SUPPLEMENTARY MATERIAL

Supplement to "J. P. Williams and J. Hannig (2017). Non-penalized variable selection in high-dimensional linear model settings via generalized fiducial inference. *Submitted.*"

S1. Technical details for algorithm computations.

S1.1. *Evaluating the model complexity decision function.* The purpose of this section is to provide the technical details for evaluating $h(\cdot)$ as defined in (4). Algorithm S1.1 which is adapted from Bertsimas, King and Mazumder (2016) is implemented for this purpose. Following the discussion in Section 2.1, evaluating $h(\beta_M)$ amounts to solving

$$\min_{b \in \mathbb{R}^p} g(b) \quad \text{subject to} \quad \|b\|_0 \leq |M| - 1,$$

with

$$g(b) = \frac{1}{2}\|X'(X_M\beta_M - Xb)\|_2^2.$$

As discussed in Bertsimas, King and Mazumder (2016), this L_0 minimization problem can be solved for a first-order stationary point with Algorithm S1.1 since $g(b) \geq 0$ is convex and has Lipschitz continuous gradient:

$$\nabla g(b) = X'XX'(Xb - X_M\beta_M) \quad \text{and}$$
$$\|\nabla g(b) - \nabla g(\widetilde{b})\|_2 \leq \lambda_{max}((X'X)^2)\|b - \widetilde{b}\|_2,$$

where $\lambda_{max}((X'X)^2)$ is the maximum of the eigenvalues of $(X'X)^2$.

The basic intuition is to update the solution vector iteratively in a gradient decent fashion. The cardinality constraint is imposed by only retaining the $|M| - 1$ largest in magnitude vector components in the gradient direction, at every iteration.

ALGORITHM S1.1. **(1)** *Initialize with some* $b^{(0)} \in \mathbb{R}^p$ *with* $\|b^{(0)}\|_0 \leq |M|$, *and set* $b^{(1)} = b^{(0)}_{-1}$ *where* $b^{(0)}_{-1}$ *is the vector* $b^{(0)}$ *with its smallest component (in absolute value) removed.*

(2) *For* $m \geq 1$, *set*

$$b_i^{(m+1)} = \begin{cases} c_i & \text{if } \ i \in \{(1), \ldots, (|M|-1)\} \\ 0 & \text{else} \end{cases}, \quad \text{for } \ i \in \{1, \ldots, p\},$$

where

$$c = b^{(m)} - \frac{1}{l}\nabla g(b^{(m)}) = b^{(m)} - \frac{X'XX'(Xb^{(m)} - X_M\beta_M)}{\lambda_{max}((X'X)^2)},$$

and $|c_{(1)}| \geq |c_{(2)}| \geq \cdots \geq |c_{(p)}|$.

(3) *Repeat until one of the following conditions are satisfied.*

(i) $g(b^{(m+1)}) = \frac{1}{2}\|X'(X_M\beta_M - Xb^{(m+1)})\|_2^2 < \varepsilon$, *or*

(ii) $g(b^{(m)}) - g(b^{(m+1)})$ *is arbitrarily small (not in absolute value), or*

(iii) *Some maximum number of iterations has been exceeded.*

S1.2. *Setting up the MCMC algorithm.* This section serves to provide the details of pseudo-marginal MCMC from Andrieu and Roberts (2009) used to compute the subset probabilities, $r(M|y)$ as in (6). Begin by defining

$$r(M, v|y) := C \cdot \pi^{\frac{|M|}{2}} \Gamma\Big(\frac{n-|M|}{2}\Big) \mathrm{RSS}_M^{-(\frac{n-|M|-1}{2})} h(v),$$

for some normalizing constant C, which is not a probability density function. Further, let

$$r_M(v|y) := \frac{r(M, v|y)Q_M(v)}{\int r(M, v|y)Q_M(v)\ dv}$$

denote the conditional density of v given a subset of covariates M, where $Q_M(v)$ is the density function associated with the location-scale multivariate T distribution in (7). Then

$$\frac{r_M(v|y)}{Q_M(v)} \underbrace{\int r(M, v|y)Q_M(v)\ dv}_{= r(M|y)} = r(M, v|y). \tag{S1}$$

Lastly, let the columns $B(i)$ of a new matrix B consist of a sample of size N from distribution (7), and denote the joint density function of the sample as $Q_M^N(B) := \prod_{i=1}^N Q_M(B(i))$, by independence. Then, in the convention of Andrieu and Roberts (2009), the GIMH algorithm has target distribution

$$\begin{aligned} r^N(M, B|y) &:= r(M|y) \cdot Q_M^N(B) \cdot \frac{1}{N} \sum_{i=1}^N \frac{r_M(B(i)|y)}{Q_M(B(i))} \\ &= Q_M^N(B) \cdot \frac{1}{N} \sum_{i=1}^N r(M, B(i)|y), \end{aligned}$$

where the second line is true by (S1). Observe that $r^N(M, B|y)$ has the desired distribution, $r(M|y)$, as its marginal distribution. The results of Andrieu and Roberts (2009) guarantee that MCMC with target distribution $r^N(M, B|y)$ will produce samples of M according to $r(M|y)$ asymptotically, as long as N is large enough.

Use $M_{(t)}$ and $B_{(t)}$ to denote the subset of covariates and sample of vectors, respectively, at step t of the GIMH algorithm. Then at step $t+1$ propose a new model, $\widetilde{M} \sim q(\cdot|M_{(t)})$, and a new sample of vectors, $\widetilde{B} \sim Q_{\widetilde{M}}^N(\cdot)$. This results in the following acceptance ratio

$$
\begin{aligned}
\rho(M_{(t)}, \widetilde{M}) &= \min\left\{ \frac{r^N(\widetilde{M}, \widetilde{B}|y) q(M_{(t)}|\widetilde{M}) Q^N_{M_{(t)}}(B_{(t)})}{r^N(M_{(t)}, B_{(t)}|y) q(\widetilde{M}|M_{(t)}) Q^N_{\widetilde{M}}(\widetilde{B})}, 1 \right\} \\
&= \min\left\{ \frac{\left[\frac{1}{N}\sum_{i=1}^N r(\widetilde{M}, \widetilde{B}(i)|y)\right] q(M_{(t)}|\widetilde{M})}{\left[\frac{1}{N}\sum_{i=1}^N r(M_{(t)}, B_{(t)}(i)|y)\right] q(\widetilde{M}|M_{(t)})}, 1 \right\}.
\end{aligned} \tag{S2}
$$

The pseudo-code for the constructed MCMC algorithm is presented next.

ALGORITHM S1.2. *Given some subset, $M_{(t)}$, of the p covariates at time t,*

(1) Sample.

$$
\widetilde{M} = \begin{cases} M_{(t)} \cup \{\textit{a new covariate}\} & \textit{w.p.} \ \frac{1}{3} \\ M_{(t)} \setminus \{\textit{an existing covariate}\} & \textit{w.p.} \ \frac{1}{3} \\ \big(M_{(t)} \setminus \{\textit{an existing covariate}\}\big) \cup \{\textit{a new covariate}\} & \textit{w.p.} \ \frac{1}{3} \end{cases}
$$

where a covariate is added to the subset $M_{(t)}$ with probability $w_j^{(t)}$ for $j \in \{1, \ldots, p - |M_{(t)}|\}$, and is dropped from $M_{(t)}$ with probability $v_i^{(t)}$ for $i \in \{1, \ldots, |M_{(t)}|\}$. This yields the proposal probability function

$$
q(\widetilde{M}|M_{(t)}) = \begin{cases} \frac{1}{3} w_j^{(t)} & \textit{if} \ \ |\widetilde{M}| > |M_{(t)}| \\ \frac{1}{3} v_i^{(t)} & \textit{if} \ \ |\widetilde{M}| < |M_{(t)}| \\ \frac{1}{3} w_j^{(t)} v_i^{(t)} & \textit{if} \ \ |\widetilde{M}| = |M_{(t)}| \end{cases},
$$

for $j \in \{1, \ldots, p - |M_{(t)}|\}$ and $i \in \{1, \ldots, |M_{(t)}|\}$. The vectors $\vec{w}^{(t)}$ and $\vec{v}^{(t)}$ are vectors of weights depending on $M_{(t)}$, which sum to 1.

Given the proposal $\widetilde{M}$, for $k \in \{1, \ldots, N\}$ generate

$$
\widetilde{B}(k) \sim t_{n-|\widetilde{M}|}\Big((X'_{\widetilde{M}} X_{\widetilde{M}})^{-1} X'_{\widetilde{M}} y, \frac{RSS_{\widetilde{M}}}{n - |\widetilde{M}|} (X'_{\widetilde{M}} X_{\widetilde{M}})^{-1} \Big).
$$

(2) Update.

$$
M_{(t+1)} = \begin{cases} \widetilde{M} & \textit{w.p.} \ \ \rho(M_{(t)}, \widetilde{M}) \\ M_{(t)} & \textit{w.p.} \ \ 1 - \rho(M_{(t)}, \widetilde{M}) \end{cases}
$$

where the acceptance ratio is given by $\rho(M_{(t)}, \widetilde{M})$ as in (S2).

One choice of weights is

$$
w_j^{(t)} := \frac{\widehat{\beta}_j^2}{\sum_{k=1}^{p-|M_{(t)}|} \widehat{\beta}_k^2}, \quad \text{for} \quad j \in \{1, \ldots, p - |M_{(t)}|\},
$$

and

$$v_i^{(t)} = \frac{\widehat{\beta}_i^{-2}}{\sum_{k=1}^{|M_{(t)}|} \widehat{\beta}_k^{-2}}, \quad \text{for} \quad i \in \{1, \ldots, |M_{(t)}|\},$$

where the coefficient estimates are the least squares estimates for the simple linear regression of each covariate on the response, y, separately. Another choice of weights could correspond to penalized regression coefficient estimates for the weights, such as those from LASSO. In practice, a well thought out choice of weights (versus uniform weights) can greatly improve the time it takes for the algorithm to find the true subset of covariates.