




## ThrEEBoost: Thresholded Boosting for Variable Selection and Prediction via Estimating Equations

Ben Brown, Christopher J. Miller & Julian Wolfson

To cite this article: Ben Brown, Christopher J. Miller & Julian Wolfson (2017): ThrEEBoost: Thresholded Boosting for Variable Selection and Prediction via Estimating Equations, Journal of Computational and Graphical Statistics, DOI: [10.1080/10618600.2016.1247005](https://doi.org/10.1080/10618600.2016.1247005)

To link to this article: <http://dx.doi.org/10.1080/10618600.2016.1247005>

 View supplementary material 

 Accepted author version posted online: 16 Mar 2017.

 Submit your article to this journal 

 Article views: 39

 View Crossmark data 

# ThrEEBoost: Thresholded Boosting for Variable Selection and Prediction via Estimating Equations

Ben Brown\*

Christopher J. Miller<sup>†</sup>

Julian Wolfson<sup>‡</sup>

## Abstract

Most variable selection techniques for high-dimensional models are designed to be used in settings where observations are independent and completely observed. At the same time, there is a rich literature on approaches to estimation of low-dimensional parameters in the presence of correlation, missingness, measurement error, selection bias, and other characteristics of real data. In this paper, we present ThrEEBoost (*Thresholded EEBoost*), a general-purpose variable selection technique which can accommodate such problem characteristics by replacing the gradient of the loss by an estimating function. ThrEEBoost generalizes the previously-proposed EEBoost algorithm (Wolfson, 2011) by allowing the number of regression coefficients updated at each step to be controlled by a thresholding parameter. Different thresholding parameter values yield different variable selection paths, greatly diversifying the set of models that can be explored; the optimal degree of thresholding can be chosen by cross-validation. ThrEEBoost was evaluated using simulation studies to assess the effects of different threshold values on prediction error, sensitivity, specificity, and the number of iterations to identify minimum prediction error under both sparse and non-sparse true models with correlated continuous outcomes. We show that when the true model is sparse, ThrEEBoost achieves similar prediction error to EEBoost while requiring fewer iterations to locate the set of coefficients yielding the minimum error. When the true model is less sparse, ThrEEBoost has lower prediction error than EEBoost and also finds the point yielding the minimum error more quickly. The technique is illustrated by applying it to the problem of identifying predictors of weight change in a longitudinal nutrition study. Supplementary materials are available online.

*Keywords:* correlation, GEE, thresholding

---

\*Division of Biostatistics, School of Public Health, University of Minnesota, A460 Mayo Building MMC 303, Minneapolis, MN 55455

<sup>†</sup>3D Communications, 8386 Six Forks Road, Raleigh, NC 27615

<sup>‡</sup>Division of Biostatistics, School of Public Health, University of Minnesota, A460 Mayo Building MMC 303, Minneapolis, MN 55455

## 1 Introduction

Driven by the ever-increasing amount of high-dimensional data in biomedicine, much recent research has focused on how to do variable selection and prediction in problems where the number of predictors,  $p$ , is large in comparison to the number of observations,  $n$ . Traditional approaches like forward selection and backward elimination are widely employed but have limitations, particularly when the number of covariates is very large. For instance, it has been shown that the first variable selected in forward selection candidate models can often be the first removed in backwards elimination (Hocking, 1976). Methods such as the LASSO (Tibshirani, 1996) and SCAD (Fan and Li, 2001) generally offer superior variable selection and predictive performance to stepwise techniques, but have been applied almost exclusively to general linear (Park et al., 2006) and survival regression models (Fan and Li, 2002). Some authors have extended penalized approaches to more complex modeling situations such as correlated outcomes (Johnson et al., 2008) and missing covariates (Yang et al., 2005). However, the resulting statistical procedures often involve constrained optimization of nonconvex functions, and may therefore be too computationally intensive to apply in settings where  $p$  is on the order of hundreds or thousands. Ueki (2009) proposes a smooth thresholding approach to penalizing estimating equations, with the selection threshold determined by an adaptive LASSO type estimator. While smooth thresholding avoids convex optimization and therefore offers a computational speedup, the method still requires that a set of estimating equations be solved numerically for a large number of points on a two-dimensional grid of tuning parameters. Further, since the thresholding relies on an initial “full model” estimator, it is unclear how this technique generalizes to problems where  $p$  is large in relation to  $n$ .

As an alternative to penalization methods, Wolfson (2011) introduced EEBoost, a gradient descent-based method that can be used to perform variable selection for any regression problem where estimation of low-dimensional coefficients can be performed by solving an estimating equation. EEBoost iteratively constructs a set of models defined by coefficients using a modified steepest descent algorithm wherein the gradient of the loss function is replaced by the relevant estimating equation. The generic EEBoost algorithm is easily implemented using existing statistical software and can be applied to a wide variety of problems. Wolfson (2011) applied EEBoost to generalized estimating equations (GEE) (Liang and Zeger, 1986) for correlated data, and inverse probability weighted estimating equations methods for time-to-event data with missing covariates, and Janes et al. (2012) applied it to doubly robust semiparametric efficient estimating equations for continuous outcome data.

In this paper, we propose Thresholded EEBoost (ThrEEBoost), an extension to EEBoost wherein multiple coefficients may be updated at each iteration; the number of coefficients updated is controlled by a threshold parameter on the magnitude of the estimating equation. By allowing more coefficients to be updated at each iteration, ThrEEBoost can explore a greater diversity of variable selection “paths” (i.e., sequences of coefficient vectors) through the model space, possibly finding models with smaller prediction error than any of those on the path defined by EEBoost.

## 2 Boosting, EEBoost, and ThrEEBoost

Suppose we observe outcome data  $\mathbf{Y}_i$  and covariates  $\mathbf{X}_i$ ,  $i = 1, \dots, n$  with  $\mathbf{X}_i = \{X_{i1}, \dots, X_{ip}\}$ . We wish to predict future observations  $\mathbf{Y}_{n+1}, \dots, \mathbf{Y}_{n+K}$  that arise from the same distribution  $F(\mathbf{X}, \mathbf{Y})$

as the observed data. One common approach to prediction is to use a regression model in which the relationship between the outcome and covariates is governed by the linear predictor  $X_i\beta$ . The goal, then, is to estimate a set of coefficients,  $\hat{\beta}$ , that minimizes risk for a nonnegative loss function  $L$ :  $R(\beta) \equiv E_F[L(X, \beta)]$ , i.e., to obtain  $\hat{\beta}$  such that  $R(\hat{\beta}) \approx \min_{\beta} R(\beta) \equiv \beta_0$ . When  $p$  is small compared to  $n$ , estimation involves directly minimizing  $L$  with respect to  $\beta$  either analytically or numerically. In the case of least squares regression with independent scalars  $Y_i$ , parameter estimates are determined by  $\hat{\beta}_{LS} = \arg \min_{\beta} \sum_i (Y_i - X_i\beta)^2$ . More generally, if a complete or partial log-likelihood  $\ell$  is available, we can compute parameter estimates  $\hat{\beta}_{MLE} = \arg \min_{\beta} [-\ell(\beta, X)]$ . It is well known that when the number of covariates,  $p$ , is large in comparison to the sample size,  $n$ , using a subset of the  $p$  covariates to estimate  $Y_i$  will often lead to better prediction characteristics than estimating nonzero coefficients for the entire  $\beta$  vector (Wasserman, 2004). Hence, for large  $p$ , variable selection is an important step in computing  $\hat{\beta}$ .

The most commonly used variable selection techniques are penalization methods which restrict the magnitude of  $\beta$  to discourage unimportant predictors from having non-zero coefficients. Stronger restrictions yield simpler models with fewer selected covariates, while weaker ones lead to more nonzero coefficient estimates. For example, the LASSO (Tibshirani, 1996) and ridge regression (Hoerl and Kennard, 1970) restrict the  $L_1$  and  $L_2$  norms of  $\beta$  respectively.

An alternative to penalized methods is boosting or functional gradient descent (Freund and Schapire, 1997; Friedman et al., 2000; Friedman, 2004), a variable selection technique that additively builds a model using subsets of the predictors. Given a loss function  $L$ , one sets  $\beta \equiv \beta^{(0)} = \mathbf{0}$ , and then iteratively “nudges” the entry in  $\beta$  corresponding to the element of the gradient which is

largest in magnitude by some small amount  $\epsilon$ . A small increment  $\epsilon$  is chosen since the direction of steepest descent of  $L$  is only valid in a local neighborhood of  $\beta$ . Algorithm 1 describes the steps in a generic “ $\epsilon$ -boosting” algorithm. For linear regression with squared error loss, Algorithm 1 corresponds to the Forward Stagewise algorithm described in Efron et al. (2004), which is shown to be approximately equivalent (for large  $n$  and small  $\epsilon$ ) to Least Angle Regression and the LASSO. Prior to implementing the algorithm, all predictors need to be scaled and centered.

---

**Algorithm 1**  $\epsilon$ -boosting

---

**procedure**  $\epsilon$ -BOOSTSet  $\beta^{(0)}$  to the zero  $p$ -vector  $\mathbf{0}_p$ .**for**  $t = 0, \dots, T$  **do**    Compute the gradient of  $L$  at the current estimate  $\beta^{(t)}$ :  $\Delta = (\partial L(X, \beta) / \partial \beta_j)_{\beta = \beta^{(t)}}$     Identify the largest element of  $|\Delta|$ :  $j_t = \operatorname{argmax}_j |\Delta_j|$     Update  $\beta^{(t)}$  in the direction of  $j_t$ :  $\beta_{j_t}^{(t+1)} = \beta_{j_t}^{(t)} + \epsilon \operatorname{sign}(\Delta_{j_t})$ 

Algorithm 1 produces a sequence of coefficient estimates  $\mathbf{B} = \{\beta^{(0)}, \dots, \beta^{(T)}\}$  which define a path through the  $p$ -dimensional parameter space for the coefficients. Variable selection is achieved by “early stopping”, i.e., by selecting an element of  $\mathbf{B}$  for which some of the coefficients remain at zero (i.e., were never updated by the iterative boosting procedure). This step can employ holdout data, cross-validation, direct model scoring (via, e.g., the AIC or BIC), depending on the problem in question. The primary purpose of boosting techniques (and penalization methods) is to identify a set of candidate models from among a very large number of potential models; the hope is that at least some of these candidate models will have small mean squared prediction error (MSPE). We will emphasize this point later in arguing that the loss function used to calculate the MSPE need not play a central role in identifying a “good” set of candidate models.

## 2.1 EEBoost

Most existing variable selection procedures, whether based on penalization or boosting, focus on regression models which apply to relatively “clean” data, i.e., where outcomes are independent, completely observed, not subject to measurement error, etc. However, there is a vast and ever-expanding toolbox of regression techniques which accommodate these various types of “dirty” data. Many of these techniques avoid specifying a likelihood as the data characteristics being accommodated (e.g., correlation) may be poorly understood and not amenable to modeling. For such techniques, estimation typically involves solving a set of estimating equations.

As an alternative, Wolfson (2011) introduced EEBoost, an extension of the boosting algorithm applicable to problems where coefficient estimation is carried out by solving an estimating equation. The key to EEBoost is that estimating equations, while not exactly corresponding to the gradient of a loss function, often behave much like gradients and hence can take their place in a boosting algorithm. The predictors are scaled to have mean 0 and variance 1. In the rare instance of identical gradients, one of the variables with the tied max gradient could be selected at random to be updated. In the following iteration, it is then very unlikely that the gradient for that variable would again be tied with the others. Algorithm 2 presents EEBoost; note that the vector of estimating equations  $\mathbf{g}(\mathbf{X}, \boldsymbol{\beta})$  takes the place of the gradient  $|\partial L(\mathbf{X}, \boldsymbol{\beta})|/\partial \boldsymbol{\beta}$  from Algorithm 1.

---

### Algorithm 2 EEBoost

---

**procedure** EEBoost

Set  $\boldsymbol{\beta}^{(0)}$  to the zero  $p$ -vector  $\mathbf{0}_p$ .

**for**  $t = 0, \dots, T$  **do**

    Compute the estimating equations at the current estimate  $\boldsymbol{\beta}^{(t)}$ :  $\Delta = \mathbf{g}(\mathbf{X}, \boldsymbol{\beta})_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}}$

    Identify the largest element of  $|\Delta|$ :  $j_t = \operatorname{argmax}_j |\Delta_j|$

    Update  $\boldsymbol{\beta}^{(t)}$  in the direction of  $j_t$ :  $\boldsymbol{\beta}_{j_t}^{(t+1)} = \boldsymbol{\beta}_{j_t}^{(t)} + \epsilon \operatorname{sign}(\Delta_{j_t})$

---

By making use of estimating equations which account for important features of the data, EEBoost aims to produce paths containing coefficient estimates which yield smaller MSPE. Since there is no explicit loss function to minimize, the technique used to generate the variable selection path may not be directly linked to the procedure employed to select the point on that path which minimizes MSPE. For example, it can be shown that when observations are correlated within clusters, accounting for the correlation in estimation of regression parameters yields a smaller MSPE, even though the form of the MSPE does not acknowledge the correlated nature of the data. Hence, in this setting, applying EEBoost with the Generalized Estimating Equations produces variable selection paths which contain coefficient estimates yielding smaller MSPE than a standard LASSO approach which ignores correlation.

As an added benefit, EEBoost is also much faster than competing penalized estimating equation-based techniques, as it does not require solving constrained optimization problems. Wolfson (2011) reported computational speedups of up to 100-fold over existing methods.

## 2.2 Diversifying variable selection paths

The primary goal of EEBoost is to identify a set of candidate models (i.e., a sequence of regression coefficient estimates),  $\mathbf{B}$ , whose predictive performance can be assessed using external data, cross-validation, or other model scoring techniques. The hope is that there exists at least one  $\beta^{(k)} \in B$ , say  $\beta^{(k^*)}$ , such that  $|R(\beta^{(k^*)}) - R(\beta_0)| \leq \delta$  for some acceptably small  $\delta$ . In other words, the path  $\mathbf{B}$  must pass “close enough” to the true  $\beta_0$ ; no amount of cross-validation or model scoring can find a suitable  $\beta$  in  $\mathbf{B}$  otherwise.



In certain settings, there are theoretical guarantees that  $\mathbf{B}$  will contain a suitable  $\boldsymbol{\beta}^{(k^*)}$ . For instance, oracle results for several variants of the LASSO (Zou, 2006; Bunea et al., 2007; Van De Geer, 2008; Huang et al., 2013) guarantee that, if the penalty parameter  $\lambda_n$  is suitably chosen as  $n$  increases, then the LASSO solution  $\hat{\boldsymbol{\beta}}(\lambda_n)$  converges to  $\boldsymbol{\beta}_0$ . Previous work by Efron et al. (2004); Rosset et al. (2004); Rosset and Zhu (2007) demonstrated the equivalence (as  $T \rightarrow \infty$  and  $\epsilon \rightarrow 0$  with  $T \cdot \epsilon \rightarrow 0$ ) between boosting and  $L_1$  penalized paths, suggesting that similar results also hold for boosting. For a broad class of estimating equations, EEBoost can be viewed as gradient descent on a projected likelihood (see Wolfson (2011), using results from Small and Wang (2003), for details), and hence EEBoost closely approximates the variable selection path obtained by applying the LASSO to the aforementioned projected likelihood.

Unfortunately, these theoretical results provide limited insight into the real-world performance of boosting methods. Beyond the fact that asymptotic results may not apply with finite samples, in practice one must choose fixed values of the step length,  $\epsilon$ , and the number of iterations,  $T$ . Further, in settings where the loss function is more complex (e.g., projected likelihoods), existing oracle inequalities may not be applicable. In such cases, it is not clear that the boosting algorithms will yield good variable selection paths. We therefore propose a generalization of the EEBoost algorithm which allows it to generate a wide variety of variable selection paths by setting values of a single threshold parameter.

### 2.3 ThrEEBoost: Thresholded EEBoost

Algorithms 1 and 2 update one coefficient at each iteration, corresponding to the largest element of the gradient or estimating equation. Hence, if  $j_t = \arg \max_j \Delta_j$  is unique at each step,  $\beta^{(K)}$  can have at most  $K$  nonzero entries. Friedman (2004) proposed a generalization of boosting called Thresholded Gradient Descent Regularization (TGDR) wherein multiple elements of the coefficient vector  $\beta^{(K)}$  can be updated at each iteration. The elements to be updated correspond to the largest gradient values; how large the gradient needs to be for the corresponding coefficient to be updated is determined by a threshold parameter  $\tau \in [0, 1]$ . Specifically, given scaled predictors, coefficients are updated if  $|\Delta_j| \geq \tau \cdot \max_j |\Delta_j|$ .  $\tau = 0$  corresponds to updating every coefficient at every iteration, while  $\tau = 1$  is equivalent to the original boosting algorithm, assuming that the entries of  $\Delta$  are distinct.

We apply this idea to EEBoost, yielding ThrEEBoost, presented in Algorithm 3. Each value of  $\tau$  yields a distinct coefficient path,  $\mathbf{B}(\tau)$ . Further, for a fixed value of  $\tau$ , the computational burden of ThrEEBoost is no higher than EEBoost. When using cross-validation to select the optimal value of  $\tau$ , ThrEEBoost will be a factor of  $K$  times more computationally expensive, where  $K$  is the number of thresholding values that are chosen.

---

**Algorithm 3** ThrEEBoost

---

**procedure** THR3E3B3OST

Set  $\beta^{(0)} = \mathbf{0}$ 
**for**  $t = 0, \dots, T$  **do**

  Compute  $\Delta = \mathbf{g}(\mathbf{X}, \beta)_{\beta = \beta^{(t-1)}}$ 

  Identify  $J_t = \{j : |\Delta_j| \geq \tau \cdot \max_j |\Delta_j|\}$ 

  **for** all  $j_t \in J_t$  **do**

    Update  $\beta_{j_t}^{(t)} = \beta_{j_t}^{(t-1)} + \epsilon \text{sign}(\Delta_{j_t})$ 


---

## 2.4 Selecting the best model

In standard applications of boosting and EEBoost, the algorithm is run for a pre-determined number of iterations, producing a variable selection path from which one chooses the model (i.e., set of coefficient estimates) yielding the smallest MSPE,  $\frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m [y_{ij} - x_{ij}\beta^{(l)}]^2$ . The process is analogous to solving a LASSO problem for a sequence of values of the penalty parameter  $\lambda$ , then choosing the optimal value of  $\lambda$ .

The ThrEEBoost procedure involves repeating this process for different settings of the threshold parameter  $\tau$ , yielding a family of variable selection paths indexed by  $\tau$ . While applying ThrEEBoost with multiple  $\tau$  values increases the number of coefficient sets for which MSPE must be estimated, it poses no conceptual challenges. In practice, we recommend the following algorithm to choose  $\tau$  via cross-validation, minimizing the MSPE.

---

### Algorithm 4 Model Selection for ThrEEBoost

---

#### **procedure** CROSS VALIDATION

Divide the observations into  $K$  folds where  $\frac{1}{K}$  of the observations are used as a test set.

**for**  $k = 1, \dots, K$  **do**

    Apply ThrEEBoost for several values of  $\tau$ .

    Obtain the minimum MSPE of each candidate model on the test set.

    Select the  $\tau_k$  that minimizes MSPE.

Repeat across the  $K$  possible test sets and compute the mean of the selected  $\tau_k$ 's.

---

If cross-validation is computationally infeasible, then a model scoring criterion such as the QIC (Pan, 2001) can also be used: Assuming  $Q()$  is the quasi-likelihood,  $R$  is the working correlation structure,  $D$  is the data  $(X, Y)$ ,  $\Omega_I$  is the observed information, and  $\hat{V}_\tau$  is the sandwich variance estimate:

$$QIC(R) = -2Q(\hat{\beta}(R); I, D) + 2 * trace(\hat{\Omega}_I \hat{V}_r)$$

Both approaches are illustrated as part of the simulation study in Section 3. Cross validation is preferred and is utilized in the data application in Section 4.

### 3 Simulation Study

Simulations were conducted in R version 3.2.0 (R Core Team, 2015) using the `threeboost` package provided in the supplementary materials. The code for conducting this simulation study is also available in the supplementary materials.

#### 3.1 Sparse regression model with correlated outcomes

We simulated data for  $n = 30$  individuals with four correlated observations from each individual. A vector of covariates  $X_{ij}$  of length 50 was generated for each individual from a multivariate normal distribution with mean  $\mathbf{0}$  and covariance matrix  $\Sigma_X$  where  $\text{Var}(X_{ijk}) = 0.25$  and for each  $\text{Corr}(X_{ijk}, X_{ijl}) = 0.0, 0.3, 0.5,$  and  $0.7 \forall k \neq l$ . Each correlation level yielded similar results for all of our performance metrics, so we will focus our results on the scenario where  $\text{Corr}(X_{ijk}, X_{ijl}) = 0.3$ . The outcome variables for each individual  $Y_{ij}, i = 1, \dots, 30, j = 1, \dots, 4$ , were generated from a multivariate normal distribution with mean  $\mu_i = X_i \beta$ , with an exchangeable correlation matrix such that  $\text{Var}(Y_{ij}) = 1, \text{Corr}(Y_{ij}, Y_{ik}) = \rho, \forall j \neq k$ . The true values of the coefficient vector  $\beta = (\beta_0, \beta_1, \dots, \beta_{50})$  were set as:

$$\beta_m = \begin{cases} 0.5, 1 \leq m \leq 2 \\ 0.2, 3 \leq m \leq 5 \\ 0.0, 6 \leq m \leq 50 \end{cases}$$

Models which accommodate correlated data are generalized linear mixed models (GLMMs) and marginal models estimated via generalized estimating equations (GEE). GLMMs may be sensitive to assumptions about the distribution of the outcome and random effects. Variable selection techniques for GLMMs typically require maximizing the penalized likelihood and selecting both random and fixed effects (Schelldorfer et al., 2014), which can be computationally demanding. GEE provides an approach to estimation which is more robust to misspecification of the variance; however, existing approaches for variable selection with GEE (Johnson et al., 2008) are based on solving a set of penalized estimating equations, which is also computationally expensive. For this simulation, ThrEEBoost was performed using GEE,

$$\mathbf{g}(\boldsymbol{\beta}) = \sum_{i=1}^{30} \mathbf{X}'_i \mathbf{V}_i^{-1} (\mathbf{Y}_i - \mathbf{X}_i \boldsymbol{\beta})$$

where  $\mathbf{V}_i = \mathbf{A}_i^{1/2} \mathbf{R}_i(\rho) \mathbf{A}_i^{1/2}$  with  $\mathbf{A}_i = \text{diag}(\text{Var}(\mathbf{Y}_i))$  and  $\mathbf{R}_i(\rho)$  is the working correlation matrix. For these simulations, we assumed an exchangeable working correlation matrix such that  $\mathbf{R}_i = (1 - \rho)\mathbf{I} + \rho\mathbf{1}\mathbf{1}'$ .  $\rho$  was estimated at each iteration via a method of moments estimator using the current value  $\boldsymbol{\beta}^{(t)}$  at iteration  $t$ .

For each combination of  $\rho = \{0.0, 0.3, 0.6\}$  and  $\tau = \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0, \tau_{CV}\}$ , we generated 1000 datasets as outlined above and ran 500 ThrEEboost iterations, producing a variable selection path  $\{\boldsymbol{\beta}^1, \dots, \boldsymbol{\beta}^k\}$  for  $k = 1, \dots, 500$  for each simulated dataset. We selected  $\tau_{CV}$  using cross-validation using  $K = 10$  folds. We estimated MSPE at each point on a path by estimating

the average MSE across 100 datasets generated under the same assumptions used to generate the original data.

Table 1 shows the minimum MSPE, minimum QIC, number of iterations to reach the minimum MSPE, and variable selection sensitivity and specificity across the 1000 simulations for each combination of  $\rho$  and  $\tau$ . Sensitivity and specificity are given by

$$\text{Sensitivity} = \frac{\sum_{m=1}^p |\text{sign}(\hat{\beta}_m^k)|}{\sum_{m=1}^p |\text{sign}(\beta_m^{\text{true}})|}, \quad \text{Specificity} = \frac{\sum_{m=1}^p 1 - |\text{sign}(\hat{\beta}_m^k)|}{\sum_{m=1}^p 1 - |\text{sign}(\beta_m^{\text{true}})|}$$

where  $\text{sign}(\beta) = 0$  if  $\beta = 0$ .

For some simulation runs, the ThrEEBoost algorithm led to a sequence of coefficient estimates which began to alternate between 2 models before finding a solution that uniquely minimized the MSE. These are easy to detect and can be remedied in practice by selecting another thresholding value. The proportion of simulation runs resulting in numerical instability are reported in part (f) of Tables 1 and 2.

For each value of  $\rho$ , mean minimum MSPE decreased as  $\tau$  increased from 0.0 to 0.6. However, values of  $\tau \geq 0.6$  resulted in very similar MSEs. ThrEEBoost had similar median sensitivity to EEBoost across values of  $\rho$ . For each value of  $\rho$ , the sensitivity ranged from 0.80 to 1.00. Specificity increased with  $\tau$ , ranging from 0 for  $\tau = 0$  to 0.87 for  $\tau = 1$ . ThrEEBoost with  $\tau < 1$  reached minimum MSPE with considerably fewer iterations than with  $\tau = 1$  (i.e., EEBoost). On average, ThrEEBoost with  $\tau < 1$  located the point on the variable selection path achieving minimum MSPE in 3.5 to 14.4 times fewer iterations than EEBoost. Minimum mean QIC decreased as  $\tau$  increased. Figure 1 shows the average  $L_1$  distance from the true  $\beta$ , coefficient values, and MSE across the it-

erations of ThrEEBoost for different  $\tau$  values in the scenario where  $\rho=0.3$ . Using cross-validation to select an optimal thresholding value  $\tau_{CV}$ , all three cases chose a median  $\tau$  of 0.58. The distribution of the chosen  $\tau$  values are shown in figure 2. The results followed the same patterns for each simulated value of  $\rho$  and for each of  $\text{Corr}(X_{ijk}, X_{ijl})=0.0, 0.3, 0.5, \text{ and } 0.7$ .

### 3.2 Less sparse regression model with correlated outcomes

Next, we undertook an additional simulation study using the same setup as described in the previous section but with a less sparse true regression model for the mean defined by:

$$\beta_m = \begin{cases} 0.5, & 1 \leq m \leq 15 \\ 0.2, & 16 \leq m \leq 25 \\ 0.0, & 26 \leq m \leq 50 \end{cases}$$

Note that the number of nonzero regression coefficients (25) was nearly equal to the number of independent individuals (30). Due to the reduced sparsity of the model, we increased the number of iterations to 1500 for each of 1000 simulated datasets.

Table 2 summarizes the MSPE, QIC, sensitivity, specificity, number of iterations to find minimum MSPE, and rate of numerical instability of the algorithm. For all three settings of the correlation parameter  $\rho$ , mean minimum MSPE and QIC both showed a clear "U"-shaped pattern across  $\tau$ . MSPE achieved the lowest value at  $\tau = 0.4$ , with  $\tau = 0$  and  $\tau = 1$  yielding MSPE values 6-28% higher than this minimum value. The optimal  $\tau$  value to minimize QIC varied from 0.4 to 0.8 depending on  $\rho$ . The sensitivity and specificity results show the trade-off that is at play: sensitivity decreases and specificity increases as  $\tau$  goes from 0 to 1. In this case, specificity improves

dramatically up to  $\tau = 0.4$  but does not improve substantially with larger  $\tau$  values; and sensitivity declines steadily but modestly until  $\tau = 0.6$ . Figure 3 shows the  $L_1$  distance from the true  $\beta$ , the coefficient traceplots, and MSPE across iterations. Figure 5 shows the mean QIC across  $\tau$  values of 0, 1, and  $\tau_{CV}$  for the various  $\rho$  values. The results followed the same pattern for  $\rho = 0$  and  $\rho = 0.6$ . Results were also similar in scenarios where the pairwise correlation between covariates was set to 0, 0.5, and 0.7 (data not shown).

Using cross-validation to select  $\tau$  offered an improvement over EEBoost (i.e., ThrEEBoost with  $\tau = 1$ ). The MSPE shrunk by about 22%, 18%, and 7% for the cases where  $\rho=0.0, 0.3,$  and  $0.6,$  respectively. The median  $\tau$  selected was lower than in the sparse case with values of 0.38, 0.40, and 0.38, respectively. The distributions of  $\tau_{CV}$  are shown in Figure 4.

## 4 Data application - Box Lunch Study

We illustrate the application of ThrEEBoost to outcome data from the Box Lunch Study, a randomized controlled trial to evaluate the effect of portion size availability on caloric intake and weight gain (French et al., 2014). Two hundred and thirty-three eligible individuals were randomized to one of four groups: three “free lunch” groups and a “no free lunch” group which served as a control. The three “free lunch” conditions differed according to the number of calories provided in the daily box lunch: 400, 800, and 1600.

Here, we explore the factors associated with BMI in the “no free lunch” group consisting of  $n = 49$  individuals on whom BMI measurements were taken at four time points (baseline, 1, 3, and 6 months). There were 54 covariates of interest, including demographic (e.g. age, gender,



race, height, education), lifestyle (e.g. smoking status, physical activity levels), and psychosocial (e.g. frequency of self-weighing, degree of satisfaction with current weight) covariates recorded at baseline, and a variety of longitudinally-recorded food-related outcomes such as average daily caloric intake and average daily servings of fruits and vegetables. The outcome and predictors were scaled to have zero mean and unit variance prior to analysis.

ThrEEBoost was applied using the Gaussian Generalized Estimating Equations with an exchangeable working correlation structure. The algorithm was run for  $\tau = 0, 0.2, 0.4, 0.6, 0.8,$  and  $1$ , and the optimal model for each  $\tau$  was selected as the one which minimized the MSPE estimated by five fold cross-validation. The smallest MSPE overall (0.60) was achieved by ThrEEBoost with  $\tau = 0.4$ . To implement the LASSO, least angle regression (LARS) was utilized over five fold cross-validation to select an optimal penalty parameter which minimized the MSPE. Fitting the optimal LASSO model on the full data set, we obtained MSPE of 0.83. The non-zero coefficients for this model are summarized in Table 3, and compared to the coefficients from the LASSO fit with smallest cross-validated MSPE. The models selected by LASSO and ThrEEBoost share some covariates in common, but remain quite distinct. Overall, the ThrEEBoost model is more parsimonious than the LASSO model. Notably, the LASSO estimates relatively large coefficients for some variables (e.g., Dissatisfied with weight) which are not selected by ThrEEBoost. This may be due to the fact that the LASSO ignores the correlated nature of the outcome, and is therefore overly optimistic about the amount of statistical signal present in the data. Figure 6 summarizes the coefficients of the optimal ThrEEBoost model for various values of the threshold parameter  $\tau$ . The estimated coefficients for  $\tau = 0.4, 0.6, 0.8,$  and  $1$  are generally similar, with higher  $\tau$  values leading

to slightly more parsimonious models. However, as shown in Table 4, these subtle differences can yield very different prediction errors, hence the path diversity offered by ThrEEBoost is an asset.

## 5 Discussion

We have introduced a thresholded extension of the EEBoost algorithm, ThrEEBoost, and critically assessed its operating characteristics in variable selection and prediction in high-dimensional models. We have shown via a detailed simulation study that ThrEEBoost provides a predictive advantage over EEBoost. In cases when the true regression model was relatively sparse, ThrEEBoost required considerably fewer iterations than EEBoost to locate models with comparable performance. When the regression model was less sparse, varying the thresholding parameter in ThrEEBoost allowed for the exploration of a larger set of variable selection paths, leading to the discovery of models with lower MSPE.

Several limitations of the present study should be acknowledged. This simulation study focused solely on cases of normally distributed correlated outcome data, using GEE with an exchangeable working correlation. Further research is needed to clarify the benefits of thresholded variable selection with other correlation structures, and for other classes of estimating equations. Second, while the numerical experiments are promising, we have not provided theoretical results that guarantee, e.g., that ThrEEBoost possesses an oracle property. In ongoing work, we are exploring these theoretical properties of ThrEEBoost and clarifying its relationship to “hybrid” penalized variable selection procedures such as the elastic net.

## References

- Bunea, F., Tsybakov, A., and Wegkamp, M. (2007). Sparsity oracle inequalities for the Lasso. *Electronic Journal of Statistics*, 1:169–194.
- Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least Angle Regression. *The Annals of Statistics*, 32(2):407–451.
- Fan, J. and Li, R. (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association*, 96(456):1348–1360.
- Fan, J. and Li, R. (2002). Variable Selection for Cox’s Proportional Hazards Model and Frailty. *The Annals of Statistics*, 30:74–99.
- French, S. A., Mitchell, N. R., Wolfson, J., Harnack, L. J., Jeffery, R. W., Gerlach, A. F., Blundell, J. E., and Pentel, P. R. (2014). Portion size effects on weight gain in a free living setting. *Obesity (Silver Spring, Md.)*.
- Freund, Y. and Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting,. *Journal of Computer and System Sciences*, 55(1):119–139.
- Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive Logistic Regression: A Statistical View of Boosting. *The Annals of Statistics*, 28(2):337–374.
- Friedman, J. H. (2004). Gradient Directed Regularization. *Solutions*, 2004(3):1–30.
- Hocking, R. (1976). The analysis and selection of variables in linear regression. *Biometrics*, 32(1):1–49.

- Hoerl, A. A. E. and Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.
- Huang, J., Sun, T., Ying, Z., Yu, Y., and Zhang, C. H. (2013). Oracle inequalities for the lasso in the cox model. *Annals of Statistics*, 41(3):1142–1165.
- Janes, H., Frahm, N., DeCamp, A., Rolland, M., Gabriel, E., Wolfson, J., Hertz, T., Kallas, E., Goepfert, P., Friedrich, D. P., Corey, L., Mullins, J. I., McElrath, M. J., and Gilbert, P. (2012). MRKAd5 HIV-1 Gag/Pol/Nef vaccine-induced T-cell responses inadequately predict distance of breakthrough HIV-1 sequences to the vaccine or viral load. *PloS one*, 7(8):e43396.
- Johnson, B. A., Lin, D. Y., and Zeng, D. (2008). Penalized Estimating Functions and Variable Selection in Semiparametric Regression Models. *Journal of the American Statistical Association*, 103(482):672–680.
- Liang, K.-Y. and Zeger, S. L. (1986). Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22.
- Pan, W. (2001). Akaike’s Information Criterion in Generalized Estimating Equations. *Biometrics*, 57(1):120–125.
- Park, M. Y., Hastie, T., Young, M., and Hastie, P. T. (2006). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):659–677.
- R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

- Rosset, S. and Zhu, J. (2007). Piecewise linear regularized solution paths. *The Annals of Statistics*, 35(3):1012–1030.
- Rosset, S., Zhu, J., and Hastie, T. (2004). Boosting as a Regularized Path to a Maximum Margin Classifier. *Journal of Machine Learning Research*, 5:941–973.
- Schelldorfer, J., Meier, L., and Bühlmann, P. (2014). Glmlasso: An algorithm for high-dimensional generalized linear mixed models using  $l_1$ -penalization. *Journal of Computational and Graphical Statistics*, 23(2):460–477.
- Small, C. G. and Wang, J. (2003). *Numerical Methods for Nonlinear Estimating Equations*. Clarendon Press - Oxford.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B*, 58:267–288.
- Ueki, M. (2009). A note on automatic variable selection using smooth-threshold estimating equations. *Biometrika*, 96(1):1005–1011.
- Van De Geer, S. A. (2008). High-dimensional generalized linear models and the Lasso. *Annals of Statistics*, 36(2):614–645.
- Wasserman, L. (2004). *All of Statistics: A Concise Course in Statistical Inference*. Springer, New York.
- Wolfson, J. (2011). EEBoost: A General Method for Prediction and Variable Selection Based on Estimating Equations. *Journal of the American Statistical Association*, 106(493):296–305.

# ACCEPTED MANUSCRIPT

Yang, X., Belin, T. R., and Boscardin, W. J. (2005). Imputation and Variable Selection in Linear Regression Models with Missing Covariates. *Biometrics*, 61(2):498–506.

Zou, H. (2006). The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association*, 101(476):1418–1429.

$\rho$	$\tau = 0.0$	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$	$\tau = 1.0$	$\tau = \tau_{CV}$
(a) Mean Minimum Prediction Error							
0.0	1.17	1.13	1.09	1.08	1.07	1.07	1.09
0.3	1.16	1.12	1.09	1.06	1.06	1.06	1.08
0.6	1.15	1.11	1.07	1.06	1.05	1.06	1.06
(b) Median Sensitivity							
0.0	1.00	1.00	1.00	0.80	0.80	0.80	0.80
0.3	1.00	1.00	1.00	0.80	0.80	0.80	1.00
0.6	1.00	1.00	1.00	1.00	0.80	0.80	1.00
(c) Median Specificity							
0.0	0.00	0.36	0.64	0.80	0.84	0.87	0.76
0.3	0.00	0.31	0.62	0.78	0.82	0.87	0.76
0.6	0.00	0.29	0.60	0.76	0.82	0.87	0.73
(d) Mean Iterations to Minimum Prediction Error (IQR)							
0.0	11 (9, 13)	15 (12, 17)	22 (17, 26)	34 (25, 41)	44 (34, 53)	158 (125, 183)	32 (21, 41)
0.3	12 (10, 14)	16 (13, 17)	23 (18, 27)	34 (26, 40)	45 (36, 52)	159 (129, 180)	32 (22, 42)
0.6	14 (10, 15)	18 (14, 21)	26 (20, 32)	36 (28, 43)	45 (38, 54)	160 (132, 184)	35 (26, 44)
(e) Minimum Mean QIC							
0.0	212	199	181	169	162	155	175
0.3	210	199	179	165	160	153	173
0.6	210	197	177	169	160	156	175
(f) Proportion of simulations with numerical instability							
0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.3	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.01	0.01	0.00	0.01	0.01	0.01	0.01

Table 1: Mean minimum prediction error (a), median variable selection (b) sensitivity and (c) specificity, (d) mean number of iterations (25th and 75th percentile) to attain minimum prediction error, (e) minimum mean QIC, and (f) proportion of simulations where algorithm did not find a unique minimum MSE for ThrEEBoost in the sparse true model under different values of the threshold,  $\tau$  and correlation between intra-individual observations,  $\rho$ . Results are based on 1000 simulations, each with 500 iterations.

$\rho$	$\tau = 0.0$	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$	$\tau = 1.0$	$\tau = \tau_{CV}$
(a) Mean Minimum Prediction Error							
0.0	1.95	1.78	1.65	1.77	2.02	2.12	1.65
0.3	1.53	1.45	1.36	1.42	1.53	1.63	1.35
0.6	1.82	1.71	1.73	1.78	1.86	1.88	1.74
(b) Median Sensitivity							
0.0	1.00	0.96	0.92	0.88	0.84	0.80	0.92
0.3	1.00	0.96	0.92	0.88	0.88	0.88	0.92
0.6	1.00	1.00	0.92	0.92	0.92	0.88	0.96
(c) Median Specificity							
0.0	0.00	0.24	0.56	0.64	0.68	0.72	0.52
0.3	0.00	0.24	0.56	0.60	0.64	0.64	0.52
0.6	0.00	0.24	0.56	0.60	0.60	0.68	0.52
(d) Mean Iterations to Minimum Prediction Error (IQR)							
0.0	40 (40, 51)	43 (44, 50)	49 (49, 58)	63 (59, 79)	88 (73, 116)	696 (213, 966)	53 (49, 59)
0.3	47 (46, 52)	46 (46, 51)	52 (51, 58)	68 (63, 79)	102 (93, 120)	845 (871, 980)	55 (50, 61)
0.6	43 (45, 54)	42 (46, 51)	45 (49, 58)	59 (58, 76)	89 (83, 117)	767 (834, 1000)	53 (49, 59)
(e) Minimum Mean QIC							
0.0	340	314	295	317	354	378	296
0.3	334	319	287	287	299	391	282
0.6	470	470	460	458	418	536	487
(f) Proportion of simulations with numerical instability							
0.0	0.14	0.10	0.04	0.07	0.10	0.12	0.07
0.3	0.02	0.02	0.01	0.01	0.01	0.03	0.01
0.6	0.03	0.02	0.02	0.02	0.04	0.03	0.03

Table 2: Mean minimum prediction error (a), median variable selection (b) sensitivity and (c) specificity, (d) mean number of iterations (25th and 75th percentile) to attain minimum prediction error, (e) minimum mean QIC, and (f) proportion of simulations where algorithm did not find a unique minimum MSE for ThrEEBoost in the less sparse true model under different values of the threshold,  $\tau$ , and correlation between intra-individual observations,  $\rho$ . Results are based on 1000 simulations, each with 1500 ThrEEBoost iterations.



Variable	Coefficients	
	ThrEEBoost	LASSO
Race (Black)	0.27	0.24
Race (Hispanic)	0.24	0.35
Health (1=exc 5=poor)	0.17	0.08
Age	0.17	0.11
Lost control past 28 days	0.15	–
Education (HS)	0.14	0.14
Have fridge at work	0.12	0.19
TFEQ Disinhibition	0.10	0.32
Lbs gain before you noticed	0.06	0.16
Dissatisfied with weight	–	0.20
Light actvty min/day (251-2100)	–	0.15
Freq fast food (0=never 5=7+ times/week)	–	0.06
Limit food you eat	–	0.05
Marital status (Married)	-0.088	-0.11
Moderate activity min/day (2101-5900)	–	-0.05
Frequency self-weigh (0=never 5=every day)	–	-0.08
Freq restaurant/week	–	-0.10
TFEQ Hunger	–	-0.16

Table 3: Coefficients for the optimal ThrEEBoost ( $\tau = 0.4$ ) and LASSO models selected by cross-validated MSE. Small coefficients (magnitude  $< 0.05$ ) are omitted. “–” indicates that the variable was not selected in the model.

	ThrEEBoost $\tau$						LASSO
	0	0.2	0.4	0.6	0.8	1.0	
CV MSE	0.72	0.78	0.60	0.66	0.66	0.75	0.83

Table 4: Estimated mean squared prediction error for ThrEEBoost and LASSO models. (CV MSE) denotes models selected by minimizing cross-validated MSE.

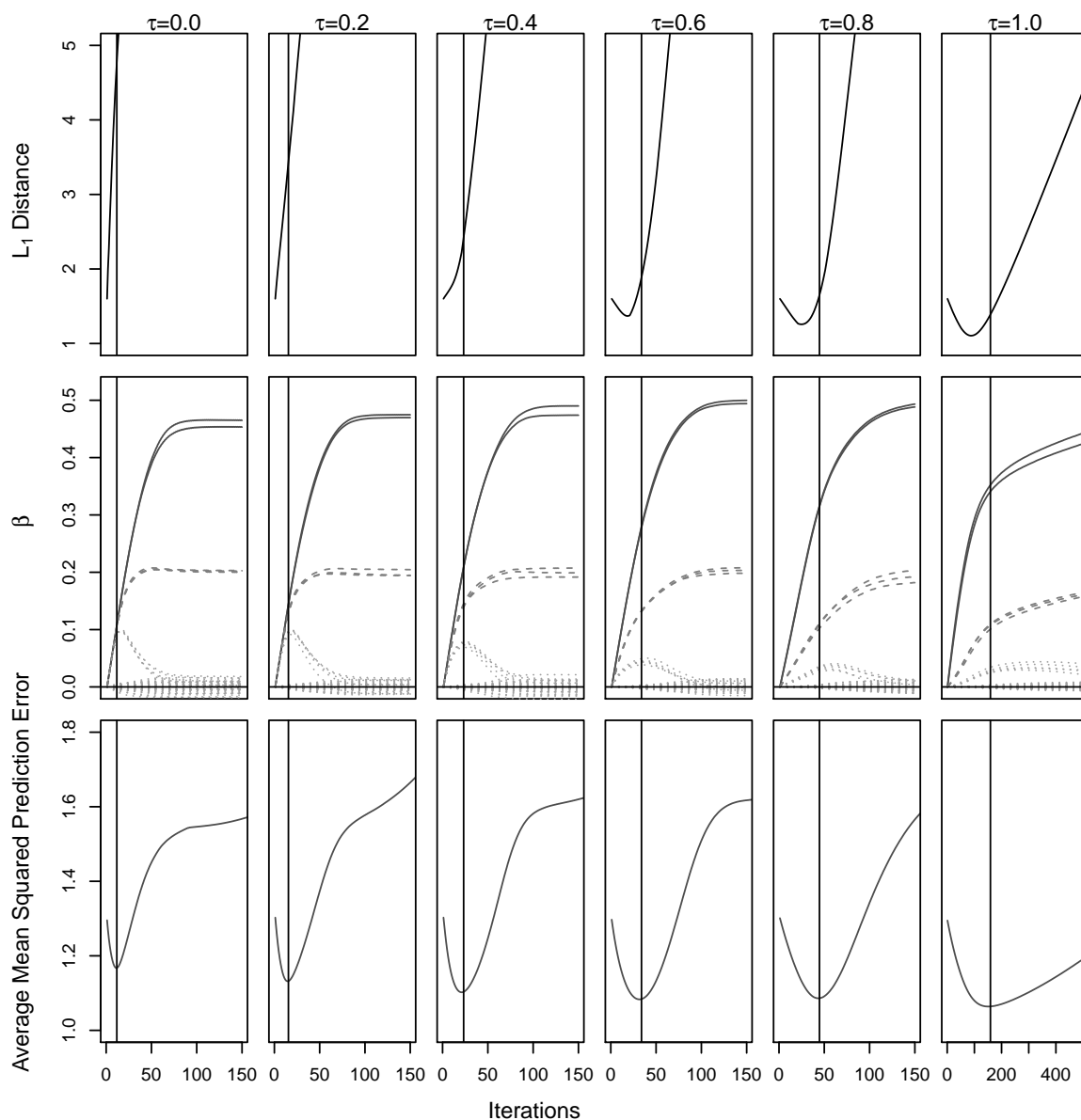


Figure 1: Average  $L_1$  distances from the true  $\beta$  (top row), estimated coefficient values (middle row) and MSPE (bottom row) across iterations for various values of  $\tau$ , when data are generated from a very sparse true regression model with an intra-individual correlation of  $\rho = 0.3$ . The solid, dashed, and dotted lines in the coefficient plots (middle row) represent coefficients with true values of 0.5, 0.2, and 0.0 respectively. Results are based on 1000 simulations, each with 500 ThrEEBoost iterations. The solid vertical lines show the iteration where the minimum mean squared error is achieved in each scenario.

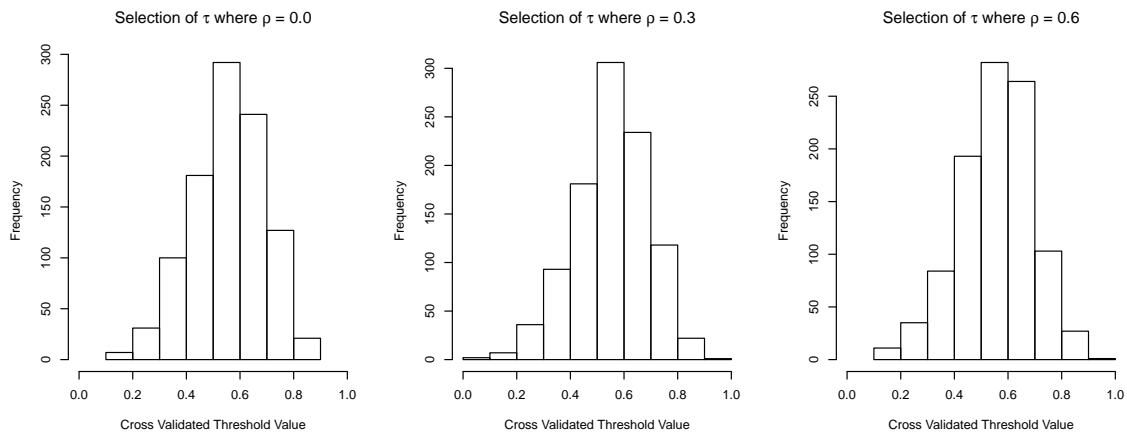


Figure 2: The distribution of selected  $\tau$  values via cross-validation. For each value of  $\rho$ , the median  $\tau_{CV}$  selected was 0.58.

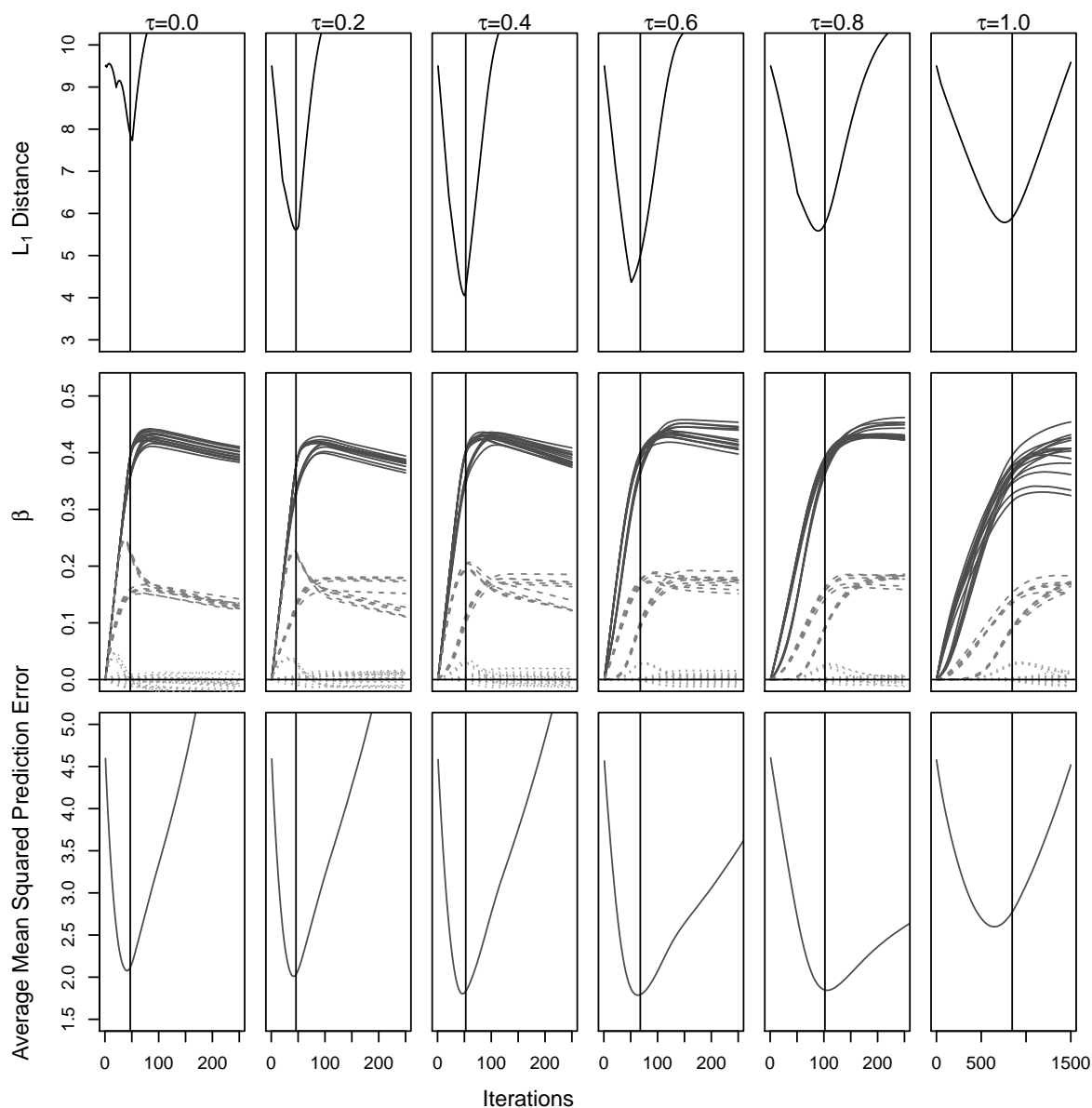


Figure 3: Average  $L_1$  distances from the true  $\beta$  (top row), estimated coefficient values (middle row) and MSPE (bottom row) across iterations for various values of  $\tau$ , when data are generated from a less sparse true regression model with an intra-individual correlation of  $\rho = 0.3$ . The solid, dashed, and dotted lines in the coefficient plots (middle row) represent coefficients with true values of 0.5, 0.2, and 0.0 respectively. Results are based on 1000 simulations, each with 1500 ThrEEBoost iterations. The solid vertical lines show the iteration where the minimum mean squared error is achieved in each scenario.

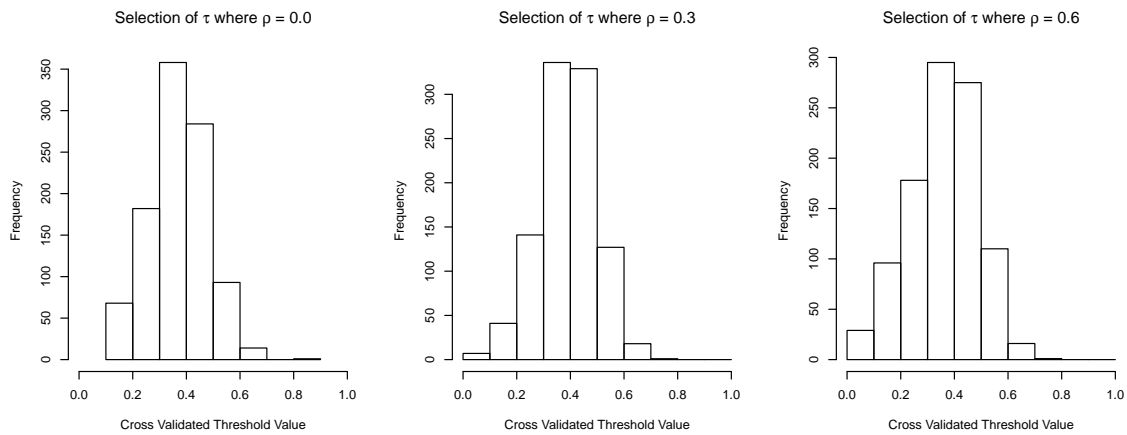


Figure 4: The distribution of selected  $\tau$  values via cross-validation. For each value of  $\rho$ , the median  $\tau_{CV}$  selected were 0.38, 0.40, and 0.38.

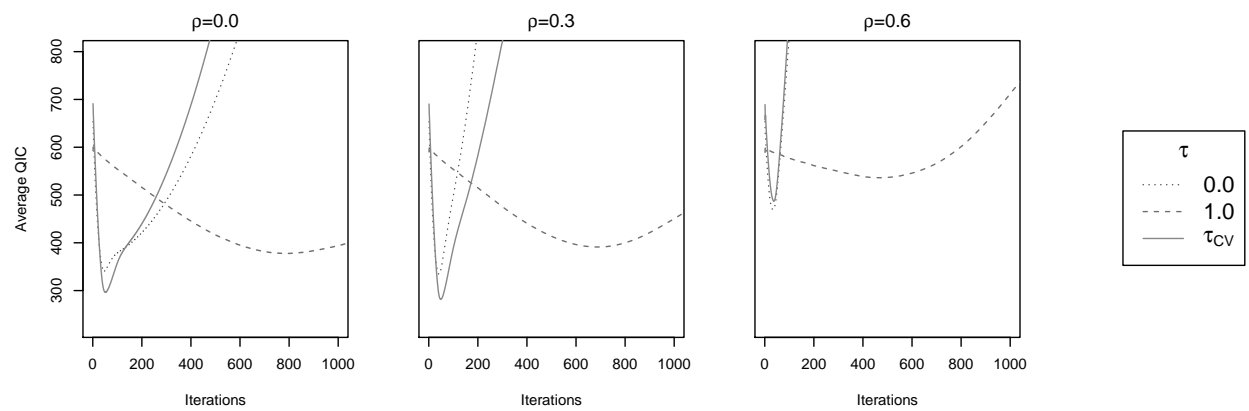


Figure 5: Average QIC when data are generated from a less sparse true regression model with an intra-individual correlation of  $\rho = 0.3$ . Results are based on 1000 simulations, each with 1500 ThrEEBoost iterations.

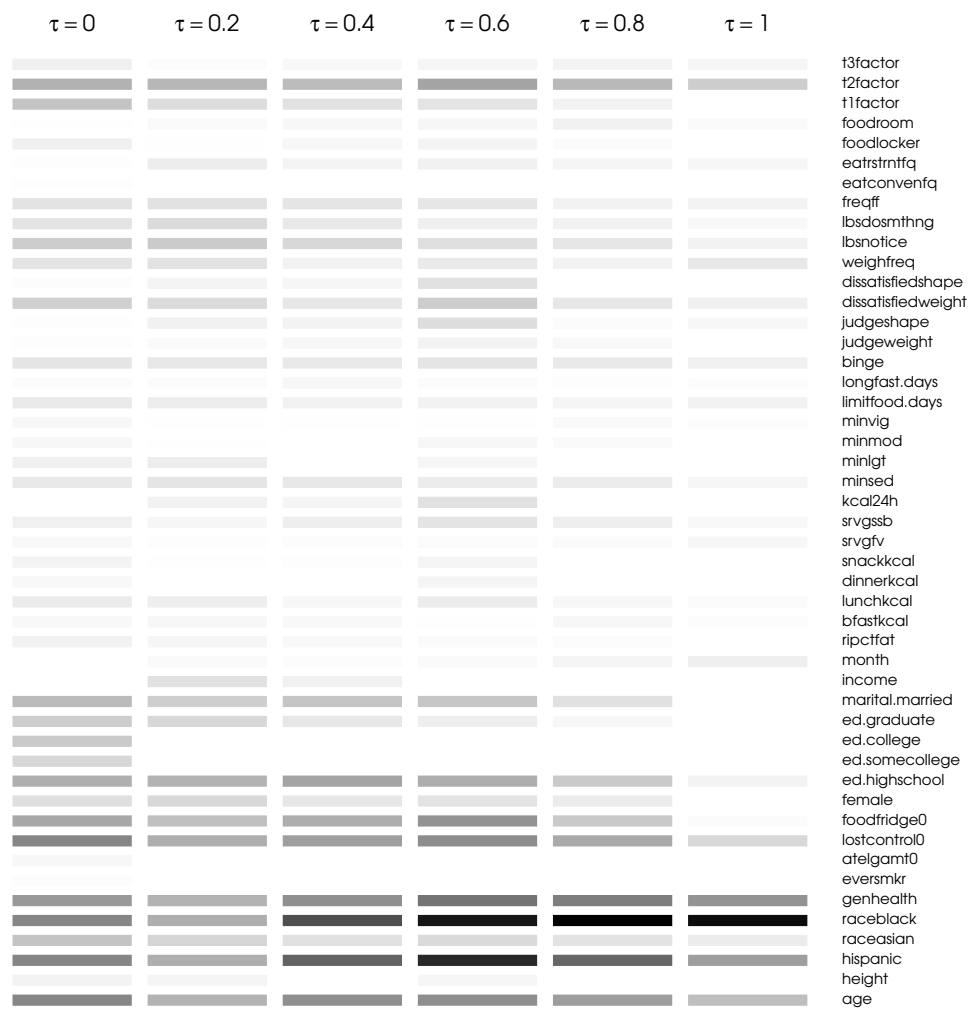


Figure 6: Coefficient magnitudes for the optimal models (chosen by cross-validated MSE) for different values of  $\tau$ . Each row corresponds to a different variable; darker shades of gray correspond to higher coefficient magnitudes. The names of the variables are displayed on the right; a data dictionary giving the variable descriptions is provided in the Supplementary Materials.