing penalties, e.g. λ in (1). Second, our framework enables a continuous (and therefore, larger) search space for choosing the precision parameter, as opposed to pre-specifying a finite but potentially very large grid of trimming parameters for many observations. Third, our framework estimates both the regression coefficients and the precision parameter within an optimization framework, enabling convergence guarantees over the iterates.

A second key difference between the approach in Yang et al. (2018) and ours is that the precision parameter in our framework performs a "soft-trimming" action by adaptively choosing new down-weights for observations that are less consistent with the proposed model in each iteration. Rather than a single trim applied to all the observations, this enables additional flexibility for automatically varying the contribution of individual observations to the model fit. Section 4.4 demonstrates these advantages.

2 Robust regression with the L_2 criterion

Let f be the true but unknown density generating the observed data $y_1, \ldots, y_n \in \mathbb{R}$, and let $\hat{f}_{\boldsymbol{\theta}}$ be a probability density function indexed by a parameter $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^q$ approximating f. We assume throughout that all vectors are column vectors. If we were to estimate f using the $\hat{f}_{\boldsymbol{\theta}}$ that is closest to it, we could minimize the L_2 distance between f and $\hat{f}_{\boldsymbol{\theta}}$ in lieu of the negative log-likelihood with

$$\min_{\hat{\boldsymbol{\theta}} \in \Theta} \int \left[\hat{f}_{\boldsymbol{\theta}}(y) - f(y) \right]^2 dy. \tag{2}$$

In practice, however, identifying $\hat{\boldsymbol{\theta}}$ in this way is impossible since f remains unknown. While we typically cannot minimize the L_2 distance between f and its estimate $\hat{f}_{\boldsymbol{\theta}}$ directly, we *can* minimize an unbiased estimate of this distance. To do this, we first expand (2) as

$$\int \hat{f}_{\boldsymbol{\theta}}(y)^2 \, dy - 2 \int \hat{f}_{\boldsymbol{\theta}}(y) \, f(y) \, dy + \int f(y)^2 \, dy.$$

Notice that the second integral is the expectation $E_Y[\hat{f}_{\boldsymbol{\theta}}(Y)]$, where Y is a random variable drawn from f. Therefore, the sample mean provides an unbiased estimate of this quantity. Meanwhile, the third integral does not depend on $\boldsymbol{\theta}$. Therefore, we arrive at the the

following fully data-based loss function $h(\boldsymbol{\theta})$ that provides an unbiased estimate for (2) up to an irrelevant additive constant

$$h(\boldsymbol{\theta}) \;=\; \int \hat{f}_{\boldsymbol{\theta}}(y)^2 dy - \frac{2}{n}\sum_{i=1}^{n} \hat{f}_{\boldsymbol{\theta}}(y_i), \tag{3}$$

assuming $\hat{f}$ is square integrable over an appropriate region. Minimizing over this fully observed loss function presents us with our estimator $\hat{\boldsymbol{\theta}}$, also called an L_2E (Scott, 2001). Section 3.2 provides intuition for how the L_2E imparts robustness in our framework.

2.1 Regression model formulation

Let $\mathbf{y} \in \mathbb{R}$ denote a vector of n observed responses and let $\mathbf{X} \in \mathbb{R}^{n\times p}$ denote the corresponding observed design matrix of p-dimensional covariates. The standard linear model assumes the response and covariates are related via the model

$$\mathbf{y} \;=\; \mathbf{X}\boldsymbol{\beta}_0 + \tau_0^{-1}\boldsymbol{\varepsilon},$$

where $\boldsymbol{\beta}_0 \in \mathbb{R}^p$ is an unobserved vector of regression coefficients, $\tau_0 \in \mathbb{R}_+$ is an unobserved precision parameter, and the unobserved noise $\varepsilon_i \in \mathbb{R}$ for $1 \le i \le n$ are independently and identically distributed (iid) standard Gaussian random variables. We phrase the regression model in terms of the precision rather than the variance to obtain a more straightforward optimization problem later.

Let $\boldsymbol{\theta} = (\boldsymbol{\beta}^\mathsf{T}, \tau)^\mathsf{T}$ denote the vector of unknown parameters. Additionally, let $\mathbf{r}$ denote the residual vector obtained from the current prediction estimate for $\boldsymbol{\beta}$ so that its i^{th} component is $r_i = y_i - \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}$, where $\mathbf{x}_i \in \mathbb{R}^p$ contains the i^{th} row of $\mathbf{X}$. Given any suitable pair of $\boldsymbol{\beta}$ and τ, the conditional density of y_i for $1 \le i \le n$ is

$$\hat{f}^{(i)}_{\boldsymbol{\theta}}(y_i) \;=\; \frac{\tau}{\sqrt{2\pi}} \exp\left(-\frac{\tau^2}{2} r_i^2\right).$$

Following Scott (2001), we utilize the L_2E loss function for linear regression by averaging the L_2 distance over the observed data and minimize

$$h(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} h^{(i)}(\boldsymbol{\theta}) = \frac{\tau}{2\sqrt{\pi}} - \frac{\tau}{n}\sqrt{\frac{2}{\pi}}\sum_{i=1}^{n}\exp\left(-\frac{\tau^2}{2}r_i^2\right), \tag{4}$$

where

$$h^{(i)}(\boldsymbol{\theta}) = \int_{-\infty}^{\infty}\left[\hat{f}_{\boldsymbol{\theta}}^{(i)}(y_i)\right]^2 dy_i - 2\,\hat{f}_{\boldsymbol{\theta}}^{(i)}(y_i) = \frac{\tau}{2\sqrt{\pi}} - \tau\sqrt{\frac{2}{\pi}}\exp\left(-\frac{\tau^2}{2}r_i^2\right).$$

The solution $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}^{\mathsf{T}}, \hat{\tau})^{\mathsf{T}}$ of (4) contains the L_2E regression estimates.

3 Computational framework

We pose our estimation and model fitting task as a nonsmooth optimization problem. For references on optimization techniques employed in this paper, please refer to Lange (2010, 2013); Lange et al. (2014); Polson et al. (2015). Our computational framework for performing robust structured regression via the L_2 criterion is a general algorithm that combines the L_2E method (Scott, 2001, 2009) with a structural constraint or penalty term $\phi(\boldsymbol{\beta})$. As an example, suppose we wish to enforce a nonnegativity constraint on the regression coefficients $\boldsymbol{\beta}$. Then we can take $\phi(\boldsymbol{\beta}) = \iota_C(\boldsymbol{\beta})$, the indicator function of the nonnegative orthant $C = \{\boldsymbol{\beta} \in \mathbb{R}^p : \beta_j \geq 0, 1 \leq j \leq p\}$. Recall that the indicator function of a set C, denoted $\iota_C(\boldsymbol{\beta})$, is a function that takes values on the extended reals and is zero when $\boldsymbol{\beta} \in C$ and is ∞ otherwise. As another example, $\phi(\boldsymbol{\beta})$ may be an indicator function requiring that the elements of $\boldsymbol{\beta}$ satisfy a monotonicity constraint. Other examples include taking $\phi(\boldsymbol{\beta})$ to be sparsity inducing penalities like the ℓ_1-norm (Tibshirani, 1996) or elastic net (Zou and Hastie, 2005). Section 4 contains several examples of potential constraint

terms $\phi(\boldsymbol{\beta})$. Concretely, we seek a minimizer of the objective function

$$\ell(\boldsymbol{\beta}, \tau) \;\; = \;\; h(\boldsymbol{\beta}, \tau) + \phi(\boldsymbol{\beta}) \tag{5}$$

subject to $\boldsymbol{\beta} \in \mathbb{R}^p$ and $\tau \in [\tau_{\min}, \tau_{\max}]$, where $\tau_{\min} \in \mathbb{R}$ and $\tau_{\max} \in \mathbb{R}$ are minimum and maximum values for τ, respectively.

There are two computational challenges in minimizing (5). The first is that ℓ is non-convex in $\boldsymbol{\theta}$ since $h(\boldsymbol{\theta})$ is non-convex. The second is that commonly-used constraint terms $\phi(\boldsymbol{\beta})$ are often non-smooth or non-differentiable. We focus on the case where the ϕ are nonnegative, continuous, and convex functions. Continuity and convexity ensure that the proximal mappings of ϕ always exist and are unique. In minimizing (5), we utilize the key property that the block derivatives of h with respect to $\boldsymbol{\beta}$ and τ, that is $\nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau)$ and $\frac{\partial}{\partial \tau} h(\boldsymbol{\beta}, \tau)$, respectively, are Lipschitz differentiable.

Proposition 1. *The L_2E loss function $h(\boldsymbol{\beta}, \tau)$ is block Lipschitz differentiable with respect to $\boldsymbol{\beta}$ and τ so that*

$$\|\nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau) - \nabla_{\boldsymbol{\beta}} h(\tilde{\boldsymbol{\beta}}, \tau)\|_2 \;\; \leq \;\; L_\beta(\tau) \|\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}}\|_2$$

for all $\boldsymbol{\beta}$ and $\tilde{\boldsymbol{\beta}}$, and

$$\left| \frac{\partial}{\partial \tau} h(\boldsymbol{\beta}, \tau) - \frac{\partial}{\partial \tau} h(\boldsymbol{\beta}, \tilde{\tau}) \right| \;\; \leq \;\; L_\tau(\boldsymbol{\beta}) \, |\tau - \tilde{\tau}|$$

for all τ and $\tilde{\tau}$. The Lipschitz constant $L_\beta(\tau)$ is $L_\beta(\tau) = \frac{\tau^3}{n} \sqrt{\frac{2}{\pi}} \sigma(\mathbf{X})^2$, where $\sigma(\mathbf{X})$ is the largest singular value of $\mathbf{X}$. The Lipschitz constant $L_\tau(\boldsymbol{\beta})$ is $L_\tau(\boldsymbol{\beta}) = \frac{3}{n} \sqrt{\frac{2}{\pi}} \frac{\|\mathbf{r}\|_2^2}{\rho} \exp\left(-\frac{1}{2}\right)$, where $\rho = \min_{i: r_i \neq 0} |r_i|$.

The supplement contains the proof. The block Lipschitz differentiability of $h(\boldsymbol{\beta}, \tau)$ and the regularity conditions on ϕ lead us to employ block coordinate descent to minimize (5).

At a high level, we alternate between minimizing with respect to $\boldsymbol{\beta}$ holding τ fixed, and minimizing with respect to τ holding $\boldsymbol{\beta}$ fixed. We solve two subproblems at the k^{th} update:

Subproblem 1: Update $\boldsymbol{\beta}$

$$\boldsymbol{\beta}^{(k)} \quad = \quad \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min}\, h(\boldsymbol{\beta}, \tau^{(k-1)}) + \phi(\boldsymbol{\beta}), \text{ and} \tag{6}$$

Subproblem 2: Update τ

$$\tau^{(k)} \quad = \quad \underset{\tau \in [\tau_{\min}, \tau_{\max}]}{\arg\min}\, h(\boldsymbol{\beta}^{(k)}, \tau). \tag{7}$$

In practice, we do not exactly solve either subproblem and instead take a few proximal gradient descent steps to partially minimize or inexactly solve (6) and (7). Note that each update is guaranteed to monotonically decrease the loss function $\ell(\boldsymbol{\theta})$. This is a feature that all block coordinate descent algorithms possess as a special case of majorization-minimization algorithms (Lange, 2016).

Recall that proximal gradient descent is a first order iterative method for solving optimization problems of the form

$$\underset{\boldsymbol{\theta}}{\text{minimize}}\; h(\boldsymbol{\theta}) + \phi(\boldsymbol{\theta}), \tag{8}$$

where h is a Lipschitz differentiable function and ϕ is a convex and lower semicontinuous function (Combettes and Wajs, 2005; Parikh and Boyd, 2014). Further recall that the proximal map of ϕ

$$\text{prox}_{\phi}(\boldsymbol{\theta}) \quad = \quad \underset{\tilde{\boldsymbol{\theta}}}{\arg\min}\, \frac{1}{2}\|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_2^2 + \phi(\tilde{\boldsymbol{\theta}})$$

exists and is unique whenever $\phi(\boldsymbol{\theta})$ is convex and lower semicontinuous. Many regularizers $\phi(\boldsymbol{\beta})$ that are useful for recovering models with structure satisfy these conditions and

also admit proximal maps that can be evaluated via explicit formulation or an efficient algorithm. For example, the proximal map of the scaled ℓ_1-norm $\lambda\|\cdot\|_1$ is the element-wise soft-thresholding operator, namely

$$\left[\text{prox}_{\lambda\|\cdot\|_1}(\boldsymbol{\theta})\right]_i \;=\; \text{sign}(\theta_i)\max(|\theta_i| - \lambda, 0). \tag{9}$$

Notice that the proximal map can be viewed as the generalization of the Euclidean projection, which we refer to as the projection. Specifically, the projection of a point $\boldsymbol{\theta}$ onto a set C is the point $\mathcal{P}_C(\boldsymbol{\theta}) \in C$ that is closest in Euclidean distance to $\boldsymbol{\theta}$, namely

$$\mathcal{P}_C(\boldsymbol{\theta}) \;=\; \underset{\tilde{\boldsymbol{\theta}} \in C}{\arg\min}\, \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_2.$$

Similarly, the proximal map of the indicator function ι_C of a set C is the projection onto the set C. This projection exists and is unique when C is a closed convex set. For example, when $C = [\tau_{\min}, \tau_{\max}]$, $\text{prox}_{\iota_{[\tau_{\min},\tau_{\max}]}}(\tau) = \mathcal{P}_{[\tau_{\min},\tau_{\max}]}(\tau)$. As its name suggests, the proximal gradient descent method for solving problems of the form in (8) combines a gradient descent step with a proximal step. Given an iterate $\boldsymbol{\theta}$, the update $\boldsymbol{\theta}^+$ is

$$\boldsymbol{\theta}^+ \;=\; \text{prox}_{t\phi}[\boldsymbol{\theta} - t\nabla h(\boldsymbol{\theta})], \tag{10}$$

where t is a positive step-size parameter and $t\phi$ is the function ϕ scaled by t.

We emphasize that our framework does not require exactly computing the global minimizers in (6) and (7) at each iteration. Nonetheless, we will see that the algorithm still comes with some convergence guarantees.

Remark. We make the modestly stronger assumption that ϕ is continuous to establish convergence guarantees. Assuming continuity is not restrictive as commonly employed, convex nonsmooth ϕ includes norms, compositions of norms with linear mappings, and indicator functions of closed convex sets.

3.1 A general algorithm for L_2E robust structured regression

Algorithm 1 presents pseudocode for minimizing (5) via inexact block coordinate descent. For the update step on τ, the operator $\mathcal{P}_{[\tau_{\min},\tau_{\max}]}$ denotes the projection onto $[\tau_{\min}, \tau_{\max}]$. When updating $\boldsymbol{\beta}$ in (6) and τ in (7), we take a fixed number of proximal gradient steps, N_β and N_τ respectively, in (10). The gradients for updating $\boldsymbol{\beta}$ and τ are

$$\nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau) \;=\; -\frac{\tau^3}{n}\sqrt{\frac{2}{\pi}}\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{r}, \tag{11}$$

where $\mathbf{W} \in \mathbb{R}^{n\times n}$ is a diagonal matrix that depends on $\boldsymbol{\beta}$ with i^{th} diagonal entry

$$w_{ii} \;=\; \exp\left[-\frac{\tau^2}{2}r_i^2\right], \quad \text{and} \tag{12}$$

$$\frac{\partial}{\partial\tau}h(\boldsymbol{\beta}, \tau) \;=\; \frac{1}{2\sqrt{\pi}} - \frac{1}{n}\sqrt{\frac{2}{\pi}}\left[\sum_{i=1}^{n} w_{ii}\left(1 - \tau^2 r_i^2\right)\right].$$

Algorithm 1 has the following convergence guarantee. Recall that a point $\boldsymbol{\theta} = (\boldsymbol{\beta}^\mathsf{T}, \tau)^\mathsf{T}$ is a first order stationary point of a function $f(\boldsymbol{\theta})$ if for all directions $\mathbf{v}$, the directional derivative $f'(\boldsymbol{\theta}; \mathbf{v})$ of f is nonnegative.

Proposition 2. *For any choice of N_β and N_τ, under modest regularity conditions on (5) and step-sizes $t_\beta = L_\beta(\tau)^{-1}$ and $t_\tau = L_\tau^{-1}$, where $L_\beta(\tau)$ and $L_\tau(\boldsymbol{\beta})$ are in Proposition 1, the sequence $(\boldsymbol{\beta}^{(k)}, \tau^{(k)})$ generated by Algorithm 1 has at least one limit point and all limit points are first order stationary points of (5). If there are finitely many first order stationary points of (5), then the sequence $(\boldsymbol{\beta}^{(k)}, \tau^{(k)})$ will converge to one of them.*

The supplement contains a proof. We briefly comment on the assumption about the number of stationary points. It may seem strong to assume that the L_2E objective $\ell(\boldsymbol{\beta}, \tau)$ in

(5) has finitely many first order stationary points but a closer inspection of $h(\boldsymbol{\beta}, \tau)$ suggests that this is reasonable. The supplement contains an exploration of this assumption.

Algorithm 1 Block coordinate descent for minimizing (5)

Initialize $\boldsymbol{\beta}^{(0)}, \tau^{(0)}$ and fix N_β, N_τ

$k \leftarrow 0$
repeat
  $t_\beta \leftarrow L_\beta(\tau^{(k)})^{-1}$ // Update $L_\beta(\tau^{(k)})$ via Proposition 1
  $\boldsymbol{\beta} \leftarrow \boldsymbol{\beta}^{(k)}$ // Update $\boldsymbol{\beta}$ (6)
  for $i = 1, \ldots, N_\beta$ **do**
    $\boldsymbol{\beta} \leftarrow \text{prox}_{t_\beta \phi}\left[\boldsymbol{\beta} - t_\beta \nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau^{(k)})\right]$
  end for
  $\boldsymbol{\beta}^{(k+1)} \leftarrow \boldsymbol{\beta}$
  $t_\tau \leftarrow L_\tau(\boldsymbol{\beta}^{(k+1)})^{-1}$ // Update $L_\tau(\boldsymbol{\beta}^{(k+1)})$ via Proposition 1
  $\tau \leftarrow \tau^{(k)}$ // Update τ (7)
  for $i = 1, \ldots, N_\tau$ **do**
    $\tau \leftarrow \mathcal{P}_{[\tau_{\min}, \tau_{\max}]}\left[\tau - t_\tau \frac{\partial}{\partial \tau} h(\boldsymbol{\beta}^{(k+1)}, \tau)\right]$
  end for
  $\tau^{(k+1)} \leftarrow \tau$
  $k \leftarrow k + 1$
until convergence

3.2 Algorithmic intuition

We present a simple scenario illustrating intuition for Algorithm 1. This scenario applies to isotonic and convex regression, which we discuss in Section 4. Let the design matrix $\mathbf{X}$ be the $n \times n$ identity matrix $\mathbf{I}_n$, and let the structural constraint $\phi(\boldsymbol{\beta})$ be the indicator function of a closed and convex set C. Then $\phi(\boldsymbol{\beta}) = \iota_C(\boldsymbol{\beta})$ is zero if $\boldsymbol{\beta} \in C$, and is ∞

otherwise. This results in simplifications to (6) and the update rule for $\boldsymbol{\beta}$ becomes

$$\boldsymbol{\beta}^{+} \;=\; \mathcal{P}_{C}\left(\mathbf{z}\right),$$

where $\mathcal{P}_{C}\left(\mathbf{z}\right)$ is the Euclidean projection of $\mathbf{z} = \mathbf{W}\mathbf{y} + (\mathbf{I} - \mathbf{W})\boldsymbol{\beta}$ onto C, and $\mathbf{W}$ is a diagonal matrix with diagonal elements as defined in (12).

We observe how the L_2E imparts robustness through the action of $\mathbf{W}$. Consider $\mathbf{z} \in \mathbb{R}^n$ as a vector of pseudo-observations, where each element z_i is a convex combination of y_i and the current prediction β_i. If the current residual r_i is large compared to the current precision τ, w_i is small and the corresponding pseudo-observation z_i resembles the current predicted value β_i. Meanwhile, if the current residual r_i is small compared to the current precision τ, the corresponding pseudo-observation resembles the observed response y_i.

Therefore, the pseudo-observations impart the following algorithmic intuition. Given an estimate $\tilde{\boldsymbol{\beta}}$ of the regression coefficients, the algorithm performs constrained least squares regression using a pseudo-response $\mathbf{z}$, whose entries are a convex combination of the entries of the observed response $\mathbf{y}$ and the prediction $\tilde{\boldsymbol{\beta}}$. Observations with large current residuals relative to the current precision are essentially replaced by their predicted value.

In this way, the algorithm can fit a fraction of the observations very well while also accounting for outlying observations by replacing them with pseudo-response values more consistent with a model that fits the data. Notice that the algorithm is oblivious to whether large residuals arse from outliers in the response or in the predictor variables. Consequently, it can handle outliers arising from either source or both.

3.3 Robustifying existing non-robust implementations

We now discuss how one can employ this framework to automatically "robustify" existing non-robust structured regression implementations solving problems of the form

$$\min_{\boldsymbol{\beta}} \ \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \phi(\boldsymbol{\beta}).$$

Concretely, we can utilize existing non-robust solvers to perform line 6 in Algorithm 1. Recall that line 6 performs the $\boldsymbol{\beta}$ update with

$$\boldsymbol{\beta}^+ \ = \ \mathcal{P}_C\left(\mathbf{z}\right) \qquad \text{or} \qquad \boldsymbol{\beta}^+ \ = \ \text{prox}_{t_\beta \phi}(\mathbf{z})$$

depending on whether ϕ is a projection operator or a more general proximal mapping. In both cases, we perform this step by calling the existing non-robust solver and inputing $\mathbf{z}$ in place of the original response $\mathbf{y}$. Our computation for $\mathbf{z}$ depends on whether $\mathbf{X}$ is the identity. If $\mathbf{X}$ is the identity, as in isotonic and convex regression, then $\mathbf{z}$ in Algorithm 1 line 6 simplifies to $\mathbf{z} \ = \ \mathbf{W}\mathbf{y} + (\mathbf{I} - \mathbf{W})\boldsymbol{\beta}$ with $\mathbf{W}$ as described in (12). Therefore, Algorithm 1 line 6 inputs $\mathbf{z} \ = \ \mathbf{W}\mathbf{y} + (\mathbf{I} - \mathbf{W})\boldsymbol{\beta}$ in place of $\mathbf{y}$ into the existing non-robust solver.

If $\mathbf{X}$ is not the identity, as in Lasso regression, then $\mathbf{z}$ in Algorithm 1 line 6 is the more complex $\mathbf{z} = \boldsymbol{\beta} - t_\beta \nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau)$ with $\nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau)$ as described in (11). Recall that the $\boldsymbol{\beta}$ update involves a penalized least squares problem with identity design $\mathbf{I}$ (Section 3.2)

$$\boldsymbol{\beta}^+ \ \ = \ \ \text{prox}_{t_\beta \phi}(\mathbf{z}) \ \ = \ \ \underset{\tilde{\boldsymbol{\beta}}}{\text{minimize}} \ \frac{1}{2}\|\mathbf{z} - \mathbf{I}\tilde{\boldsymbol{\beta}}\|_2^2 + t_\beta \phi(\tilde{\boldsymbol{\beta}}).$$

Therefore, Algorithm 1 line 6 inputs $\mathbf{z} \ = \ \boldsymbol{\beta} - t_\beta \nabla_{\boldsymbol{\beta}} h(\boldsymbol{\beta}, \tau)$ in place of $\mathbf{y}$ and $\mathbf{I}$ in place of $\mathbf{X}$ into the existing non-robust solver.

3.4 Practical considerations

We discuss guidance on setting the hyperparameters in Algorithm 1. The constraint set $[\tau_{\min}, \tau_{\max}]$ on τ was introduced to establish the existence of a limit point for the algorithm iterate sequence. In practice, the constraints do not appear to strongly influence performance. Nonetheless, it is possible to run into a numerical issue if $\tau_{\min}$ is set to zero. Specifically, it is possible that the gradient step in the τ-update outputs a negative value, which would then be projected to 0. This results in $L_\beta(\tau)$ set to zero, which leads to an undefined step-size t_β. To guard against this, we recommend setting $\tau_{\min}$ as follows. A conservative estimate of the standard deviation follows from assuming no association between the response and covariates and all the variation in the response $\mathbf{y}$ is due to noise, namely $\hat{\sigma} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{y})^2}$. Therefore, take $\tau_{\min} = \hat{\sigma}^{-1}$. For the upper bound, taking $\tau_{\max}$ to be infinity does not appear to create any issues in practice.

A natural question is how to set N_β and N_τ in Algorithm 1. Choosing these values too small or too large can lead to slow convergence. In our experience, setting $N_\beta = N_\tau = 1$ does not make sufficient progress in minimizing the objective functions in (6) and (7). Meanwhile, setting N_β and N_τ to be a larger value such as 1,000 leads to diminishing returns in minimizing the objective functions in (6) and (7). In our experiments, we set N_β and N_τ to be 100 as it strikes a balance between these two extremes.

Finally, given the nonconvexity of the L_2E objective function in (5), some thought to initialization is required. We recommend the following simple "null model" initialization strategy. When we have a non-identity design matrix $\mathbf{X}$, similar to choosing $\tau_{\min}$, we assume there is no association between the response and covariates. Therefore, we set the initial regression coefficient vector $\boldsymbol{\beta}^{(0)} = \mathbf{0}$. When $\mathbf{X}$ is the identity, we set $\boldsymbol{\beta}^{(0)} = \overline{y}\mathbf{1}$, namely the vector of all ones $\mathbf{1}$ multiplied by the mean $\overline{y}$ of the response $\mathbf{y}$. In both cases,

we set the initial precision to be $\tau^{(0)} = \text{MAD}(\mathbf{y})^{-1}$, the reciprocal of the median absolute deviations of the response $\mathbf{y}$. We employ this initialization strategy throughout Section 4. The supplement contains a simulation study demonstrating that the output of Algorithm 1 appears stable to perturbations in this initialization heuristic.

4 Examples of L_2E robust structured regression

We demonstrate our framework on a variety of robust structured regression methods. Our examples illustrate how our framework can "robustify" existing structural regression solvers. We refer to the estimates obtained from optimizing the maximum likelihood and the L_2 criterion as the MLE and L_2E, respectively. Software for our framework is available in the `L2E` package on the Comprehensive R Archive Network (CRAN).

4.1 L_2E robust multiple linear regression

We begin with multivariate L_2E regression (Scott, 2001, 2009), where $\phi(\boldsymbol{\beta}) = 0$. Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ with $\text{rank}(\mathbf{X}) = p$. The data come from an Italian bank (Riani et al., 2014) where the response $\mathbf{y} \in \mathbb{R}$ is the annual investment earnings for $n = 1{,}949$ banking customers. The design $\mathbf{X}$ contains measurements on $p = 13$ bank services.

Since $\phi(\boldsymbol{\beta}) = 0$, $\text{prox}_{t_\beta \phi}$ is simply the identity operation. Subproblem 1 for updating $\boldsymbol{\beta}$ in (6) reduces to iteratively performing: 1) Compute current residuals, 2) Update weights w_{ii} in (12), and 3) Update $\boldsymbol{\beta}$ with current residuals and gradient described in Section 3.1.

Figures 1a and 1b depict scatter plots of the fitted values against the residuals. A good fit is evidenced by normally distributed noise in the residuals, or symmetric scatter of points about the zero residual level (depicted by the orange dashed line). Figure 1a shows

a discernible pattern in the MLE residuals with asymmetric scatter of points about the zero residual level. This indicates that additional trends in the data not captured by the Gaussian linear model remain in the residuals and are not captured by the MLE fit.

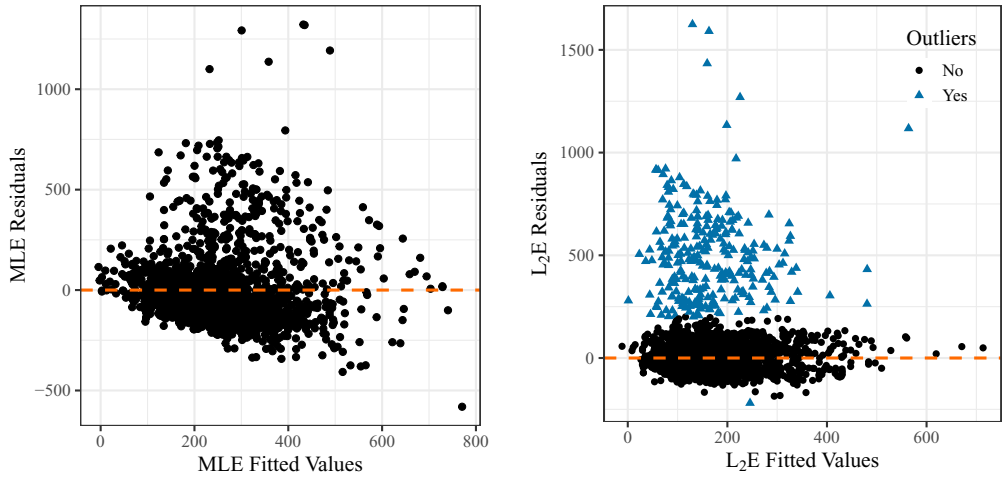
Meanwhile, Figure 1b shows that after excluding outlying points identified by automatic tuning of τ in our framework (depicted by the blue triangles), the residuals from the L_2E fit are normally distributed about zero. To identify outliers, we compute the L_2E residuals and select observations whose residuals exceed a factor of the precision parameter, e.g. 3 divided by τ. Thus, the L_2E adequately captures the linear relationship between investment earnings and bank services for the non-outlying customers. Notice that one can recursively repeat L_2E regression on outlying customers to identify an appropriate linear relationship between investment earnings and bank services for customer subgroups.

This example also highlights how our framework enables joint estimation of the regression coefficient vector $\boldsymbol{\beta}$ and the precision τ, enabling automatic identification of outlying observations in the data. This is practically useful since the L_2E can simultaneously identify subpopulations within the data and appropriate fits for each of those groups when applied recursively to the subgroups.

4.2 L_2E robust isotonic regression

Our next example is L_2E robust isotonic regression. Let an observed response $\mathbf{y} \in \mathbb{R}^n$ consist of n samples drawn from a monotonic function f sampled at discrete time points $t_1 \leq t_2 \leq \cdots \leq t_n$ with additive independent Gaussian noise. The i^{th} entry of $\mathbf{y}$ is

$$y_i = f(t_i) + \epsilon_i \quad \text{for} \quad 1 \leq i \leq n,$$



(a) MLE fitted values vs. residuals. (b) L_2E fitted values vs. residuals.

Figure 1: *(a) Asymmetric spread of points about the zero residual line suggests additional variation in the data not captured by the MLE linear fit. (b) Blue triangles denote outlying observations identified by the L_2E. Non-outlying observations are well-fit by the L_2E linear fit, as seen in normal distribution of points about the zero residual line.*

where f is monotonic, $\epsilon_i \overset{\text{iid}}{\sim} N(0, \frac{1}{\tau})$, and $\tau \in \mathbb{R}_+$. The goal of isotonic regression (Barlow and Brunk, 1972; Brunk et al., 1972; Dykstra et al., 1982; Lee et al., 1981; Mair et al., 2009) is to estimate f by solving

$$\min_{\beta(t_1),\dots,\beta(t_n)} \sum_{i=1}^{n} [y_i - \beta(t_i)]^2$$
$$\text{subject to} \quad \beta(t_1) \leq \beta(t_2) \leq \cdots \leq \beta(t_n).$$

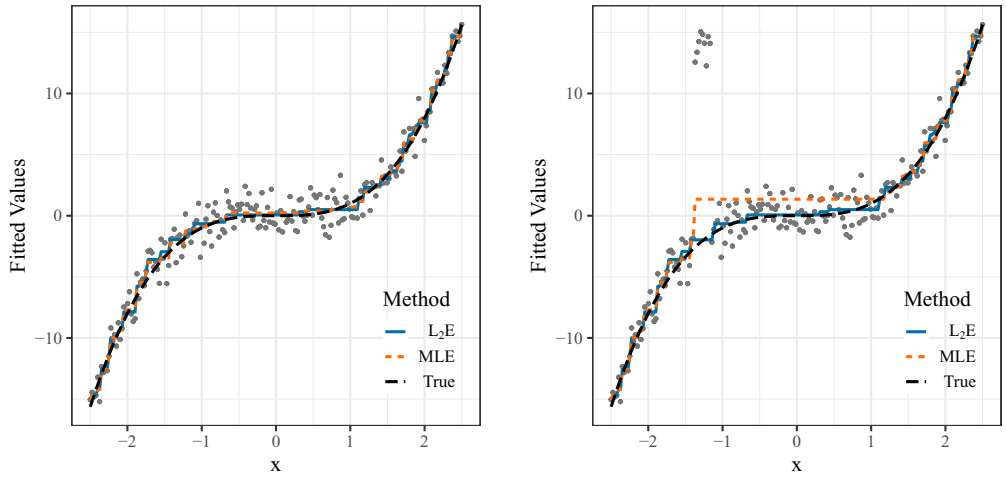
We construct a piece-wise constant estimate for f using the elements of the estimator $\hat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\beta}(t_1) & \hat{\beta}(t_2) & \cdots & \hat{\beta}(t_n) \end{pmatrix}^{\mathsf{T}}$.

For the corresponding L_2E problem, the design $\mathbf{X} = \mathbf{I}_n$ and $\phi(\boldsymbol{\beta}) = \iota_{\mathcal{M}}(\boldsymbol{\beta})$ is the indicator function over the set of vectors $\mathcal{M}$ satisfying element-wise monotonicity so that

$\beta_1 \leq \beta_2 \leq \cdots \leq \beta_n$ for $\boldsymbol{\beta} \in \mathbb{R}^n$. Subproblem 1 for updating $\boldsymbol{\beta}$ in (6) reduces to iteratively performing: 1) Compute current residuals, 2) Update weights w_{ii} in (12), and 3) Update $\boldsymbol{\beta}$ with current residuals and gradient described in Section 3.1 and project onto the set $\mathcal{M}$. The `gpava` function for implementing the generalized pool-adjacent-violators algorithm (generalized PAVA) in the `isotone` package (Mair et al., 2009) for R performs this last step. Therefore, we utilize it in Algorithm 1 line 6.

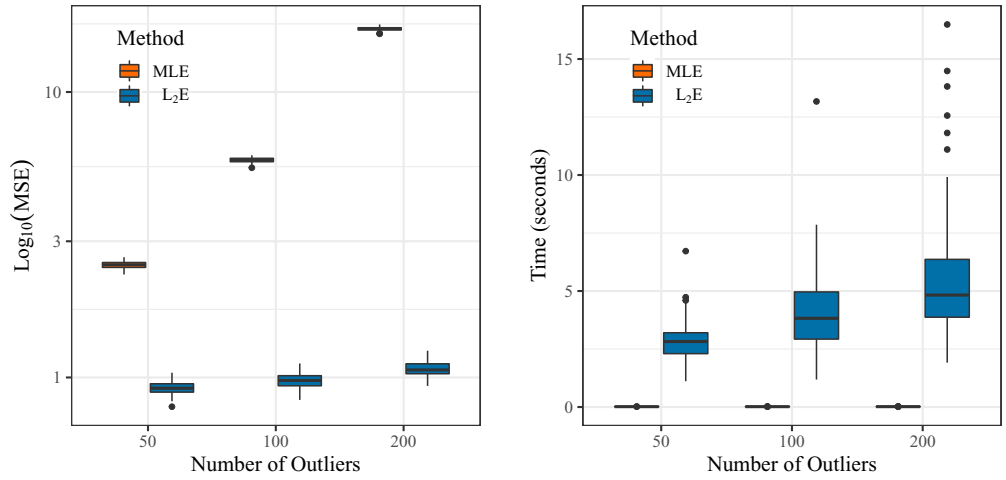
We illustrate with a univariate cubic function. Figure 2a shows how the MLE and L_2E produce similar estimates in the absence of outliers. The true underlying cubit fit f is in black. The gray points depict observations generated from f with additive Gaussian noise. The dashed orange line depicts the MLE from generalized PAVA while the solid blue line depicts the L_2E . Meanwhile, Figure 2b shows how the MLE is skewed towards the outliers while the L_2E estimate remains less sensitive to them.

Figure 3 depicts results of Monte Carlo simulations comparing the MLE and the L_2E while varying the number of outliers. We simulate three datasets with n = 1,000 observations of a cubic function with additive Gaussian noise and $50, 100$, and 200 outliers, respectively. We introduce outliers by selecting points from approximately the 25^{th} quartile along the x-axis and assigning them a value equal to slightly less than the maximal polynomial value and additive standard Gaussian noise. This corresponds to simulating samples from a bimodal distribution to create high leverage points in the covariate space. We employ the `gpava` function in the `isotone` package (Mair et al., 2009) for R to obtain the MLE. We obtain 100 replicates for each scenario on a 3.00 GHz Intel Core i7 computer with 32 GB of RAM and present boxplots of the mean squared error (MSE) and time in seconds. We obtain the MSE between the model $\mathbf{y}$ and the computed solution.



(a) Isotonic regression without outliers. (b) Isotonic regression with outliers.

Figure 2: *The black, orange, and blue lines depict the true, MLE, and L_2E fits for isotonic regression, respectively. The MLE and L_2E produce similar results in the absence of outliers. The MLE is skewed towards the outliers while the L_2E provides a more robust estimate.*



(a) Isotonic regression MSE. (b) Isotonic regression time (sec).

Figure 3: *Monte Carlo experiments for isotonic regression with $n = 1{,}000$ observations drawn from a univariate cubic function with additive Gaussian noise. Boxplots depict the (a) mean squared error (MSE) and (b) time in seconds required for $50, 100,$ and 200 outliers over 100 replicates for the MLE (orange) and L_2E (blue). The L_2E requires more time since its solution requires multiple computations of the MLE. Experiments highlight the trade-off between MSE and time between the MLE and L_2E solutions.*

The MLE produces increasingly larger MSE as the number of outliers increases. Meanwhile, the L_2E produces a smaller increase in MSE for the same number of outliers but requires more computation time since the L_2E employs multiple computations of the MLE procedure. Thus, the L_2E can produce an isotonic regression fit that is much less sensitive to outliers than the MLE.

4.3 L_2E robust convex regression

Our next example is L_2E robust convex regression. For illustration, we consider the univariate case (Ghosal and Sen, 2017; Wang and Ghosh, 2012). However, our framework applies to multivariate convex regression (Aybat and Wang, 2016; Bertsimas and Mundru, 2021; Birke and Dette, 2007; Chen and Mazumder, 2021; Guntuboyina and Sen, 2015; Hannah and Dunson, 2013; Lim and Glynn, 2012; Lin et al., 2020; Mazumder et al., 2019; Meyer, 2003; Seijo and Sen, 2011) in a similar manner. Let an observed response $\mathbf{y} \in \mathbb{R}^n$ consist of n samples drawn from a convex function f sampled at discrete time points $t_1 \leq t_2 \leq \cdots \leq t_n$ with additive independent Gaussian noise. The i^{th} entry of $\mathbf{y}$ is

$$y_i = f(t_i) + \epsilon_i \quad \text{for} \quad 1 \leq i \leq n,$$

for convex f, $\epsilon_i \overset{\text{iid}}{\sim} N(0, \frac{1}{\tau})$, and $\tau \in \mathbb{R}_+$. Shape-restricted convex regression estimates f via

$$\begin{aligned} &\min_{\beta(t_1),\ldots,\beta(t_n)} \sum_{i=1}^{n} [y_i - \beta(t_i)]^2 \\ &\text{subject to} \quad \beta(t_i) \leq \frac{t_{i+1} - t_i}{t_{i+1} - t_{i-1}} \beta(t_{i-1}) + \frac{t_i - t_{i-1}}{t_{i+1} - t_{i-1}} \beta(t_{i+1}) \quad \text{for} \quad 2 \leq i \leq n-1. \end{aligned}$$

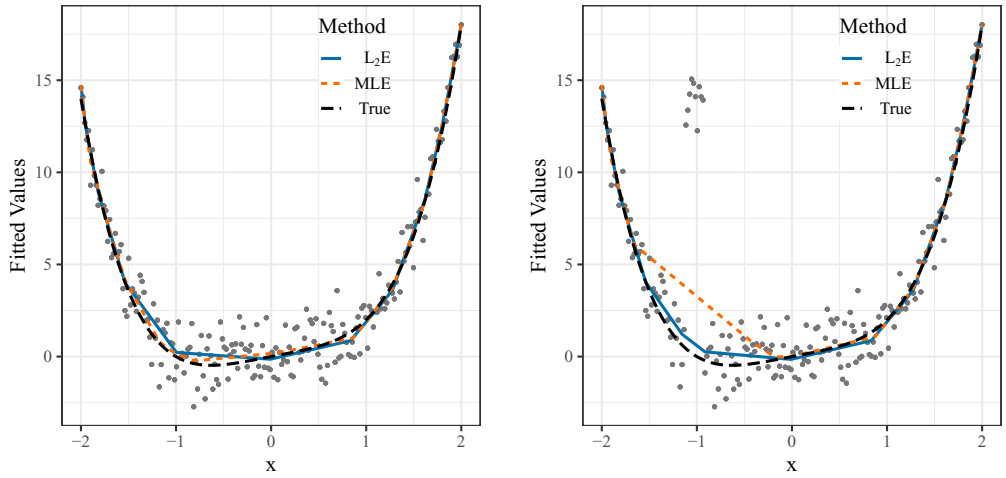
Let $\boldsymbol{\beta} = \begin{pmatrix} \beta(t_1) & \beta(t_2) & \cdots & \beta(t_n) \end{pmatrix}^\mathsf{T}$. We recast this constraint in terms of a scaled second-order differencing matrix $\mathbf{D} \in \mathbb{R}^{n \times n}$ with $\mathbf{D}\boldsymbol{\beta} \geq \mathbf{0}$ so that all the elements of $\mathbf{D}\boldsymbol{\beta}$ are

non-negative. We construct a piece-wise constant estimate for f utilizing the elements of $\hat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\beta}(t_1) & \hat{\beta}(t_2) & \cdots & \hat{\beta}(t_n) \end{pmatrix}^{\mathsf{T}}$.

For the corresponding L_2E problem, the design $\mathbf{X} = \mathbf{I}_n$ and $\phi(\boldsymbol{\beta}) = \iota_{\mathcal{C}}(\boldsymbol{\beta})$ is the indicator function over the set of vectors in $\mathcal{C} \equiv \{\boldsymbol{\beta} : \mathbf{D}\boldsymbol{\beta} \geq \mathbf{0}\}$. Subproblem 1 for updating $\boldsymbol{\beta}$ in (6) reduces to iteratively performing: 1) Compute current residuals, 2) Update weights w_{ii} in (12), and 3) Update $\boldsymbol{\beta}$ with current residuals and gradient described in Section 3.1 and project onto the convex cone $\mathcal{C}$. The `conreg` function in the `cobs` package (Ng and Maechler, 2007) for R performs this last step so we can employ it in Algorithm 1 line 6.

Figure 4a shows how the MLE and L_2E produce similar fits in the absence of outliers. The true underlying convex fit f is in black and gray points depict observations generated from f with additive Gaussian noise. The dashed orange line depicts the MLE obtained from the `cobs` package in R while the solid blue line depicts the L_2E . Meanwhile, Figure 4b shows how the MLE is substantially skewed towards the outliers while the L_2E is less distorted. This example highlights how the L_2E is less sensitive to outliers than the MLE.

Figure 5 depicts results of Monte Carlo simulations comparing the MLE and L_2E on shape-restricted convex regression while varying the number of outliers. We simulate three datasets with $n = 1{,}000$ observations using a fourth-order polynomial with additive Gaussian noise and $50, 100$, and 200 outliers, respectively. We introduce outliers by selecting points from approximately the 25^{th} quartile along the x-axis and assigning them a value that is equal to a little less than the maximal polynomial value and additive standard Gaussian noise. This corresponds to simulating samples from a bimodal distribution to create high leverage points in the covariate space. We employ the `conreg` function in the `cobs` package for R (Ng and Maechler, 2007) to obtain the MLE. We obtain 100 replicates on a 3.00 GHz Intel Core i7 computer with 32 GB of RAM.



(a) Convex regression without outliers. (b) Convex regression with outliers.

Figure 4: *Black, orange, and blue lines depict the true, MLE, and L_2E fits for convex regression. The MLE and L_2E produce similar results in the absence of outliers while the L_2E is much less sensitive to outliers.*

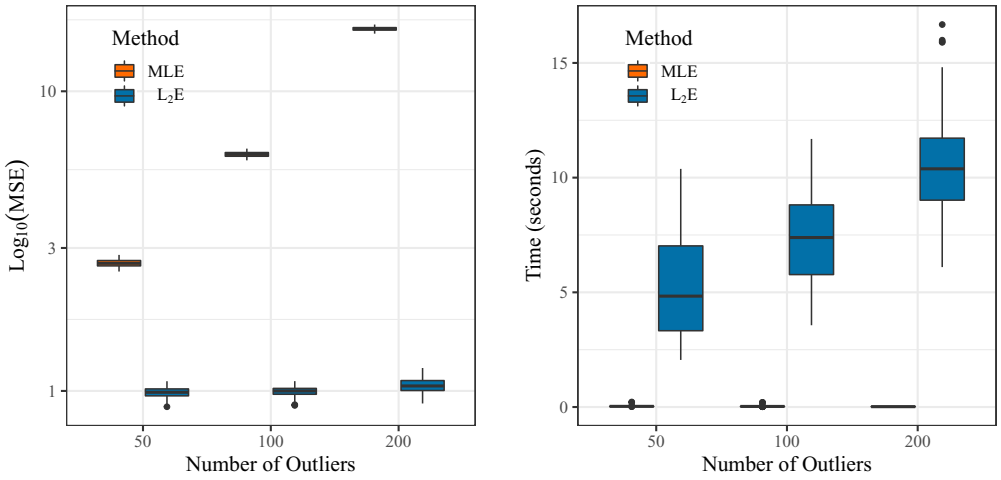
Figure 5a highlights how the MLE produces increasingly larger MSE values as the number of outliers increases. Meanwhile, the L_2E MSE is much less sensitive to outliers. This example again underscores how our framework can perform a robust version of a structured regression problem utilizing a readily available non-robust implementation.

4.4 L_2E robust ℓ_1 penalized regression

Our last example is L_2E ℓ_1 penalized regression. We utilize the Lasso (Tibshirani, 1996)

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1$$

as our reference. For the corresponding L_2E problem, let $\mathbf{X} \in \mathbb{R}^{n\times p}$ with $\text{rank}(\mathbf{X}) = p$ and let $\phi(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_1$. Subproblem 1 for updating $\boldsymbol{\beta}$ in (6) reduces to iteratively performing:



(a) Convex regression MSE. (b) Convex regression time (sec).

Figure 5: *Monte Carlo experiments for convex regression with* $n = 1{,}000$ *observations drawn from a convex function with additive Gaussian noise. Boxplots of the (a) mean squared error (MSE) and (b) time in seconds required for* $50, 100,$ *and* 200 *outliers over* 100 *replicates compare the MLE (orange) and the* L_2E *(blue). The* L_2E *requires more time as its solution employs multiple computations of the MLE. Experiments highlight the trade-off between MSE and time between the MLE and* L_2E *solutions.*

1) Compute current residuals, 2) Update weights w_{ii} in (12), and 3) Update $\boldsymbol{\beta}$ with current residuals and gradient in Section 3.1 and apply the soft-thresholding operator in (9).

We illustrate with real data on prostate cancer patients from Stamey et al. (1989). The response $\mathbf{y} \in \mathbb{R}^n$ is the percent of Gleason score (measure of a prostate-specific antigen) for $n = 97$ patients receiving a radical prostatectomy. The design $\mathbf{X}$ contains measurements on $p = 8$ clinical variables. To introduce outliers in the covariates, we identify the top five percent of observations in $\mathbf{X}$ with highest leverage and scale these points by 3.3.

Figure 7 in the supplement depicts solution paths for Lasso, L_2E ℓ_1 penalized regression, sparse least trimmed squares (Sparse LTS) (Alfons et al., 2013; Yang et al., 2018), and exponential squared loss Lasso (ESL Lasso) (Wang et al., 2013) as a function of the

shrinkage factor $s = \frac{\|\boldsymbol{\beta}(\lambda)\|_1}{\|\hat{\boldsymbol{\beta}}_0\|_1}$. We set $\hat{\boldsymbol{\beta}}_0$ as the $\boldsymbol{\beta}$ estimate obtained at $\lambda = 0$ for each method and employ a λ sequence with a log linear scale of 15 values between 10^{-5} and a conservative data-dependent estimate of λ at which $\hat{\boldsymbol{\beta}}(\lambda) = \mathbf{0}$.

Since all methods employ the ℓ_1 penalty, the latter three can be viewed as alternative approaches to robust Lasso. Therefore, the Lasso solution paths, which quantify the relative contributions of the covariates to the regression model, without outliers (top-left panel) serve as a control. Ideally, a robust implementation preserves these relative contributions in the presence of outliers. Qualitatively, the L_2E solution paths most closely resemble the Lasso solution paths and suffer the least distortion in the presence of outliers. By comparison, Sparse LTS and ESL Lasso qualitatively appear very different from Lasso, even without outliers. We employ the default trimming percentage (retains 75 percent of the data) for Sparse LTS so it should be robust to the five percent of outliers.

Section 5.1 of the supplement contains more quantitative experiments with these four methods utilizing synthetic data. Table 3 of the supplement shows that the L_2E obtains lower relative error on average and additionally selects fewer false positives. Although Sparse LTS and ESL Lasso employ the ℓ_1 penalty for variable selection, they both select nearly all the variables in those experiments in the presence of outliers.

5 Discussion

Least squares regression models can be extended to encode a wide array of prior structure through non smooth penalties and constraints. While regression via least squares – and its constrained and penalized extensions – does not require any parametric assumptions, making a normality assumption on the residuals opens the door to applying the L_2E method

for robustly fitting a parametric regression model. In this work, we introduce a user-friendly computational framework, or recipe, for performing a wide variety of robust structured regression methods by minimizing the L_2 criterion. We highlight that our framework can "robustify" existing structured regression solvers by utilizing existing non-robust solvers in the $\boldsymbol{\beta}$-update step in a plug-and-play manner. Thus, our framework can readily incorporate newer and improved technologies for existing structured regression methods; as faster and better algorithms for these non-robust structured regression solvers appear, users may simply replace the previous solver with the new one in the $\boldsymbol{\beta}$-update step.

We also highlight the significance of the convergence properties of our computational framework. As long as the structural constraints or penalties satisfy convexity and continuity conditions, a solution obtained with our framework is guaranteed to converge to a first order stationary point. Since many commonly-used structural constraints and penalties satisfy these conditions, our framework provides convergence guarantees for robust versions of many non-robust methods with readily available software.

We close by noting that our L_2E framework focuses on structured regression problems under a normality assumption, which may not be appropriate in all situations. Meanwhile, the L_2E framework has also been used to robustly estimate parametric models under different distributional assumptions, e.g. Weibul (Yang and Scott, 2013), Poisson (Scott, 2001), and logistic (Chi and Scott, 2014). An interesting direction for future work is the development of a unified computational framework for fitting structured regression models under a wider range of distributional assumptions.

SUPPLEMENTARY MATERIAL

Title: Supplement to "A User-Friendly Computational Framework for Robust Structured

Regression with the L_2 Criterion" (.tex file)

Software: `L2E` R-package for performing L_2E structured regression. (GNU zipped tar file)

References

Alfons, A., Croux, C., Gelper, S. et al. (2013), "Sparse least trimmed squares regression for analyzing high-dimensional large data sets," *The Annals of Applied Statistics*, 7, 226–248.

Álvarez, E. E., and Yohai, V. J. (2012), "M-estimators for isotonic regression," *Journal of Statistical Planning and Inference*, 142, 2351–2368.

Andrews, D. F. (1974), "A robust method for multiple linear regression," *Technometrics*, 16, 523–531.

Audibert, J.-Y., Catoni, O. et al. (2011), "Robust linear least squares regression," *The Annals of Statistics*, 39, 2766–2794.

Aybat, N. S., and Wang, Z. (2016), "A Parallelizable Dual Smoothing Method for Large Scale Convex Regression Problems," arXiv:1608.02227 [math.OC].

Barlow, R. E., and Brunk, H. D. (1972), "The isotonic regression problem and its dual," *Journal of the American Statistical Association*, 67, 140–147.

Basu, A., Harris, I. R., Hjort, N. L., and Jones, M. C. (1998), "Robust and efficient estimation by minimising a density power divergence," *Biometrika*, 85, 549–559.

Bertsimas, D., and Mundru, N. (2021), "Sparse Convex Regression," *INFORMS Journal on Computing*, 33, 262–279.

Birke, M., and Dette, H. (2007), "Estimating a convex function in nonparametric regression," *Scandinavian Journal of Statistics*, 34, 384–404.

Blanchet, J., Glynn, P. W., Yan, J., and Zhou, Z. (2019), "Multivariate distributionally robust convex regression under absolute error loss," in *Advances in Neural Information Processing Systems*, pp. 11817–11826.

Brunk, H., Barlow, R. E., Bartholomew, D. J., and Bremner, J. M. (1972), "Statistical inference under order restrictions: The theory and application of isotonic regression," Tech. rep., Missouri Uuniversity Columbia Department of Statistics.

Chang, L., Roberts, S., and Welsh, A. (2018), "Robust lasso regression using Tukey's biweight criterion," *Technometrics*, 60, 36–47.

Chen, W., and Mazumder, R. (2021), "Multivariate Convex Regression at Scale," arXiv:2005.11588 [math.OC].

Chi, E. C., and Scott, D. W. (2014), "Robust Parametric Classification and Variable Selection by a Minimum Distance Criterion," *Journal of Computational and Graphical Statistics*, 23, 111–128.

Combettes, P. L., and Wajs, V. R. (2005), "Signal Recovery by Proximal Forward-Backward Splitting," *Multiscale Modeling & Simulation*, 4, 1168–1200.

Davies, P. L. (1993), "Aspects of robust linear regression," *The Annals of statistics*, 1843–1899.

Donoho, D. L., Liu, R. C. et al. (1988), "The "automatic" robustness of minimum distance functionals," *The Annals of Statistics*, 16, 552–586.

Dykstra, R. L., Robertson, T. et al. (1982), "An algorithm for isotonic regression for two or more independent variables," *The Annals of Statistics*, 10, 708–716.

Ghosal, P., and Sen, B. (2017), "On univariate convex regression," *Sankhya A*, 79, 215–253.

Guntuboyina, A.— (2015), "Global risk bounds and adaptation in univariate convex regression," *Probability Theory and Related Fields*, 163, 379–411.

Hannah, L. A., and Dunson, D. B. (2013), "Multivariate convex regression with adaptive partitioning," *The Journal of Machine Learning Research*, 14, 3261–3294.

Hjort, N. L. (1994), "Minimum L2 and Robust Kullback–Leibler Estimation," in *Proceedings of the 12th Prague Conference*.

Holland, P. W., and Welsch, R. E. (1977), "Robust regression using iteratively reweighted least-squares," *Communications in Statistics-theory and Methods*, 6, 813–827.

Lane, J. W. (2012), "Robust Quantile Regression Using L2E," Ph.D. thesis.

Lange, K. (2010), *Numerical analysis for statisticians*, Springer Science & Business Media.

— (2013), *Optimization*, Springer, 2nd ed.

— (2016), *MM Optimization Algorithms*, Philadelphia, PA: Society for Industrial and Applied Mathematics.

Lange, K., Chi, E. C., and Zhou, H. (2014), "A Brief Survey of Modern Optimization for Statisticians," *International Statistical Review*, 82, 46–70.

Lee, C.-I. C. et al. (1981), "The quadratic loss of isotonic regression under normality," *The Annals of Statistics*, 9, 686–688.

Lee, J. (2010), "L2E estimation for finite mixture of regression models with applications and L2E with penalty and non-normal mixtures," Ph.D. thesis.

Lim, C. H. (2018), "An efficient pruning algorithm for robust isotonic regression," in *Advances in Neural Information Processing Systems*, pp. 219–229.

Lim, E., and Glynn, P. W. (2012), "Consistency of multidimensional convex regression," *Operations Research*, 60, 196–208.

Lin, M., Sun, D., and Toh, K.-C. (2020), "Efficient algorithms for multivariate shape-constrained convex regression problems," arXiv:2002.11410 [math.OC].

Lozano, A. C., Meinshausen, N., and Yang, E. (2016), "Minimum Distance Lasso for robust high-dimensional regression," *Electronic Journal of Statistics*, 10, 1296 – 1340.

Ma, J., Qiu, W., Zhao, J., Ma, Y., Yuille, A. L., and Tu, Z. (2015), "Robust $L_{-}\{2\}E$ estimation of transformation for non-rigid registration," *IEEE Transactions on Signal Processing*, 63, 1115–1129.

Ma, J., Zhao, J., Tian, J., Tu, Z., and Yuille, A. L. (2013), "Robust estimation of nonrigid transformation for point set registration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2147–2154.

Mair, P., Hornik, K., and de Leeuw, J. (2009), "Isotone optimization in R: pool-adjacent-violators algorithm (PAVA) and active set methods," *Journal of Statistical Software*, 32, 1–24.

Mazumder, R., Choudhury, A., Iyengar, G., and Sen, B. (2019), "A computational framework for multivariate convex regression and its variants," *Journal of the American Statistical Association*, 114, 318–331.

Meng, X., and Mahoney, M. W. (2013), "Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression," in *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pp. 91–100.

Meyer, M. C. (2003), "A test for linear versus convex regression function using shape-restricted regression," *Biometrika*, 90, 223–232.

Ng, P., and Maechler, M. (2007), "A fast and efficient implementation of qualitatively constrained quantile smoothing splines," *Statistical Modeling*, 7, 315–328.

Nguyen, N. H., and Tran, T. D. (2013), "Robust Lasso With Missing and Grossly Corrupted Observations," *IEEE Transactions on Information Theory*, 59, 2036–2058.

Parikh, N., and Boyd, S. (2014), "Proximal Algorithms," *Found. Trends Optim.*, 1, 127–239.

Polson, N. G., Scott, J. G., and Willard, B. T. (2015), "Proximal Algorithms in Statistics and Machine Learning," *Statistical Science*, 30, 559 – 581.

Ramos, J. J. (2014), "Robust Methods for Forecast Aggregation," Ph.D. thesis.

Riani, M., Cerioli, A., Atkinson, A. C., and Perrotta, D. (2014), "Monitoring robust regression," *Electronic Journal of Statistics*, 8, 646–677.

Scott, A. I. (2006), "Denoising by Wavelet Thresholding Using Multivariate Minimum Distance Partial Density Estimation," Ph.D. thesis.

Scott, D. W. (1992), *Multivariate density estimation. Theory, practice and visualization*, John Wiley & Sons, Inc.

— (2001), "Parametric statistical modeling by minimum integrated square error," *Technometrics*, 43, 274–285.

— (2009), "The L2E method," *Wiley Interdisciplinary Reviews: Computational Statistics*, 1, 45–51.

Seijo, E., and Sen, B. (2011), "Nonparametric least squares estimation of a multivariate convex regression function," *The Annals of Statistics*, 39, 1633–1657.

She, Y., and Owen, A. B. (2011), "Outlier Detection Using Nonconvex Penalized Regression," *Journal of the American Statistical Association*, 106, 626–639.

Stamey, T., Kabalin, J., McNeal, J., Johnstone, I., Freiha, F., Redwine, E., and Yang, N. (1989), "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate II. Radical prostatectomy treated patients," *Journal of Urology*, 16, 1076–1083.

Terrell, G. R. (1990), "Linear Density Estimates," in *Proceedings of the Statistical Computing Section*, American Statistical Association, pp. 297–302.

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

Wang, J., and Ghosh, S. K. (2012), "Shape restricted nonparametric regression with Bernstein polynomials," *Computational Statistics & Data Analysis*, 56, 2729–2741.

Wang, X., Jiang, Y., Huang, M., and Zhang, H. (2013), "Robust variable selection with exponential squared loss," *Journal of the American Statistical Association*, 108, 632–643.

Warwick, J., and Jones, M. (2005), "Choosing a robustness tuning parameter," *Journal of Statistical Computation and Simulation*, 75, 581–588.

Yang, E., Lozano, A. C., Aravkin, A. et al. (2018), "A general family of trimmed estimators for robust high-dimensional data analysis," *Electronic Journal of Statistics*, 12, 3519–3553.

Yang, J., and Scott, D. W. (2013), "Robust fitting of a Weibull model with optional censoring," *Computational Statistics & Data Analysis*, 67, 149–161.

Yang, K., Pan, A., Yang, Y., Zhang, S., Ong, S. H., and Tang, H. (2017), "Remote sensing image registration using multiple image features," *Remote Sensing*, 9, 581.

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