# Appendix for “Algorithms for Fitting the Constrained Lasso”

Brian R. Gaines
Department of Statistics, North Carolina State University

Juhyun Kim and Hua Zhou
Department of Biostatistics, University of California, Los Angeles (UCLA)

March 30, 2018

All section and equation numbers in this supplementary document are preceded by the letter A, while section and equation numbers without an A refer to the main paper.

## A.1 Generalized Lasso via Constrained Lasso

As stated by Theorem 1 in Section 2, it is always possible to reformulate and solve a generalized lasso as a constrained lasso. Here we provide the supporting proof.

*Proof.* Assume that $\text{rank}(\boldsymbol{D}) = r$, and consider the singular value decomposition (SVD)

$$\boldsymbol{D} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \begin{pmatrix}\boldsymbol{U}_1, \boldsymbol{U}_2\end{pmatrix}\begin{pmatrix}\boldsymbol{\Sigma}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{V}_1^T \\ \boldsymbol{V}_2^T\end{pmatrix} = \boldsymbol{U}_1\boldsymbol{\Sigma}_1\boldsymbol{V}_1^T,$$

where $\boldsymbol{U}_1 \in \mathbb{R}^{m\times r}$, $\boldsymbol{U}_2 \in \mathbb{R}^{m\times(m-r)}$, $\boldsymbol{\Sigma}_1 \in \mathbb{R}^{r\times r}$, $\boldsymbol{V}_1 \in \mathbb{R}^{p\times r}$, and $\boldsymbol{V}_2 \in \mathbb{R}^{p\times(p-r)}$. We define an augmented matrix

$$\tilde{\boldsymbol{D}} = \begin{pmatrix}\boldsymbol{U}_1\boldsymbol{\Sigma}_1\boldsymbol{V}_1^T \\ \boldsymbol{V}_2^T\end{pmatrix} = \begin{pmatrix}\boldsymbol{U}_1\boldsymbol{\Sigma}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{p-r}\end{pmatrix}\begin{pmatrix}\boldsymbol{V}_1^T \\ \boldsymbol{V}_2^T\end{pmatrix} = \begin{pmatrix}\boldsymbol{U}_1\boldsymbol{\Sigma}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{p-r}\end{pmatrix}\boldsymbol{V}^T$$

and use the following change of variables

$$\begin{pmatrix}\boldsymbol{\alpha} \\ \boldsymbol{\gamma}\end{pmatrix} = \tilde{\boldsymbol{D}}\boldsymbol{\beta} = \begin{pmatrix}\boldsymbol{U}_1\boldsymbol{\Sigma}_1\boldsymbol{V}_1^T \\ \boldsymbol{V}_2^T\end{pmatrix}\boldsymbol{\beta}, \tag{A.1}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^m$ and $\boldsymbol{\gamma} \in \mathbb{R}^{p-r}$.

Since the matrix $\boldsymbol{V}_2^T$ forms a basis for the nullspace of $\boldsymbol{D}$, $\mathcal{N}(\boldsymbol{D})$, it has rank $p-r$ and its columns are linearly independent of the columns of $\boldsymbol{D}$. Thus, the augmented matrix $\tilde{\boldsymbol{D}}$ has full column rank, and the new variables $\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix}$ uniquely determine $\boldsymbol{\beta}$ via

$$
\begin{aligned}
\boldsymbol{\beta} &= (\tilde{\boldsymbol{D}}^T\tilde{\boldsymbol{D}})^{-1}\tilde{\boldsymbol{D}}^T \begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \left[\boldsymbol{V}\begin{pmatrix} \boldsymbol{\Sigma}_1\boldsymbol{U}_1^T & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\begin{pmatrix} \boldsymbol{U}_1\boldsymbol{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\boldsymbol{V}^T\right]^{-1}\tilde{\boldsymbol{D}}^T\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \left[\boldsymbol{V}\begin{pmatrix} \boldsymbol{\Sigma}_1^2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\boldsymbol{V}^T\right]^{-1}\tilde{\boldsymbol{D}}^T\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \left[(\boldsymbol{V}^T)^{-1}\begin{pmatrix} \boldsymbol{\Sigma}_1^2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}^{-1}\boldsymbol{V}^{-1}\right]\tilde{\boldsymbol{D}}^T\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \left[\boldsymbol{V}\begin{pmatrix} \boldsymbol{\Sigma}_1^{-2} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\boldsymbol{V}^T\right]\tilde{\boldsymbol{D}}^T\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \boldsymbol{V}\begin{pmatrix} \boldsymbol{\Sigma}_1^{-2} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\boldsymbol{V}^T\boldsymbol{V}\begin{pmatrix} \boldsymbol{\Sigma}_1\boldsymbol{U}_1^T & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \boldsymbol{V}\begin{pmatrix} \boldsymbol{\Sigma}_1^{-2} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\begin{pmatrix} \boldsymbol{\Sigma}_1\boldsymbol{U}_1^T & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \begin{pmatrix} \boldsymbol{V}_1 & \boldsymbol{V}_2 \end{pmatrix}\begin{pmatrix} \boldsymbol{\Sigma}_1^{-1}\boldsymbol{U}_1^T & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \\
&= \boldsymbol{V}_1\boldsymbol{\Sigma}_1^{-1}\boldsymbol{U}_1^T\boldsymbol{\alpha} + \boldsymbol{V}_2\boldsymbol{\gamma} \\
&= \boldsymbol{D}^+\boldsymbol{\alpha} + \boldsymbol{V}_2\boldsymbol{\gamma},
\end{aligned}
$$

where $\boldsymbol{D}^+$ denotes the Moore-Penrose inverse of the matrix $\boldsymbol{D}$. However, since the original change of variables is $\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} = \tilde{\boldsymbol{D}}\boldsymbol{\beta}$, $\boldsymbol{\beta}$ is uniquely determined if and only if

$$
\begin{pmatrix} \boldsymbol{\alpha} \\ \gamma \end{pmatrix} \in \mathcal{C}(\tilde{\boldsymbol{D}}) = \mathcal{C}\left(\begin{pmatrix} \boldsymbol{U}_1\boldsymbol{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{p-r} \end{pmatrix}\right),
$$

if and only if

$$\boldsymbol{\alpha} \in \mathcal{C}(\boldsymbol{U}_1\boldsymbol{\Sigma}_1) = \mathcal{C}(\boldsymbol{U}_1) = \mathcal{C}(\boldsymbol{D}),$$

if and only if

$$\boldsymbol{U}_2^T\boldsymbol{\alpha} = \mathbf{0}_{m-r},$$

where $\mathcal{C}(\boldsymbol{D})$ is the column space of the matrix $\boldsymbol{D}$. Therefore, the generalized lasso problem (3) is equivalent to a constrained lasso problem

$$\begin{aligned} &\text{minimize} && \frac{1}{2}\left\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{D}^+\boldsymbol{\alpha} - \boldsymbol{X}\boldsymbol{V}_2\boldsymbol{\gamma}\right\|_2^2 + \rho\|\boldsymbol{\alpha}\|_1 \\ &\text{subject to} && \boldsymbol{U}_2^T\boldsymbol{\alpha} = \mathbf{0}_{m-r}, \end{aligned} \tag{A.2}$$

where $\boldsymbol{\gamma}$ remains unpenalized. □

## A.2 Constrained Lasso via Generalized Lasso

As detailed in Appendix A.1, it is always possible to reformulate and solve a generalized lasso as a constrained lasso. In this section, we demonstrate that it is not always possible to transform a constrained lasso to a generalized lasso. However, we first examine a situation where it is in fact possible to transform a constrained lasso to a generalized lasso.

### A.2.1 Reparameterization

Consider a constrained lasso with only equality constraints and $\boldsymbol{b} = \mathbf{0}_q$,

$$\begin{aligned} &\text{minimize} && \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \rho\|\boldsymbol{\beta}\|_1 \\ &\text{subject to} && \boldsymbol{A}\boldsymbol{\beta} = \mathbf{0}_q, \end{aligned} \tag{A.3}$$

where $\boldsymbol{A} \in \mathbb{R}^{q\times p}$ with $\operatorname{rank}(\boldsymbol{A}) = q$. Consider a matrix $\boldsymbol{D} \in \mathbb{R}^{p\times p-q}$ whose columns span the null space of $\boldsymbol{A}$. For example, we can use $\boldsymbol{Q}_2$ from the QR decomposition of $\boldsymbol{A}^T$. Then we can use the change of variables

$$\boldsymbol{\beta} = \boldsymbol{D}\boldsymbol{\theta},$$

so the objective function becomes

$$\text{minimize} \qquad \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{D}\boldsymbol{\theta}\|_2^2 + \rho\|\boldsymbol{D}\boldsymbol{\theta}\|_1, \tag{A.4}$$

and the constraints can be written as

$$\boldsymbol{A}\boldsymbol{\beta} = \boldsymbol{A}\boldsymbol{D}\boldsymbol{\theta} = \boldsymbol{0}\boldsymbol{\theta} = \boldsymbol{0}_q.$$

Thus the constraints vanish, as they hold for all $\boldsymbol{\theta}$, and we are left with an unconstrained generalized lasso (A.4). This result is not surprising in light of the result in Section 2, which showed that a generalized lasso reformulated as a constrained lasso has the constraints $\boldsymbol{U}_2^T\boldsymbol{\alpha} = \boldsymbol{0}_{m-r}$. That is a case where $\boldsymbol{b} = \boldsymbol{0}_q$ and the resulting constrained lasso solution can be translated back to the original generalized lasso parameterization via an affine transformation, in line with the result in this section that this is a situation where the constrained lasso can in fact be transformed to a generalized lasso.

Now consider the more general case with an arbitrary $\boldsymbol{b} \neq \boldsymbol{0}_q$. We can re-arrange the equality constraints as

$$\begin{aligned} \boldsymbol{A}\boldsymbol{\beta} &= \boldsymbol{b} \\ \boldsymbol{A}\boldsymbol{\beta} - \boldsymbol{b} &= \boldsymbol{0}_q \\ \begin{pmatrix} \boldsymbol{A}, -\boldsymbol{I}_q \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{b} \end{pmatrix} &= \tilde{\boldsymbol{A}}\tilde{\boldsymbol{\beta}} = \boldsymbol{0}_q, \end{aligned}$$

and then apply the above result using $\tilde{\boldsymbol{D}} = \tilde{\boldsymbol{Q}}_2$ from the QR decomposition of $\tilde{\boldsymbol{A}}^T$. However, now the reparameterized problem is

$$\tilde{\boldsymbol{\beta}} = \tilde{\boldsymbol{D}}\tilde{\boldsymbol{\theta}} \begin{pmatrix} \tilde{\boldsymbol{D}}_1\tilde{\boldsymbol{\theta}}_1 \\ \tilde{\boldsymbol{D}}_2\tilde{\boldsymbol{\theta}}_2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{b} \end{pmatrix},$$

we are still left with the constraint $\tilde{\boldsymbol{D}}_2\tilde{\boldsymbol{\theta}}_2 = \boldsymbol{b}$. Therefore, for equality constraints with $\boldsymbol{b} \neq \boldsymbol{0}_q$, a constrained lasso can be transformed into a constrained generalized lasso. This result is trivial, however, since a constrained lasso is always a constrained generalized lasso with $\boldsymbol{D} = \boldsymbol{I}_p$.

### A.2.2 Null-Space Method

Another common method for solving least squares problems with equality constraints (LSE) is the null-space method (Björck, 2015). To apply this method to the constrained lasso, we again restrict our attention to a constrained lasso with only equality constraints for the time being,

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \rho\|\boldsymbol{\beta}\|_1 \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{\beta} = \boldsymbol{b}. \end{aligned} \tag{A.5}$$

We will show that the application of the null-space method results in a shifted generalized lasso. Consider the QR decomposition of $\boldsymbol{A}^T \in \mathbb{R}^{p\times q}$, where $\text{rank}(\boldsymbol{A}) = q$,

$$\boldsymbol{A}^T = \boldsymbol{Q}\begin{pmatrix} \boldsymbol{R} \\ \boldsymbol{0} \end{pmatrix} \text{ with } \boldsymbol{Q}\boldsymbol{Q}' = \boldsymbol{I}_p,$$

and $\boldsymbol{Q} \in \mathbb{R}^{p\times p}$, $\boldsymbol{R} \in \mathbb{R}^{q\times q}$, and $\boldsymbol{0} \in \mathbb{R}^{p-q\times q}$. Let

$$\boldsymbol{X}\boldsymbol{Q} = \begin{pmatrix} \boldsymbol{X}_1, \boldsymbol{X}_2 \end{pmatrix} \text{ and } \boldsymbol{Q}'\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_2 \end{pmatrix},$$

with $\boldsymbol{X}_1 \in \mathbb{R}^{n\times q}$, $\boldsymbol{X}_2 \in \mathbb{R}^{n\times p-q}$, $\boldsymbol{\alpha}_1 \in \mathbb{R}^{q\times 1}$, and $\boldsymbol{\alpha}_2 \in \mathbb{R}^{p-q\times 1}$, then

$$\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{X}\boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{X}_1, \boldsymbol{X}_2 \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_2 \end{pmatrix} = \boldsymbol{X}_1\boldsymbol{\alpha}_1 + \boldsymbol{X}_2\boldsymbol{\alpha}_2.$$

As for the constraints, we have

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{R}^T, \boldsymbol{0}^T \end{pmatrix}\boldsymbol{Q}^T \Rightarrow \boldsymbol{A}\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{R}^T, \boldsymbol{0}^T \end{pmatrix}\boldsymbol{Q}^T\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{R}^T, \boldsymbol{0}^T \end{pmatrix}\begin{pmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_2 \end{pmatrix} = \boldsymbol{R}^T\boldsymbol{\alpha}_1.$$

Lastly, the penalty term becomes

$$\|\boldsymbol{\beta}\|_1 = \|\boldsymbol{Q}\begin{pmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_2 \end{pmatrix}\|_1 = \|\boldsymbol{Q}_1\boldsymbol{\alpha}_1 + \boldsymbol{Q}_2\boldsymbol{\alpha}_2\|_1.$$

Thus, putting these pieces together we can re-write (A.5) as

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|(\boldsymbol{y} - \boldsymbol{X}_1\boldsymbol{\alpha}_1) - \boldsymbol{X}_2\boldsymbol{\alpha}_2\|_2^2 + \rho\|\boldsymbol{Q}_1\boldsymbol{\alpha}_1 + \boldsymbol{Q}_2\boldsymbol{\alpha}_2\|_1 \\ \text{subject to} \quad & \boldsymbol{R}^T\boldsymbol{\alpha}_1 = \boldsymbol{b}. \end{aligned} \tag{A.6}$$

Since $\boldsymbol{R}^T$ is invertible, we can further directly incorporate the constraints in the objective function by plugging in $\boldsymbol{\alpha}_1 = (\boldsymbol{R}^T)^{-1}\boldsymbol{b}$,

$$\text{minimize} \quad \frac{1}{2}\|(\boldsymbol{y} - \boldsymbol{X}_1(\boldsymbol{R}^T)^{-1}\boldsymbol{b}) - \boldsymbol{X}_2\boldsymbol{\alpha}_2\|_2^2 + \rho\|\boldsymbol{Q}_1(\boldsymbol{R}^T)^{-1}\boldsymbol{b} + \boldsymbol{Q}_2\boldsymbol{\alpha}_2\|_1, \tag{A.7}$$

or more concisely as

$$\text{minimize} \quad \frac{1}{2}\|\tilde{\boldsymbol{y}} - \boldsymbol{X}_2\boldsymbol{\alpha}_2\|_2^2 + \rho\|\boldsymbol{c} + \boldsymbol{Q}_2\boldsymbol{\alpha}_2\|_1, \tag{A.8}$$

with $\tilde{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{X}_1(\boldsymbol{R}^T)^{-1}\boldsymbol{b}$ and $\boldsymbol{c} = \boldsymbol{Q}_1(\boldsymbol{R}^T)^{-1}\boldsymbol{b}$. So (A.8) resembles an unconstrained generalized lasso problem with $\boldsymbol{D} = \boldsymbol{Q}_2$, but it has been shifted by a constant vector $\boldsymbol{c} = \boldsymbol{Q}_1(\boldsymbol{R}^T)^{-1}\boldsymbol{b}$ that can not be decoupled from the penalty term. Therefore, it once again is not possible to solve a constrained lasso by reformulating it as an unconstrained generalized lasso. It should be noted that such a transformation is not possible even in the presence of only equality constraints, and the addition of inequality constraints would only further complicate matters. As pointed out by Gentle (2007), there is no general closed-form solution to least squares problems with inequality constraints.

## A.3 Subgradient Violations

As pointed out in Section 3.3, there is the potential for the subgradient conditions (9) to become violated if an inactive coefficient is moving too slowly. Here we provide the supporting derivations for this result and what is meant by too slowly. To preview the result, an inactive coefficient, $j \in \mathcal{A}^c$, with subgradient $s_j = \pm 1$ is moved to the active set if $s_j \cdot \frac{d}{d\rho}[\rho s_j] < 1$. To see this, without loss of generality assume that $s_j = -1$ for some inactive coefficient $\beta_j$, $j \in \mathcal{A}^c$. As given in Table 1, since $\rho$ is decreasing, $s_j$ is updated along the path via

$$[\rho^{(t+1)} s_j^{(t+1)}] = \rho^{(t)} s_j^{(t)} - \Delta\rho \cdot \frac{d}{d\rho}[\rho s_j],$$

which implies

$$s_j^{(t+1)} = \left(\rho^{(t)} s_j^{(t)} - \Delta\rho \cdot \frac{d}{d\rho}[\rho s_j]\right) / \rho^{(t+1)}. \tag{A.9}$$

Since $\rho$ is decreasing, $\rho^{(t)} > \rho^{(t+1)}$, but we define $\Delta\rho > 0$ which implies that $\Delta\rho = \rho^{(t)} - \rho^{(t+1)}$. Using this and $s_j^{(t)} = -1$, then for a given inactive coefficient $j \in \mathcal{A}^c$, (A.9) becomes

$$s_j^{(t+1)} \;=\; \left(-\rho^{(t)} - (\rho^{(t)} - \rho^{(t+1)})\frac{d}{d\rho}[\rho s_j]\right)/\rho^{(t+1)}. \tag{A.10}$$

To identify the trouble ranges for $\frac{d}{d\rho}[\rho s_j]$ that would result in a violation of the subgradient conditions, we can rearrange (A.10) as follows,

$$\begin{aligned}
s_j^{(t+1)} &= \left(-\rho^{(t)} - (\rho^{(t)} - \rho^{(t+1)})\frac{d}{d\rho}[\rho s_j]\right)/\rho^{(t+1)} \\
&= -\frac{d}{d\rho}[\rho s_j]\left(\frac{\rho^{(t)}}{\rho^{(t+1)}} - 1\right) - \frac{\rho^{(t)}}{\rho^{(t+1)}} \\
&= -\frac{d}{d\rho}[\rho s_j]\left(\frac{\rho^{(t)}}{\rho^{(t+1)}} - 1\right) - \frac{\rho^{(t)}}{\rho^{(t+1)}} + 1 - 1 \\
&= -\frac{d}{d\rho}[\rho s_j]\left(\frac{\rho^{(t)}}{\rho^{(t+1)}} - 1\right) + \left(1 - \frac{\rho^{(t)}}{\rho^{(t+1)}}\right) - 1 \\
&= \left(\frac{d}{d\rho}[\rho s_j] + 1\right)\left(1 - \frac{\rho^{(t)}}{\rho^{(t+1)}}\right) - 1.
\end{aligned} \tag{A.11}$$

The second term in the product in (A.11) is always negative, since $\rho^{(t)} > \rho^{(t+1)} \Rightarrow \rho^{(t)}/\rho^{(t+1)} > 1 \Rightarrow 0 > 1 - \left(\rho^{(t)}/\rho^{(t+1)}\right)$. Now, consider different values for $\frac{d}{d\rho}[\rho s_j]$:

i) $\frac{d}{d\rho}[\rho s_j] > -1$: When $\frac{d}{d\rho}[\rho s_j] > -1$, then $\left(\frac{d}{d\rho}[\rho s_j] + 1\right) > 0$, so the product term in (A.11) involves a positive number multiplied by a negative number and is thus negative. However, this would lead to $s_j < -1$ when 1 is subtracted from the product term, which is a violation of the subgradient conditions.

ii) $\frac{d}{d\rho}[\rho s_j] = -1$: This is fine as it maintains $s_j = -1$, since $\frac{d}{d\rho}[\rho s_j] = -1 \Rightarrow \left(\frac{d}{d\rho}[\rho s_j] + 1\right) = 0 \Rightarrow s_j^{(t+1)} = -1$.

iii) $\frac{d}{d\rho}[\rho s_j] < -1$: This situation is also fine as $\frac{d}{d\rho}[\rho s_j] < -1 \Rightarrow \left(\frac{d}{d\rho}[\rho s_j] + 1\right) < 0$, so the product term in (A.11) is positive and the subgradient is moving towards zero, which is fine since $j \in \mathcal{A}^c$.

The only issue, then, arises when $\frac{d}{d\rho}[\rho s_j] > -1$. These results can be summarized visually by looking at a plot of $\rho s_j$ as a function of $\rho$ (Figure A.1). Starting at point A in the graph, which corresponds to the point $(\rho^{(t)}, -\rho^{(t)})$ since $s_j = -1$, we see that $\frac{d}{d\rho}[\rho s_j] \leq -1$ is fine because it would result in $s_j$ moving towards 0, which is valid. However, $\frac{d}{d\rho}[\rho s_j] > -1$ would result in $s_j < -1$ and a violation of the subgradient conditions at the next kink. The main issue is the range $-1 < \frac{d}{d\rho}[\rho s_j] < 0$ (lower triangle in Figure A.1), in which the coefficient is moving in the correct direction but too slowly. Otherwise, for $\frac{d}{d\rho}[\rho s_j] > 0$, the coefficient would have already become active. The corresponding range for $j \in \mathcal{A}^c$ but $s_j = 1$ is $\frac{d}{d\rho}[\rho s_j] < -1$, which is derived similarly. Combining these two situations, the range to monitor can be written more succinctly as $s_j \cdot \frac{d}{d\rho}[\rho s_j] < 1$. Thus, to summarize, an inactive coefficient $j \in \mathcal{A}^c$ with subgradient $s_j = \pm 1$ and $s_j \cdot \frac{d}{d\rho}[\rho s_j] < 1$ needs to be moved back into the active set, $\mathcal{A}$, before the path algorithm proceeds to prevent a violation of the subgradient conditions (9).

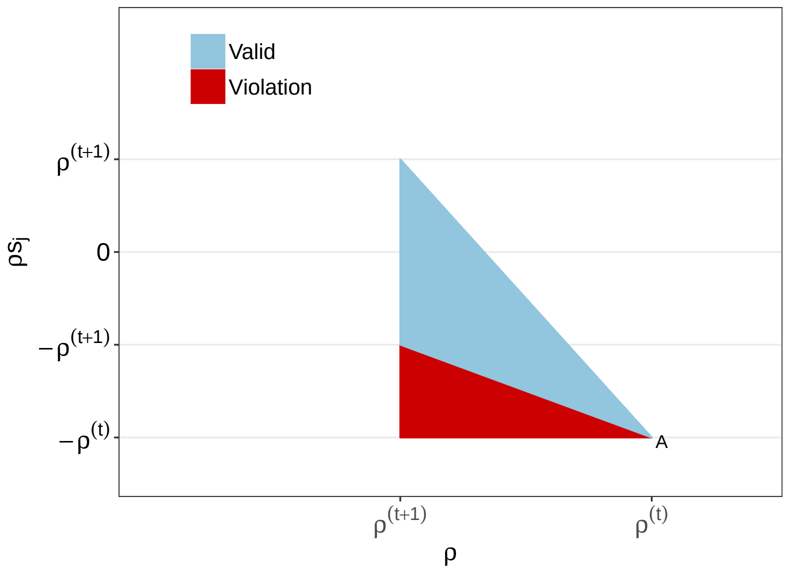


Figure A.1: Diagram of the event that needs to be monitored along the path to ensure the subgradient conditions are satisfied for $s_j = -1$. Starting at point A, as $\rho$ decreases the lower triangle represents the range of $\frac{d}{d\rho}[\rho s_j]$ that would result in a subgradient violation at kink $\rho^{(t+1)}$.

## A.4 Additional Simulation 1 Results

Figure A.2 shows the objective value errors (percent relative to QP), using original (left panel) and log (right panel) scales, of the path algorithm and ADMM in simulation 1 at $(n, p) = (500, 1000)$. The results are qualitatively the same for the other combinations of $(n, p)$.

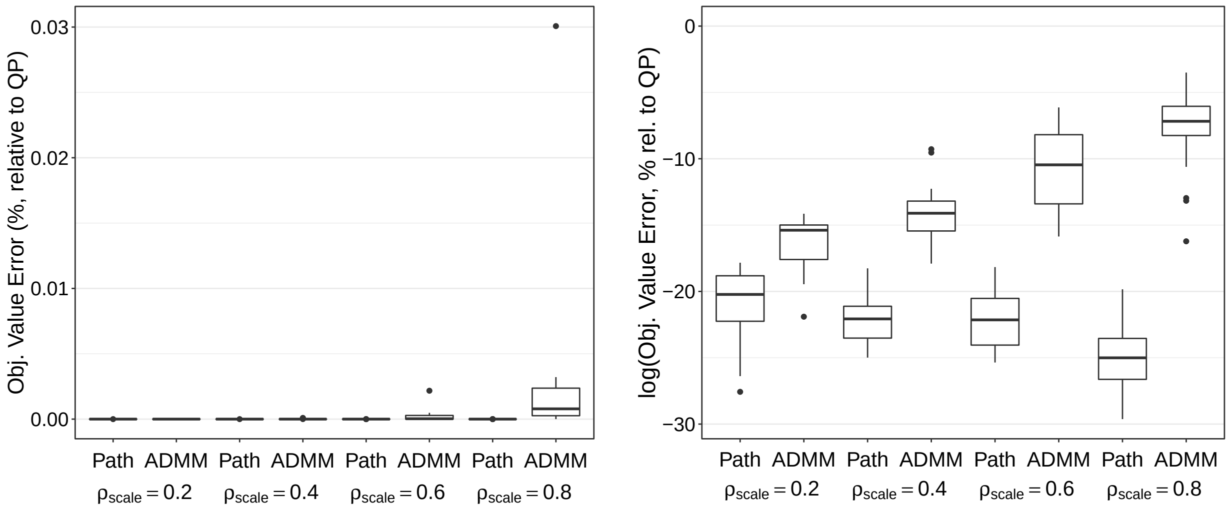


Figure A.2: Path algorithm yields solutions with smaller errors than ADMM. Objective value errors (percent) are relative to quadratic programing (QP) on the original scale (left panel) and log scale (right panel) at different values of $\rho_{\text{scale}} = \rho/\rho_{\max}$ for $(n, p) = (500, 1000)$. Although the accuracy of ADMM decreases as $\rho_{\text{scale}}$ increases, the error is generally less than 0.005% and thus is very low overall. The results are qualitatively the same for the other combinations of $(n, p)$.

## A.5 Additional Simulation 3 Results

Figure A.3 is similar to the Figure 2(c) of the main text, except with an extra setting $(n, p) = (2000, 4000)$. The results from this larger problem size show a similar pattern. ADMM outperforms QP as the problem size grows while the solution path algorithm consistently displays superior performance.

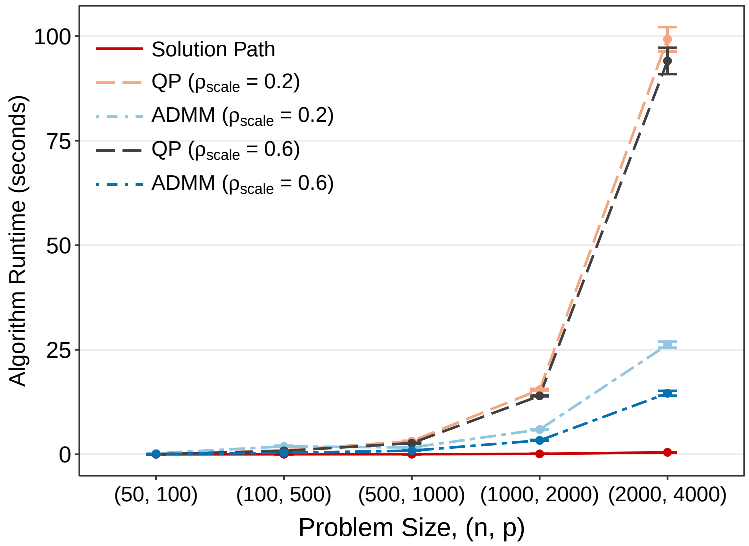


Figure A.3: Extra setting at $(n, p) = (2000, 4000)$ for simulation 3 shows a similar pattern as Figure 2(c) in the main text. The runtimes for the solution path algorithm are averaged across the number of kinks in the path to make the runtimes more comparable to the other algorithms estimated at one value of the tuning parameter, $\rho = \rho_{\text{scale}} \cdot \rho_{\max}$. ADMM outperforms QP as the problem size grows while the solution path algorithm consistently displays superior performance.

## A.6 Microbiome Data

Figures A.4 and A.5 display the solution paths and the observed vs. fitted values of the optimal model chosen using the extended Bayesian Information Criterion (EBIC) for the microbiome data discussed in Section 5.3. The EBIC includes an additional tuning parameter, which we set to 0.5 based on the results in the literature (Chen and Chen, 2008, 2012).

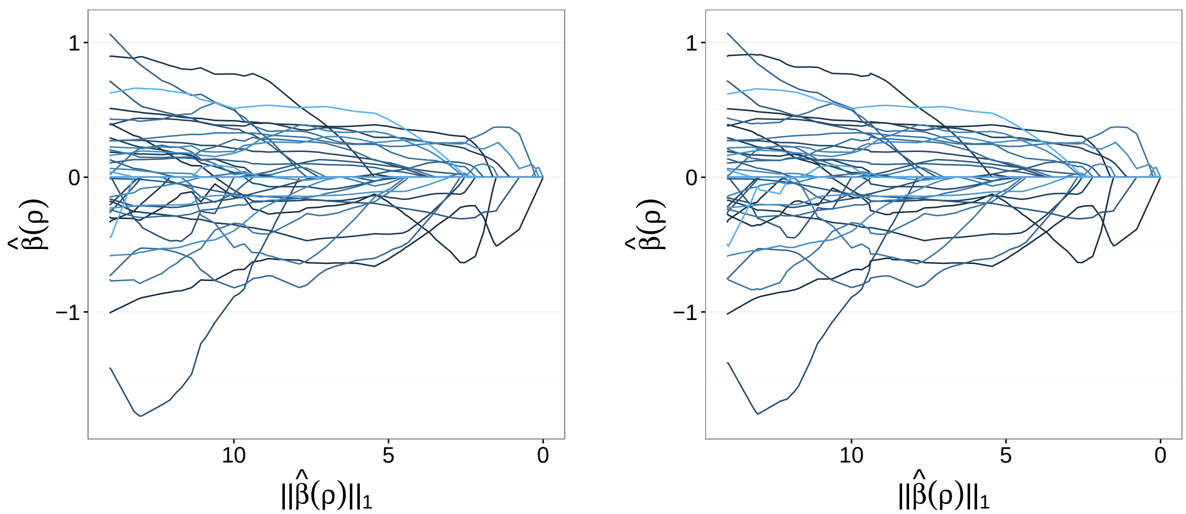


Figure A.4: Constrained lasso (left panel) and zero-sum regression (right panel) yield nearly identical solution paths for the microbiome dataset. The maximum absolute difference between the two solution paths is equal to 0.1785, at which the objective values differ by 0.0102.

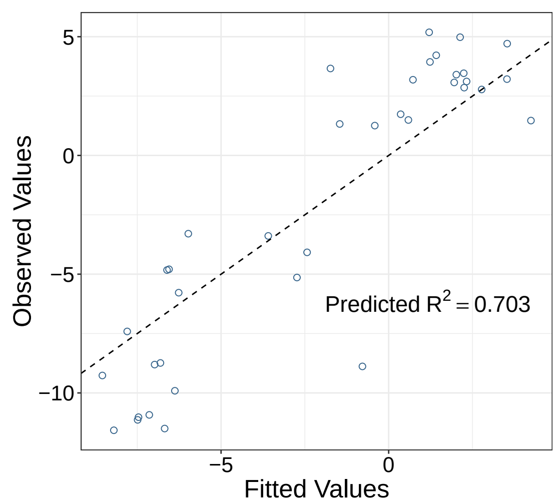


Figure A.5: The model with minimal EBIC occurs at $\rho = 120.14$ and yields predicted $R^2 = 0.703$ for urinary levels of 3-indoxyl sulfate (3-IS) in the microbiome dataset.

## A.7 Housing Data

Here we describe how we preprocessed the Ames housing dataset, which has 2,930 observations and 80 variables. (1) As per the data's documentation, the value NA represents the absence of a housing characteristic for most of the factor variables. In this case, NA values were replaced by a new factor level corresponding to "none" to distinguish them from missing entries. In situations where NA was not defined in the documentation, it was treated as a missing value. (2) For the majority of the factor variables, missing values were imputed using the variable's mode. For a handful of the factor variables, such as multiple variables describing the house's garage or basement, it was possible to infer the missing value based on the value of a closely-related variable. This was also done for continuous variables when possible, but two continuous variables had a large number of missing values and were handled differently. For "lot frontage," which has 490 missing entries, imputation was performed using the neighborhood's median value. The other continuous variable, "year garage was built," was imputed using "year built" since both variables are highly correlated ($r = .84$). After addressing the missing data, five houses were removed since they are either true outliers or represent unusual sales. Removal of these five houses was also encouraged by De Cock (2011). (3) For some factors, such as "roof material," more than 98% of the values are the same. Since factors with near-zero variance are relatively uninformative in terms of prediction, they were removed. The variables removed were "street," "utilities," "roof material," "heating," "pool quality," and "proximity to various conditions given more than one is present." (4) Highly-skewed variables were log transformed. They include "lot frontage," "overall quality," and "above ground living area," among others. (5) Indicator variables were constructed for each level of a factor variable. After the above data preprocessing, the dataset used in the analysis contains 2925 observations and 324 variables. Once the coefficient path is obtained from the path algorithm, we compute the regular BIC for each $\rho$ to select the final model. We found the lowest BIC at $\rho = 28.853$ with the predicted $R^2 = 0.893$ (based on the PRESS statistic). Figure A.6 shows the coefficient estimates for the 38 selected features at the optimal $\rho$. Based on the magnitude of the estimated coefficients, the two most important predictors are "overall quality" and "above ground living area," which is intuitive.

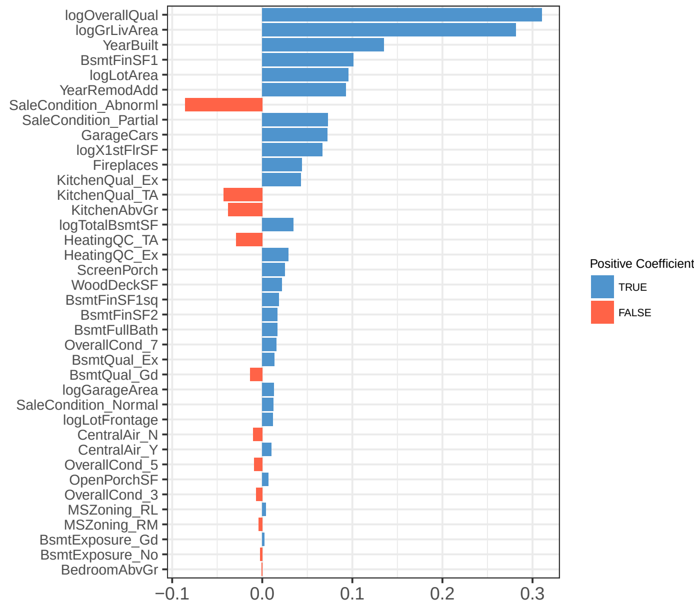


Figure A.6: The model with lowest BIC reveals 38 selected features. Blue and red bars indicate variables with positive and negative coefficient, respectively.

# References

Björck, Å. (2015), *Numerical Methods in Matrix Computations*, New York, NY: Springer.

Chen, J. and Chen, Z. (2008), "Extended Bayesian information criteria for model selection with large model spaces," *Biometrika*, 95, 759–771.

— (2012), "Extended BIC for small-n-large-P sparse GLM," *Statistica Sinica*, 555–574.

De Cock, D. (2011), "Ames, Iowa: Alternative to the Boston housing data as an end of semester regression project," *Journal of Statistics Education*, 19.

Gentle, J. E. (2007), *Matrix Algebra: Theory, Computations, and Applications in Statistics*, New York, NY: Springer.