Supplementary Material

A Matrix Visualizations

We show here visualizations of the implicit reformulation matrices used to re-express Problem 1 into Problem 4. We emphasize that these large matrices are never actually constructed and are only used to derive our greedy block selection procedure.

$$A' = \begin{bmatrix} A & 0 & 0 & 0 & 0 \\ 0 & A & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & A & 0 \\ 0 & 0 & 0 & 0 & A \end{bmatrix}$$

$$Q^\intercal = \begin{bmatrix} Q_1^\intercal & 0 & 0 & 0 & 0 \\ 0 & Q_2^\intercal & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & Q_{p-1}^\intercal & 0 \\ 0 & 0 & 0 & 0 & Q_p^\intercal \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M = \begin{bmatrix} I & I \dots & I & I \end{bmatrix}$$

B Convergence Analysis

Here we show the proofs for the convergence analysis propositions presented in §3.3 of the main text.

B.1 Proof of Proposition 1

We start with the following preliminaries. We have,

$$\min_{\theta} \quad F(\theta) = f(\theta) + \lambda \|\theta\|_1, \tag{1}$$

where θ is both coordinate separable and block separable, with specified blocks and $f(\theta)$ is coordinate-wise $\mathcal{L}$-smooth. Recall we define direction vector d (same dimensions of θ) elementwise by

$$d_j = \begin{cases} |S_\lambda(\nabla_j f(\theta)| & \text{if } \theta_j = 0 \\ |\nabla_j f(\theta) + \text{sign}(\theta_j)\lambda| & \text{if } \theta_j \neq 0, \end{cases} \tag{2}$$

and we select the block to update whose corresponding elements of d have the largest ℓ_2-norm.

We also have this formula for the generalized directional derivative for $F(\theta)$ with respect to arbitrary direction vector v (Wu and Lange, 2008).

$$\partial_v F(\theta) = \sum_j \nabla_j f(\theta) v_j + \sum_j \begin{cases} \lambda \text{sign}(\theta_j) v_j & \text{if } \theta_j \neq 0 \\ \lambda |v_j| & \text{if } \theta_j = 0 \end{cases} \tag{3}$$

Part 1a: We first want to show that any limit point of a sequence of solutions θ^t generated via BGS-GBCD coincides with a minimum point of $F(\theta)$. We have that θ^t reaches a limit point when $d = \mathbf{0}$.

First we show that limit points of θ^t correspond to minimums of $F(\theta)$. Let θ^* be a limit point of sequence θ^t. Consider the j^{th} element of θ^*.

Case 1: $\theta_j^* = 0$ Since θ^* is stationary we have that $d_j = 0$, and that $|S_{\lambda_f}(\nabla_j f(\theta)| = 0$.

This implies $|\nabla_j f(\theta)| \leq \lambda$. Consider the j^{th} component contribution to the generalized directional derivative at θ^* for arbitrary direction v. We have that

$$(\partial_v F(\theta^*))_j = f(\theta) v_j + \lambda |v_j| \geq 0 \tag{4}$$

Case 2: $\theta_j^* \neq 0$ We have that $\nabla_j f(\theta) = -\text{sign}(\theta_j)\lambda$, and that

$$(\partial_v F(\theta^*))_j = \nabla_j f(\theta) v_j + \lambda \text{sign}(\theta_j) v_j = 0 \tag{5}$$

We compute $\partial_v F(\theta^*)$ by summing up the j^{th} components for both cases. Since v is arbitrary we have that

$$\partial_v F(\theta^*) \geq 0 \ \forall \ v,$$

and therefore θ^* is the minimum for $F(\theta)$.

Part 1b: We next show that minimum points of $F(\theta)$ correspond to stationary points. This is easy to verify from the LASSO subgradient optimality conditions. Let θ^* now correspond to the minimum of $F(\theta)$. Consider the j^{th} element of θ^*.

Case 1: $\theta_j^* = 0$

We have from the subgradient optimality conditions of the LASSO that $|\nabla_j f(\theta^*)| \leq \lambda$. Therefore $d_j = |S_\lambda(\nabla_j f(\theta^*)| = 0$.

Case 2: $\theta_j^* \neq= 0$

We have from the subgradient optimality conditions of the LASSO that $\nabla_j f(\theta) + \lambda \text{sign}(\theta_j) = 0$. Therefore $d_j = 0$.

Therefore combining parts 1a and 1b, we see that stationary points of our BGS-GBCD algorithm and minimum points of $F(\theta)$ coincide.

Part 2: Next, we want to show that any sequence of solutions generated by BGS-GBCD converges to a stationary/minimum point.

We first discuss some preliminaries. We rely on this key property, since $F(\theta)$ is both block *and* coordinate separable, a block update is equivalent to updating all of the coordinates in a block.

For this convergence analysis, we adopt the notation of good and bad coordinate updates introduced by Karimireddy et al. (2019). In a good coordinate update, the updated coordinate does not cross the origin. Formally, given coordinate $c_j > 0$ and updated coordinate c_j^+ we have that $c_j c_j^+ > 0$; all updates on a zero coordinate $c_j = 0$ are good updates. Bad coordinate updates, on the other hand, cross the origin, i.e. given $c_j > 0$ we have that $c_j c_j^+ \leq 0$.

After updating a block update, we can partition the updated coordinates in the block into good and bad update. We also adopt the post-processing step used in Karimireddy et al. (2019). For bad coordinate updates, we set $c_j^+ = 0$. This ensures that each coordinate will never have two consecutive bad updates, since updates on a zero coordinates are always good.

Karimireddy et al. (2019) shows these properties for coordinate updates on $F(\theta)$.

The progress made by a good update on coordinate j is bound by:

$$F(\theta_j^+) - F(\theta_j) \leq -\frac{1}{2\mathcal{L}}(d_j)^2, \tag{6}$$

and the progress made by a bad update on coordinate j after post-processing is bound by

$$F(\theta_j^+) - F(\theta_j) \leq 0. \tag{7}$$

Again since $F(\theta)$ is both block and coordinate separable, the progress made by a block update is equivalent to the sum of the progress made by each coordinate in the block, and the same holds for the bounds.

We now start our proof. Consider some arbitrary sequence of solutions θ^t generated by BGS-GBCD, and assume that this sequence does not converge to to a stationary/minimum point.

Since θ^t does not converge to a stationary point, we have that direction vector d is nonzero for each update. Therefore, in each block update, we have at least one index j in the block s.t. $d_j \neq 0$.The decrease in objective value for a block update is at least as large as the decrease in objective value for a coordinate update of a coordinate in the block.

If the update on coordinate j where $d_j \neq 0$ is a good coordinate update, we have this bound on the decrease in objective value for the corresponding *block* update,

$$F(\theta^{t+1}) - F(\theta^t) \leq -\frac{1}{2\mathcal{L}}(d_j)^2 \text{ where } \; d_j > 0. \tag{8}$$

If the update on coordinate j is a bad update, then we have that:

$$F(\theta^{t+1}) - F(\theta^t) \leq 0. \tag{9}$$

Due to post-processing, we can not have consecutive bad updates on the same coordinate. Since we have a finite number of coordinates, there can not exist a infinite contiguous subsequence of BGS-GBCD block updates where the only contribution to the decrease in objective values come from bad coordinate updates in the block (9).

Therefore, consider our sequence of solutions θ^t generated by BGS-GBCD updates. We have that all contiguous subsequences of BGS-GBCD updates must contain updates where the decrease in objective value is bound by (8). As such, one of two things can occur. Either the objective value continues to decrease by $-\frac{1}{2\mathcal{L}}(d_j)^2$ where $d_j > 0$, and since $F(\theta)$ is continuous and bounded from below by 0, the sequence converges to the minimum (which coincides to stationary point for θ^t). Or, we can have that $F(\theta^{t+1}) - F(\theta^t)$ converge to zero for a contiguous subsequence, if d_j converges to zero for all coordinates, but this would mean that θ^t converges to a stationary point. Either way, we reach a contradiction on the assumption that θ^t does not converge to a stationary/minimum point, which completes our proof.

B.2 Proof of Proposition 2

We derive here the bound on the progress made (decrease in objective value) for a BGS-GBCD update on $\theta_k^t = 0$, where each entry in the block is zero.

We discuss the following preliminaries. Consider vector θ; the entries of θ are separable into p prespecified blocks and θ_k corresponds to the sub-vector of entries of θ in block k. We define $snorm(x)$ as the sum of the ℓ_2-norms of each block in x, i.e.

$$snorm(x) = \sum_{k=1}^{p} \|x\|_2. \tag{10}$$

We have that the dual of this norm is equal to:

$$snorm(z)_* = max(\|z_1\|, \|z_2\|, \ldots, \|z_p\|). \tag{11}$$

We want to show:

$$F(\theta^{t+1}) - F(\theta^t) \leq \min_{\gamma \in \mathbb{R}^n} \quad \nabla f(\theta^t)^\intercal \gamma \ + \frac{\mathcal{L}}{2} snorm(\gamma) + \lambda \|\theta^t + \gamma\|_1 - \lambda \|\theta^t\|_1, \tag{12}$$

We have that:

$$F(\theta^{t+1}) - F(\theta^t) \leq (\theta^{t+1} - \theta^t)^\intercal \nabla f(\theta^t) + \frac{\mathcal{L}}{2} \|\theta^{t+1} - \theta^t\|_2^2 + \lambda \|\theta^{t+1}\|_1 - \lambda \|\theta^t\|_1$$

Let B_k be the indicies in θ that correspond to block θ_k. If block $\theta_k^t = 0$ was selected via the BGS rule and updated to obtain θ^{t+1}, we have that:

$$F(\theta^{t+1}) - F(\theta^t) \leq \sum_{j \in B_k} \left[(\theta_j^{t+1} - \theta_j^t) \nabla f(\theta_j^t) + \frac{\mathcal{L}}{2} (\theta_j^{t+1} - \theta_j^t)^2 + \lambda (|\theta_j^{t+1}| - |\theta_j^t|) \right]$$

And since $\theta_k^t = 0$, we follow the steps from Karimireddy et al. (2019) (Lemma 9) to obtain:

$$F(\theta^{t+1}) - F(\theta^t) \leq -\frac{1}{2\mathcal{L}} \sum_{j \in B_k} d_j^2 = -\frac{1}{2\mathcal{L}} \|d_k\|_2^2 \tag{13}$$

Now, consider this expression:

$$\nabla f(\theta^t)^\intercal \gamma \ + \frac{\mathcal{L}}{2} snorm(\gamma) + \lambda \|\theta^t + \gamma\|_1 - \lambda \|\theta^t\|_1,$$

where $\gamma \in \mathbb{R}^n$. We follow the algebra in Karimireddy et al. (2019) (Lemma 8)[1] to obtain:

$$\nabla f(\theta^t)^\intercal \gamma \ + \frac{\mathcal{L}}{2} snorm(\gamma) + \lambda \|\theta^t + \gamma\|_1 - \lambda \|\theta^t\|_1 \geq d^\intercal \gamma + \frac{\mathcal{L}}{2} snorm(\gamma)$$

We minimize both sides of the inequality with respect to γ to obtain.

$$\min_\gamma \left[\nabla f(\theta^t)^\intercal \gamma \ + \frac{\mathcal{L}}{2} snorm(\gamma) + \lambda \|\theta^t + \gamma\|_1 - \lambda \|\theta^t\|_1 \right] \geq \min_\gamma \left[d^\intercal \gamma + \frac{\mathcal{L}}{2} snorm(\gamma) \right]$$

Using convex conjugates we have that

$$\min_\gamma \left[d^\intercal \gamma + \frac{\mathcal{L}}{2} snorm(\gamma) \right] = -\frac{1}{2\mathcal{L}} (snorm(d)_*)^2 = -\frac{1}{2\mathcal{L}} (max(\|d_1\|_2 \ldots \|d_p\|_2))^2$$

[1] The only difference between the above expression and the one presented in Lemma 8 of Karimireddy et al. (2019) is the choice of the norm on γ. The algebraic steps do not involve the norm term.

Since θ_k^t is selected using BGS we have that:

$$max(\|d_1\|_2 \dots \|d_p\|_2) = \|d_k\|_2$$

and that:

$$\min_\gamma \left[\nabla f(\theta^t)^\intercal \gamma \ + \frac{\mathcal{L}}{2} snorm(\gamma) + \lambda \|\theta^t + \gamma\|_1 - \lambda \|\theta^t\|_1 \right] \geq -\frac{1}{2\mathcal{L}} \|d_k\|_2^2. \tag{14}$$

Combining (13) and (14) yields (12) completing our proof.

C Binning Formulation and Optimization Algorithm

Given a prespecified set of bins for each feature x_j, we can formalize binning into the FAST framework by adding the constraint that all value of β_j within each bin are fit the same value.

Constraint 1 *Assume each feature $x_j,\ j \in [p]$ has m prespecified sorted bins, $b_1 \dots b_m$. For each $b \in b_1 \dots b_m$ and all pairs of indicies $\{i_1,\ i_2\} \in b$, we add the constraint that $(\beta_j)_{i_1} = (\beta_j)_{i_2}$.*

Unconstrained Reformulation: We reformulate these constraints into the following unconstrained problem by defining a new set of decision vectors $\beta'_j \in \mathbb{R}^m$ for $j \in [p]$ to represent the value fit for each bin. We define mapping matrices $P_j \in \{0,1\}^{n \times m}$ such that $(P_j)_{(i,b)} = 1$ if $(x_j)_i$ is in bin b. The unconstrained problem can be expressed by:

$$\min_{\beta'} \quad \frac{1}{2} \|y - \sum_{j=1}^{p} Q_j^\intercal P_j \beta'_j\|_2^2 + \lambda_f \sum_{j=1}^{p} \|D' \beta'_j\|_1, \tag{15}$$

where D' is the $(m-1) \times m$ differencing matrix. This problem is equivalent to Problem (1) in the main text with Constraint (1) added. Problem 15 is block separable and we can apply our BGS-GBCD algorithm to solve the problem to optimality. We first discuss block updates below.

Block Update: Given a fixed block k and residual vector $r = y - \sum_{j \in \delta} Q_j^\intercal P_j \beta'_j$, we derive that each block update problem can be expressed by:

$$\min_{\beta'_k} \quad \frac{1}{2} \|W_j \bar{r} - W_j \beta'_k\|_2^2 + \lambda_f \|D' \beta'_j\|_1, \tag{16}$$

where $\bar{r} \in \mathbb{R}^m$ is a vector of the bin means of r, sorted with respect to x_j and binned w.r.t to the prespecified bins for x_j and $W_j \in \mathbb{R}^{m \times m}$ is a diagonal matrix with the square root of the cardinalities of each bin along the main diagonal, $\sqrt{|b_1|} \dots \sqrt{|b_m|}$.

Each block update minimizes:

$$\frac{1}{2} \|Q_k r - P_k \beta'_k\|_2^2 + \lambda_f \|D' \beta'_k\|_1,$$

with respect to β'_k.

Consider the smooth loss function; we can rewrite the loss functions by summing over the loss for each bin in x_j.

$$\frac{1}{2} \sum_{h=1}^{m} \sum_{i \in b_h} (r_i - (\beta'_k)_h)^2$$

Expanding the polynomial, we get that

$$\frac{1}{2} \sum_{h=1}^{m} \sum_{i \in b_h} (r_i^2 - 2(\beta'_k)_h r_i + (\beta'_k)_h^2)$$

$$= \frac{1}{2} \sum_{h=1}^{m} \sum_{i \in b_h} (r_i^2 - 2(\beta'_k)_h r_i + (\beta'_k)_h^2)$$

$$= \frac{1}{2}\sum_{h=1}^{m}(\sum_{i\in b_h} r_i^2 - 2(\beta'_k)_h \sum_{i\in b_h} r_i + |b_h|(\beta'_k)_h^2)$$

We complete the square to get:

$$= \frac{1}{2}\sum_{h=1}^{m}\left(|b_h|\left((\beta'_k)_h - \frac{1}{|b_h|}\sum_{i\in b_h} r_i\right)^2 + \mathcal{K}\right)$$

where $\mathcal{K}$ does not depend of $(\beta'_k)_h$.

We have

$$\frac{1}{|b_h|}\sum_{i\in b_h} r_i = \bar{r}_h,$$

i.e., the mean of r for bin h.

Combining this with the expression above, we have:

$$= \frac{1}{2}\sum_{h=1}^{m}\left(\sqrt{|b_h|}(\beta'_k)_h - \sqrt{|b_h|}\bar{r}_h\right)^2 + \mathcal{K}.$$

This is equivalent to Problem 16, since $\mathcal{K}$ does not depend on β'_k.

We can restrict our binning procedure so that all bins have the same cardinality:

$$|b_1| = |b_2| = \ldots |b_m|.$$

In this case, the block update is equivalent to:

$$\min_{\beta'_k} \quad \frac{1}{2}\|\bar{r} - \beta'_k\|_2^2 + \frac{\lambda_f}{|b_1|}\|D'\beta'_j\|_1, \tag{17}$$

This 1D-FLSA can be solved using dynamic programming to complete a block update.

Block Selection: Next, we derive our BGS block selection rule. Assume that each feature is binned into equally-sized bins $b_1 \ldots b_m$. We start with our unconstrained formulation in Problem 15.

$$\min_{\beta'} \quad \frac{1}{2}\|y - \sum_{j=1}^{p} Q_j^\intercal P_j \beta'_j\|_2^2 + \lambda_f \sum_{j=1}^{p}\|D'\beta'_j\|_1,$$

Again, let $\theta'_j \in \mathbb{R}^{m-1}$ be contain the successive differences of β'_j. We again use our implicit reformulation matrices to rexpress Problem 15 into this form:

$$\min_{\theta'} \quad \frac{1}{2}\|y - MQ^\intercal PA\theta'\|_2^2 + \lambda_f\|\theta'\|_1,$$

where:

$$P = \begin{bmatrix} P_1 & 0 & 0 & 0 & 0 \\ 0 & P_2 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & P_{p-1} & 0 \\ 0 & 0 & 0 & 0 & P_p \end{bmatrix}.$$

We can apply our BGS rule on this reformulated problem.

We have that the gradient of the smooth loss function here is equivalent too:

$$-A^\intercal P^\intercal Q M^\intercal r',$$

where r' is the residual. This can be computed quickly for each block j by sorting r' with respect to x_j computing bin sums and taking a rolling sum down the bin sums.

D Group ℓ_0-FAST Thresholding

Here, we derive the thresholding procedure for the block updates in group ℓ_0-FAST. Recall the optimization objective is given by:

$$\min_{\beta} \quad \frac{1}{2}\|y - \sum_{j=1}^{p} Q_j^\intercal \beta_j\|_2^2 + \lambda_f \sum_{j=1}^{p} \|D\beta_j\|_1 + \lambda_s \sum_{j=1}^{p} \mathbb{1}(\beta_j \neq \mathbf{0}). \tag{18}$$

Given fixed block k and residual vector $r = y - \sum_{j \in \delta} Q_j^\intercal \beta_j$, we write each block update problem as:

$$\min_{\beta_k} \quad \frac{1}{2}\|Q_k r - \beta_k\|_2^2 + \lambda_f \|D\beta_k\|_1 + \lambda_s \mathbb{1}(\beta_k \neq \mathbf{0}). \tag{19}$$

Let β_k^* be the optimal solution to:

$$\min_{\beta_k} \quad \frac{1}{2}\|Q_k r - \beta_k\|_2^2 + \lambda_f \|D\beta_k\|_1.$$

Consider the following two cases.

Case 1: $\beta_k = 0$ The minimum objective value of Problem (19) is equal to

$$\frac{1}{2}\|Q_k r\|_2^2 = \frac{1}{2}\|r\|_2^2, \tag{20}$$

since Q_k is the sorting matrix.

Case 2: $\beta_k \neq 0$ The minimum objective value of Problem (19) is equal to:

$$\frac{1}{2}\|Q_k r - \beta_k^*\|_2^2 + \lambda_f \|D\beta_k^*\|_1 + \lambda_s \tag{21}$$

Combining (20) and (21) we get

$$\frac{1}{2}\|r\|_2^2 - \frac{1}{2}\|Q_k r - \beta_k^*\|_2^2 - \lambda_f \|D\beta_k^*\|_1 \leq \lambda_s,$$

which is the desired thresholding inequality.

E Experiment Details

We show here additional details for our experiments. Code to reproduce our experiments along with a packaged implementation of FAST will be made available after the review period.

E.1 Computation Time Experiment §5.1

We show here additional details for our computation time experiments.

E.1.1 Hardware

The timing experiment was conducted on a 2022 Macbook Pro (personal laptop) with an M2 Pro Chip, 16GB RAM, and 10 cores.

E.1.2 Software Packages

- EBM, InterpretML (version 0.5.0): https://pypi.org/project/interpret/
- FLAM, flam (version 3.2): https://cran.r-project.org/web/packages/flam/index.html

We implement FAST in Julia (version 1.6.7). We will release our implementation open-source after the review period.

E.1.3 Datasets

In Table 1, we show the sources used for the computation time experiment.

Data	**Source**
Black Smoke ++	https://www.maths.ed.ac.uk/∼swood34/data/black_smoke.RData
Black Smoke	https://www.maths.ed.ac.uk/∼swood34/data/black_smoke.RData
Physiochemical	OpenML: 45661
Auto Horsepower	OpenML: 1192
Ailerons BNG	OpenML: 1207
Slice Localization	OpenML: 42973
Superconduct	OpenML: 44964
Scm1d	OpenML: 41485
Rf2	OpenML: 41484
Isolet	OpenML: 300

Table 1: Data sources for computation time experiment.

E.2 Feature Selection Experiment §5.2

We show here additional details for our feature selection experiment.

E.2.1 Software Packages

Competing algorithms:

- FLAM-GL (3.2.0): https://cran.r-project.org/web/packages/flam/index.html
- EBM-RS (0.5.0): https://pypi.org/project/interpret/
- ControlBurn: https://pypi.org/project/ControlBurn/
- FastSparseGAM (0.1.2): https://pypi.org/project/fastsparsegams/
- SAM (2.10-0): https://cran.r-project.org/web/packages/spam/index.html
- LASSO (1.4.0): https://pypi.org/project/scikit-learn/

We implement AGIS and group ℓ_0-FAST in Julia (version 1.6.7) and we will open-source our implementation after the review period.

E.2.2 Datasets

Data	**Source**
Ailerons	OpenML: 296
Elevators	OpenML: 217
Houses	OpenML: 537
SpaceGA	OpenML: 737
Boston	OpenML: 853
H16H	OpenML: 574
H8L	OpenML: 218
mtp	OpenML: 405
kin8nm	OpenML: 807
bank32nh	OpenML: 558
pol	OpenML: 201
fried	OpenML: 564
no2	OpenML: 547
rf1	OpenML: 41483
wind	OpenML: 503
scm1d	OpenML: 41485
scm20d	OpenML: 41486
stock	OpenML: 223
superconduct	OpenML: 44694
US crime	OpenML: 315

Table 2: Data sources for feature selection experiment.

F Feature Selection Experiment Heavy Tails

We show here some additional plots for our feature selection experiment.

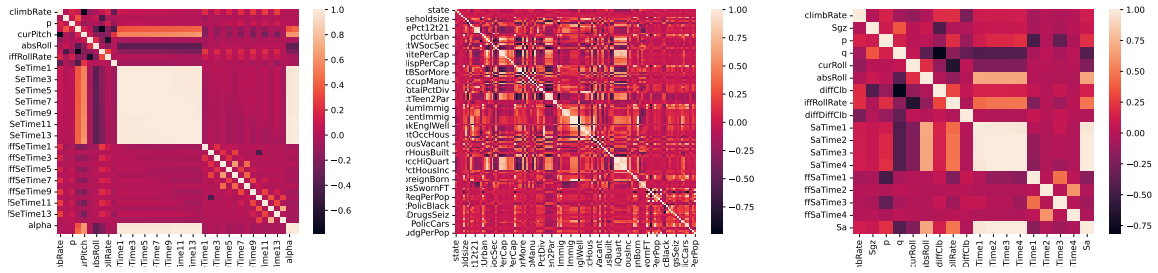

Figure 1: Our feature selection algorithms perform very well compared to our competing algorithms on datasets with highly correalted features. Correlation heat maps shown for Ailerons, US Crime, and Elevators datasets.

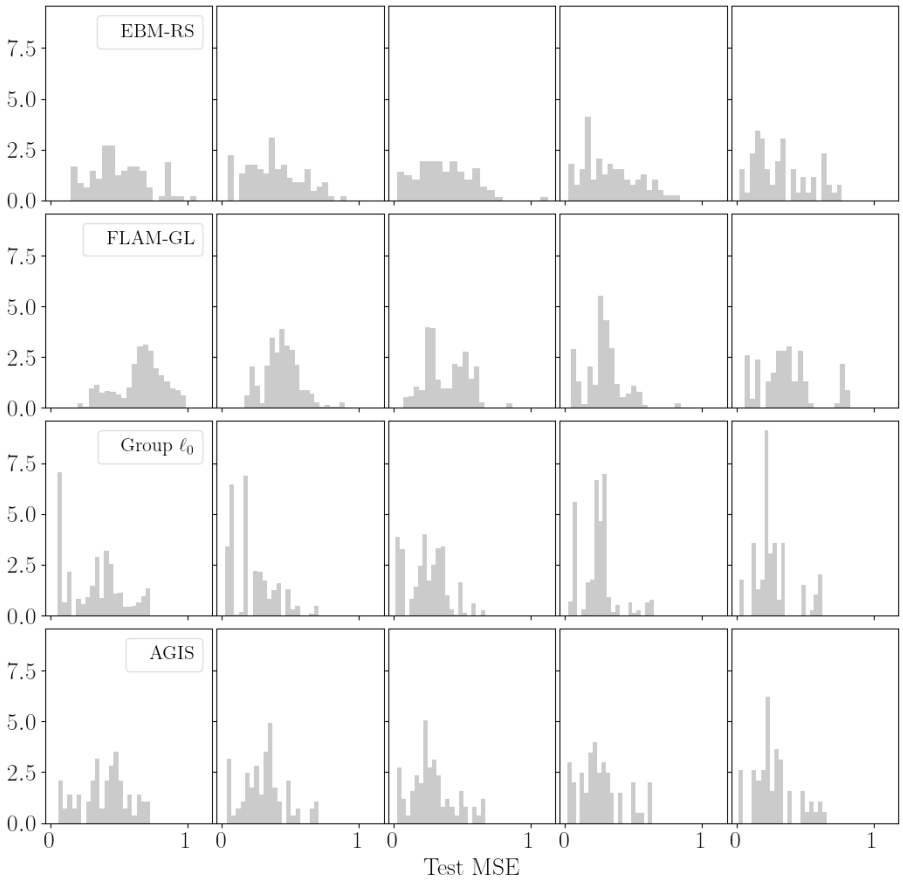

Figure 2: Histogram distribution of errors for our feature sparsity experiment. We see that EBM-RS and FLAM-GL have heavy right tails.

References

Sai Praneeth Karimireddy, Anastasia Koloskova, Sebastian U Stich, and Martin Jaggi. Efficient greedy coordinate descent for composite problems. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2887–2896. PMLR, 2019.

Tong Tong Wu and Kenneth Lange. Coordinate descent algorithms for lasso penalized regression. 2008.