Distribution Preserving Bayesian Coresets using Set Constraints

Shovik Guha
Department of Computer Science
University of Illinois at Urbana Champaign
Champaign, IL 61801
shovikg2@illinois.edu

Rajiv Khanna
Department of Statistics
University of California at Berkeley
Berkeley, CA 94720
rajivak@berkeley.edu

Sanmi Koyejo
Department of Computer Science
University of Illinois at Urbana Champaign
Champaign, IL 61801
sanmik@illinois.edu

Abstract

Bayesian coresets have become of increasing interest recently for providing a theoretically sound, scalable approach to Bayesian inference. In brief, a coreset is a (weighted) sample of a dataset that approximates the original dataset under some metric. Bayesian coresets specifically focus on approximations that approximate the posterior distribution. Unfortunately, existing Bayesian coreset approaches can significantly undersample minority subpopulations, leading to a lack of distributional robustness. As a remedy, this work extends existing Bayesian coresets from enforcing sparsity constraints to group-wise sparsity constraints. We explore how this approach helps to mitigate distributional vulnerability. We further generalize the group constraints to Bayesian coresets with matroid constraints, which may be of independent interest. We present an optimization analysis of the proposed approach, along with an empirical evaluation on benchmark datasets that support our claims.

1 Introduction

Bayesian coresets have become of interest recently in the artificial intelligence community for providing a theoretically sound, scalable approach to Bayesian inference. The main idea of a Bayesian coresets is to select a weighted subset of the original dataset such that the posterior inference using the weighted subset is a close approximation to the posterior inference using the entire dataset. If the desired cardinality of the subset is known beforehand, then we can formulate the task of choosing the subset as a constrained optimization problem, where the objective function measures the quality of the approximation of the subset, and the constraints enforce the cardinality of the subset is as desired. The sparsity constraints on the weighted subset are not convex, which poses a problem for exact optimization. Proposed solutions by Campbell & Broderick [3, 2] include a convex relaxation of the sparsity constraint to the l_1 norm so iterative schemes like Frank-Wolfe can be used in a blackbox fashion, and using a local greedy selection to build a sparse solution vector. Our work builds on is the result of Zhang et. al [7], which tackles the non-convex sparsity constraint directly via an iterative hard thresholding scheme devised by Blumensath and Davies [1].

All aforementioned previous works only consider a uniform sparsity constraint, i.e., selecting a subset of size k from the original dataset of size n. Regarding distribution shifts, uniform sparsity constraints have no knowledge of the distribution of the original dataset. This can lead to scenarios

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

where the sample distribution is very different from the distribution of the original dataset. For example, say the full dataset consists of disjoint groups with known variability across groups (e.g., demographic attributes such as age groups). In a data summarization task, one may want the relative proportions of these groups in the coreset to be roughly equal to the relative proportions of these groups in the original dataset. Still, the uniform sparsity constraint does not ensure this, and indeed in some instances, this can lead to undersampling of minority subgroups in the coreset. Experimental examples of this are given in section 5.

Contributions. More granular control over the selected elements in the representative subset is required to solve this sampling distribution shift issue. This is accomplished via more complex set constraints on the sampled subset. Specifically, we can encode the desired sample groups as a set of partition constraints (further details given in section 3). Solving the new optimization problem with the partition constraints is novel contribution. This work further generalizes this result to any sparsity constraint encoded as a matroid – proposing an efficient (weighted) selection procedure using iterated hard thresholding, which may be of independent interest. Finally, we show how the ability to generalize the sparsity constraints can help address sampling distribution shifts, with applications in algorithmic fairness.

2 Preliminaries

Given a probability mass function p, which depends on parameters θ, the liklihood of a random variable x is the probability x was sampled from p with parameters θ. We will assume we are given n samples, where $\mathcal{L}_i(\theta)$ is the log-liklihood of the i-th sample parameterized by θ. Assuming conditional independence with respect to θ, one can represent the log-likliehood of observing all samples as a summation of the individual log-liklihoods, i.e., $L(\theta) = \sum_{i=1}^{n} L_i(\theta)$. The goal of a Bayesian coreset is to approximate the full liklihood of all observations with a weighted sample, i.e., $L_w = \sum_{i=1}^{n} w_i L_i$, where $w \in R_+^n$ is a non-negative sparse vector. We will use $Dist(.\,\|\,.)$ to represent the distance functional that measures the deviation between L and L_w and we will mostly consider the case where $Dist(.\,\|\,.)$ is the 2-norm defined in the function space. In this setting, it is useful to view L, L_i, and L_w as functions in a Hilbert space. Additionally, we will use the notation $[n]$ as a shorthand for $\{1, 2, 3, ..., n\}$, the set of integers from 1 to n.

2.1 Existing Approach: Standard Sparsity Problem

As defined by [7], the sparsity constrained Bayesian Coreset problem can be formulated as follows

$$\underset{w \in R^n}{\operatorname{argmin}}\, f(w) := Dist(L, L_w)$$

$$\text{subject to} \quad ||w||_0 \leq k$$

Where $L = \sum_{i=1}^{n} L_i$, i.e., the sum of the log likelihoods of our n samples and $L_w = \sum_{i=1}^{n} w_i L_i$, the likelihood of the sample with respect to w. Note the k selected points need not be representative of the distribution of full dataset, which leads to the sampling induced distribution shifts. We will refer to this problem formulation as the “standard sparsity problem” for the remainder of this work.

2.2 Finite Sample Estimation

Given that $Dist(.\,\|\,.)$ is the 2-norm defined in the function space, the objective function can be expanded as follows

$$Dist(L, L_w)^2 = \|L - L_w\|_{\pi,2}^2 = \mathbb{E}_{\theta\sim\pi}\left[(L(\theta) - L_w(\theta))^2\right]$$

Where π is the distribution of the parameters θ. To compute this expectation exactly, one would need to integrate over all possible parameter settings, which is clearly intractable. Luckily, the expectation can be approximated by a finite dimensional L_2 norm by replacing the functions L and L_w with vectors of sampled evaluations, essentially computing a Monte Carlo approximation of the expectation. Further details are given in Zhang et. al [7].

2.3 Solution to Standard Sparsity Problem

Notice that in both problem formulations, the sparsity constraint causes the overall optimization problem to become non-convex. Therefore, it is not unreasonable to shift one’s focus from an analytical solution to an approximation computed via some iterative method. This is the solution presented by Zhang et. al [7], where an iterative approach to approximate the optimal vector w is proposed. Specifically, the authors adapt accelerated iterative hard thresholding (IHT) schemes to the Bayesian coreset problem. The classical IHT algorithm of Blumensath and Davies [1] is given below

```
Algorithm 1: Vanilla IHT
Input: Objective $f : R^n \to R$; sparsity $k$, step size $\mu$
1 Initialize $w$
2 repeat
3     $w \leftarrow \Pi_{C_k \cap R^+}(w - \mu \nabla f(w))$;
4 until stop criteria
```

Zhang et. al [7] modify the vanilla IHT algorithm by adding a line search for picking the step size μ in each iteration, as well as adding a momentum term to accelerate the rate of convergence, but these changes are more directed at improving the empirical performance of the algorithm, theoretically applying the vanilla IHT algorithm to the standard sparsity problem produces an arbitrarily good approximation, under certain regularity conditions.

3 Proposed Approach: Group Sparsity Problem

To make our sampling more robust to distribution shifts across groups, we encode the distribution of the full dataset as a set of partition constraints on the set of feasible samples. Specifically, let us consider the set of input samples D to be a collection of m disjoint sets, such that $D = \bigcup_{i=1}^{m} D_m$ and $D_i \bigcap D_j = \emptyset \ \forall\, i, j \in [m]$. Since we want our sample of to be representative of this initial structure to avoid a sample induced distribution shift, we add some additional constraints to ensure the sample does not exceed a certain number of points from each subset. Therefore we can modify the original constrained optimization problem to the following, which will be the main focus of this work.

$$\begin{aligned} &\underset{w \in R^n}{\operatorname{argmin}} f(w) := Dist(L, L_w) \\ &\text{subject to} \quad ||w_j||_0 \le k_j \ \ \forall j \in [m] \\ &\qquad\qquad w_i \ge 0 \qquad \forall i \end{aligned}$$

Where $w_j \in R^{|D_j|}$, and w is the concatenation of $\{w_1, w_2, ..., w_m\}$. In other words w_j is the vector that corresponds to the sampling of points from D_j. Also, note that we are assuming $\sum_{j=1}^{m} k_j = k$, so the constraint in the original formulation of $||w||_0 \le k$ is redundant since $||w_j||_0 \le k_j \ \ \forall j \in [m]$ implies the former constraint. We will refer to this problem formulation as the "group sparsity problem" for the remainder of this work.

3.1 Projection Step

A key step in the IHT algorithm is the projection step (step 3 of Algorithm 1), in which the current iterate is projected onto the subspace spanned by the constraint. We show that for the case of group sparsity constraints, the projection step can be computed exactly in polynomial time. Further details and proof are given in the Appendix A, and the projection is computed by greedily selecting the largest non-negative k_j entries for each $j \in [m]$.

3.2 Matroid Extension

We will now consider more general setting where we can encode our constraints as a matroid (N, E), where N is the ground set of indices corresponding to samples, and E are the subsets of indices which satisfy the given constraint. For example, in the original bayesian coreset problem on n samples, $N = [n]$ and $E = \{S \subset N \ \ s.t \ \ |S| \le k\}$, i.e., the index set corresponding to the subsets of samples we could choose to satisfy the sparsity constraint. For the special case of the original bayesian coreset problem, the resulting matroid corresponds to the uniform matroid, and in the aforementioned new problem setting where we have some initial structure from the input we would like to consider, the resulting matroid is the partition matroid. Further are details and proofs are given in the Appendix B, and a proof of convergence is given in the Appendix C.

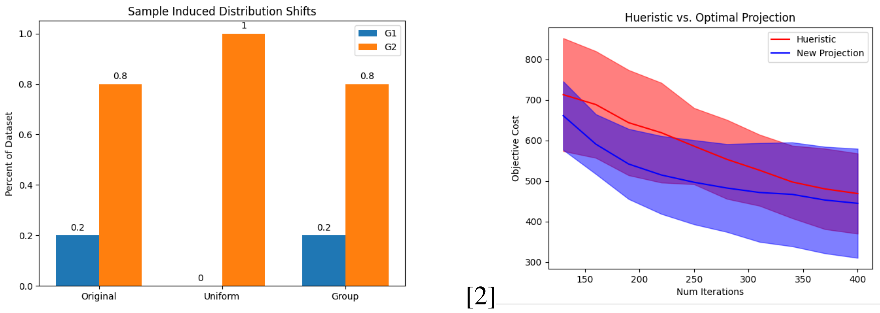
Lemma 3.1. *Given a matroid constraint (N, E), where N is a ground set of indices corresponding to input samples, in the case of the Euclidean distance metric, the projection step of IHT can be computed exactly in polynomial time via a greedy selection algorithm.*

4 Connections Between Algorithmic Fairness and Distribution Shifts

Characterizing what exactly it means for an algorithm to be fair is an open and active area of research without a general consensus, and in general a notion of fairness that makes sense in one situation may not make sense in another. The notion of fairness we employ is the notion of *balance* [4]. Say we have an initial dataset D of size n, and a sensitive feature with c possible value settings. For all $i \in c$, let D_i be the elements from the dataset which have a value of i for the sensitive feature. We will assume $D_i \cap D_j = \emptyset \, \forall \, i, j \in \{c \times c\}$, i.e., partitioning the dataset based on the sensitive feature produces disjoint sets. We say a sample of data points S is balanced with respect to a sensitive feature if $\left\lfloor \frac{|D_i|}{n} \right\rfloor \leq \frac{|S \cap D_i|}{|S|} \leq \left\lceil \frac{|D_i|}{n} \right\rceil$ holds for all $i \in c$. In other words, the relative proportions of the data with respect to the sensitive feature are preserved. It is easy to see that if the distribution of the sample with respect to the sensitive feature is not preserved, then the resulting sample S is not balanced, and thus over represents certain subgroups, while under representing others. Therefore, distribution robust sampling is inherently tied to algorithmic fairness via the notion of balance, and by using the distribution preserving sampling methods proposed here, one can prevent bias in sampling induced by distribution shifts.

5 Experimental Results

We now move our attention to empirical performance metrics of our proposed algorithm. The first experiment highlights a failure case using just the original uniform sparsity constraint, which elucidates the distribution shifts that can be caused by such constraints. We construct a synthetic dataset of size $N = 1000$, partitioned into 2 groups denoted by the orange and blue bars. Exact details on the construction of the synthetic dataset are given in the Appendix D. The left set of bars represent the true distribution of the dataset, the middle set of bars are the distribution of the sample selected using the uniform sparsity constraint, and the right set of bars represent the distribution of the sample selected using the group sparsity constraint. Note that the distribution of the sample selected using the group sparsity constraint matches the true distribution exactly, as the group sparsity constraints allow us to control the sample distribution exactly. The results shown are averaged across 10 trials and it is important to note that across all trials, none of the data in group 1 was selected to be the sample using only the uniform sparsity constraint.



[1] [2]

Figure 1: (Left) Comparison of true distribution, the distribution of the sample with uniform constraints, and the distribution of the sample with group sparsity constraints, $N = 1000$, $K = 100$ averaged across 10 trials. (Right) Comparison of objective value achieved by the hueristic approach with the objective value achieved by out proposed approach, across 400 iterations, averaged across 10 trials, shaded regions represent the standard deviations

One might think of a hueristic method to approximate the solution to the group sparsity problem by solving the uniform sparse selection for each group separately, and then concatenating the results to construct the final sample - in fact this hueristic is widely used in practice when trying to enforce group sparsity constraints. We show that our algorithm empirically outperforms the huerisitic method by finding a solution with a lower objective value in the same number of iterations. The red (blue) line represents the objective value of the hueristic (proposed algorithm) across iterations respectively, and shaded regions represent the standard deviations.

6 Conclusion

In this work we have highlighted a particular type of distribution shift caused by sampling, and shown that these types of distribution shifts relate to existing notion of algorithmic fairness. We present a solution to the problem of sampling induced distribution shifts and show the algorithm is theoretically sound, and performs well in practice. This is an ongoing work, and we hope to show empirical results on real-world datasets, as well as comparisons with other methods such as [5, 6] which also claim to improve group robustness, with (in our case) applications to fairness.

References

[1] Blumensath, T., and Davies, M. E. Iterative hard thresholding for compressed sensing. *CoRR abs/0805.0510* (2008).

[2] Campbell, T., and Broderick, T. Bayesian coreset construction via greedy iterative geodesic ascent, 2018.

[3] Campbell, T., and Broderick, T. Automated scalable bayesian inference via hilbert coresets, 2019.

[4] Halabi, M. E., Mitrovic, S., Norouzi-Fard, A., Tardos, J., and Tarnawski, J. Fairness in streaming submodular maximization: Algorithms and hardness. *CoRR abs/2010.07431* (2020).

[5] Khani, F., and Liang, P. Removing spurious features can hurt accuracy and affect groups disproportionately. *CoRR abs/2012.04104* (2020).

[6] Liu, E. Z., Haghgoo, B., Chen, A. S., Raghunathan, A., Koh, P. W., Sagawa, S., Liang, P., and Finn, C. Just train twice: Improving group robustness without training group information. *CoRR abs/2107.09044* (2021).

[7] Zhang, J. Y., Khanna, R., Kyrillidis, A., and Koyejo, O. Bayesian coresets: An optimization perspective, 2020.

[8] Zhang, J. Y., Khanna, R., Kyrillidis, A., and Koyejo, O. Learning sparse distributions using iterative hard thresholding, 2020.

Appendix

A Projection Proof

Proof: A key step in the IHT algorithm is the projection step, in which the current iterate is projected onto the subspace spanned by the constraint. Here we prove that for the case of group sparsity constraints, the projection onto the subspace spanned by the constraints can be computed exactly in polynomial time. Let us denote this subspace as C_k. Formally, let $W_j = \{w_j \;\; s.t \;\; \|w_j\|_0 \leq k_j\}$. Then we can write $C_k = [w_1 \in W_1, w_2 \in W_2, \ldots, w_m \in W_m]$, where w_i is an arbitrary vector from the set W_i. Like the case in [7], we have the additional positivity constraint, so the subspace spanned by all of our constraints is actually $(C_k \cap R^n_+)$. Unlike the case in [8], computing $\Pi_{C_k \cap R^n_+}(w)$, the projection of w onto the subspace spanned by the group sparsity constraints, is computationally easy in the case of the Euclidean distance metric. Take an arbitrary vector $w' \in (C_k \cap R^n_+)$, and let $S = \text{supp}(w')$, i.e., the set of non-zero indices of w'. We can write $\|w' - w\|_2^2 = \|w\| - \sum_{i \in S} w_i^2$, so to minimize this distance, we must maximize $\sum_{i \in S} w_i^2$, which can be done by simply picking S to correspond to the largest non-negative k_j entries for each $j \in [m]$.

B Matroid Extension

We will now consider more general setting where we can encode our constraints as a matroid (N, E), where N is the ground set of indices corresponding to samples, and E are the subsets of indices which satisfy the given constraint. For example, in the original bayesian coreset problem on n samples, $N = [n]$ and $E = \{S \subset N \;\; s.t \;\; |S| \leq k\}$, i.e., the index set corresponding to the subsets of samples we could choose to satisfy the sparsity constraint. For the special case of the original bayesian coreset problem, the resulting matroid corresponds to the uniform matroid, and in the aforementioned new problem setting where we have some initial structure from the input we would like to consider, the resulting matroid is the partition matroid.

B.0.1 Proof of Lemma 3.1

Proof: In our setting, given a matroid (N, E), we will consider the ground set $N = [n]$, where n is the total number of samples, and thus E is a set of subsets of indices, which correspond to input samples which satisfy some notion of independence. Consider an arbitrary vector w in the ambient space, an arbitrary constraint $e \in E$, and a vector w' which is the projection of w onto the constraint denoted by e, so that $\text{supp}(w') = e$. Since e denotes some sparsity constraint, we know the optimal projection can be computed as $w'_i = w_i$ if $i \in e$ and $w_i \geq 0$, else $w'_i = 0$. Under the Euclidean metric, we can compute the projection cost as $\|w' - w\|_2^2 = \|w\|_2^2 - \sum_{i \in e} w_i^2$, and since w is fixed, in order to minimize the cost of the projection, we must maximize the second term in the projection cost. Therefore, the projection step can be formulated as $\max_{e \in E} \sum_{i \in e} w_i^2$. In this form it is easy to see that the projection step reduces to maximizing a modular function subject to a matroid constraint. It is known that the solution can be computed exactly via a greedy selection algorithm, since we are simply looking for the maximum cost base of the matroid. We simply choose the index which gives the maximum gain in the objective function, while still satisfying the independence constraints, and continue until we have a maximal base. Note that when our matroid is the uniform matroid, the greedy algorithm for choosing the maximum cost base reduces to the algorithm for the IHT projection step given in [7]. The same is true for the case of the partition matroid.

C Convergence Analysis of IHT with Matroid constraints

For the purposes of this analysis, we will not consider the use of any momentum terms, i.e., the step size μ is constant. The IHT algorithm in our setting, without any momentum terms is given below

Algorithm 2: Vanilla IHT

```
Input :Objective $f : R^n \to R$; sparsity $k$, step size $\mu$
Initialize $w$
repeat
    $w \leftarrow \Pi_{C_k \cap R^+}(w - \mu \nabla f(w))$;
until stop criteria
```

To begin the analysis we start with necessary definitions

Definition C.0.1 (Restricted Isometry Property)**.** *For each integer $s \in \mathbb{N}$, the isometry constant δ_s of a matrix Φ is the smallest number such that*

$$(1-\delta_s)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta_s)\|x\|_2^2$$

for all s-sparse vectors x

In other words, Φ is close to an isometric transformation for sparse vectors.

Another useful observation is that for any set $S \subseteq [n]$, the projection operator Π_S is a selection matrix, i.e., we can write

$$\Pi_S = \{diag(\delta_i)\}_{i=1}^n$$

where δ_i is an indicator function which equals 1 if $i \in S$, and 0 otherwise. This holds regardless of the specific constraints, which is useful since the constraint matroid represents a family of constraints.

Lemma C.1. *If Φ satisfies the RIP assumption, then given a set $S \subseteq [N]$ with $|S| \leq k$, then $\forall w \in R^n$, the following holds*

$$\alpha_k\|\Pi_S w\|_2 \leq \|\Pi_S\Phi^T\Phi\Pi_S w\|_2 \leq \beta_k\|\Pi_S w\|_2$$

Proof: Recall that Π_S is a projection operator onto the index set S, so if $|S| \leq k$, then $\Pi_S w$ has sparsity k for all $w \in R^n$. Given $\Phi \in \mathcal{R}^{m\times n}$ satisfies the RIP assumption, we have

$$\alpha_k\|\Pi_S w\|_2^2 \leq \|\Phi\Pi_S w\|_2^2 \leq \beta_k\|\Pi_S w\|_2^2$$

Denoting $b = \Phi\Pi_S w$, $\mathcal{X} = \{x \in R^n \; s.t \; \|x\|_2 = 1\}$, and $\langle .,. \rangle$ as the Euclidean inner product we can write

$$\begin{aligned}\|\Pi_S\Phi^T b\|_2^2 &= \max_{x\in\mathcal{X}}(\langle\Pi_S\Phi^T b, x\rangle)^2\\ &= \max_{x\in\mathcal{X}}(b^T\Phi\Pi_S x)^2\\ &= \max_{x\in\mathcal{X}}(, \Phi\Pi_S x\rangle)^2\\ &= \max_{x\in\mathcal{X}}(\langle\Phi\Pi_S w, \Phi\Pi_S x\rangle)^2\end{aligned}$$

Where the second equality holds because Π_S is a symmetric matrix, and the fact that $, b\rangle = a^T b$. Let x^* be the optimal value which maximizes dot product. Using the Cauchy-Schwartz inequality, we can show an upper bound

$$\max_{x\in\mathcal{X}}(\langle\Phi\Pi_S w, \Phi\Pi_S x\rangle)^2 = (\langle\Phi\Pi_S w, \Phi\Pi_S x^*\rangle)^2 \leq \|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x^*\|_2^2$$

We can also show a lower bound by computing the dot product for a specific value of x. In particular, let $x' = \frac{\Phi\Pi_S w}{\|\Phi\Pi_S w\|_2}$, the normalized form of the vector $\Phi\Pi_S w$.

$$\max_{x\in\mathcal{X}}(\langle\Phi\Pi_S w, \Phi\Pi_S x\rangle)^2 \geq (\langle\Phi\Pi_S w, \Phi\Pi_S x'\rangle)^2 = \|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x'\|_2^2$$

Where the last equality is due to the fact that $\Phi\Pi_S w$ and $\Phi\Pi_S x'$ are parallel. By the bounds assumed by RIP we get

$$\|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x^*\|_2^2 \leq \beta_k\|\Pi_S w\|_2^2 \times \beta_k\|\Pi_S x^*\|_2^2$$

$$\|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x'\|_2^2 \geq \alpha_k\|\Pi_S w\|_2^2 \times \alpha_k\|\Pi_S x'\|_2^2$$

Recall that x^* is a unit vector, and Π_S modifies vectors by setting specific indices to 0, and leaving the rest of the vector unchanged. Thus $\|\Pi_s x^*\|_2^2 \leq \|x^*\|_2^2 \leq 1$. Also recall that $x' = \frac{\Phi\Pi_S w}{\|\Phi\Pi_S w\|_2}$, so x' is already sparse. Thus $\|\Pi_S x'\|_2^2 = \|x'\|_2^2 = 1$. This implies

$$\|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x^*\|_2^2 \le \beta_k^2\|\Pi_S w\|_2^2$$

$$\|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x'\|_2^2 \ge \alpha_k^2\|\Pi_S w\|_2^2$$

Note that $\|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x'\|_2^2 \le \|\Phi\Pi_S w\|_2^2 \times \|\Phi\Pi_S x *\|_2^2$ by the optimality of x^*. Putting the inequalities together, and taking the square root gives the desired bound

$$\alpha_k\|\Pi_S w\|_2 \le \|\Pi_S\Phi^T\Phi\Pi_S w\|_2 \le \beta_k\|\Pi_S w\|_2$$

This gives us a bound on the eigenvalues of $\Pi_S\Phi^T\Phi\Pi_S$, which we will need need in the proof of the iterative invariant bound.

Lemma C.2. *Given Φ which satisfies the RIP assumption, $S_1, S_2 \subseteq [n]$ such that $|S_1 \cup S_2| \le k$, then $\forall w \in \mathbb{R}^n$, the following inequality holds*

$$\|\Pi_{S_1}\Phi^T\Phi\Pi_{S_1^c}\Pi_{S_2}w\|_2 \le \frac{\beta_k - \alpha_k}{2}\|\Pi_{S_2}w\|_2$$

Similar to the proof of the previous lemma, we will begin by rewriting the norm as an inner product. Let $\mathcal{X} = \{x \in R^n \; s.t \; \|x\|_2 = 1\}$

$$\begin{aligned}\|\Pi_{S_1}\Phi^T\Phi\Pi_{S_1^c}\Pi_{S_2}w\|_2 &= \max_{b\in\mathcal{X}}\langle\Pi_{S_1}\Phi^T\Phi\Pi_{S_1^c}\Pi_{S_2}w, b\rangle \\ &= \max_{b\in\mathcal{X}}\langle\Phi\Pi_{S_1^c}\Pi_{S_2}w, \Phi\Pi_{S_1}b\rangle\end{aligned}$$

which is a valid equality since Π_{S_1} is symmetric. Note that we are examining the dot product of 2 vectors projected into the subspace of Φ. Let us define 2 normalized vectors, which will denote the vectors before their projection by into the subspace defined by Φ.

$$X = \frac{\Pi_{S_1^c}\Pi_{S_2}w}{\|\Pi_{S_1^c}\Pi_{S_2}w\|_2} \quad Y = \frac{\Pi_{S_1}b}{\|\Pi_{S_1}b\|_2}$$

Since $\Pi_{S_1^c}$ and Π_{S_1} are completely disjoint, we have $, Y\rangle = 0$, which implies that $\|X + Y\| = \|X\| + \|Y\| = 2$. Since $|S_1 \cup S_2|\langle k$, we know $X + Y$ is a k-sparse vector. Using the RIP assumption yields

$$2\alpha_k = 2\|X + Y\|_2 \le \|\Phi X + \Phi Y\|_2 \le \beta_k\|X + Y\|_2 = 2\beta_k$$

Symmetrically, we have $\|X - Y\|_2 = 2$ and $X - Y$ is also k-sparse, which yields

$$2\alpha_k \le \|\Phi X - \Phi Y\| \le 2\beta_k$$

As a generalization of the fact that $ab = \frac{(a+b)^2-(a-b)^2}{4}$ we have

$$\langle\Phi X, \Phi Y\rangle = \frac{\|\Phi X + \Phi Y\|_2^2 - \|\Phi X - \Phi Y\|_2^2}{4}$$

Plugging in the previous inequalities we have

$$-\frac{\beta_k - \alpha_k}{2} \quad \le \langle\Phi X, \Phi Y\rangle \quad \le \frac{\beta_k - \alpha_k}{2}$$

Relating back to our original quantity

$$\begin{aligned}\|\Pi_{S_1}\Phi^T\Phi\Pi_{S_1^c}\Pi_{S_2}w\|_2 &\le \max_{b\in\mathcal{X}}\langle\Phi X, \Phi Y\rangle \times \|\Phi\Pi_{S_1^c}\Pi_{S_2}w\|_2 \times \|\Phi\Pi_{S_1}b\|_2 \\ &\le \frac{\beta_k - \alpha_k}{2} \times \|\Pi_{S_1^c}\Pi_{S_2}w\|_2 \\ &\le \frac{\beta_k - \alpha_k}{2} \times \|\Pi_{S_2}w\|_2\end{aligned}$$

Where the first inequality is due to the fact that $\|b\|_2 = 1$, and the projection by Π_{S_1} can only decrease 2 norm value, and likewise for $\Pi_{S_1^c}$.

Lemma C.3. *Using the Vanilla IHT algorithm, the following iterative invariant holds*

$$\|w_{t+1} - w^*\|_2 \leq \rho \|w_t - w^*\|_2 + 2\beta_{3k}\sqrt{\beta_{2k}}\|\epsilon\|_2$$

where w_t is the iterate at time step t, w^* is the optimal value, $\rho = (2\max\{2\mu\beta_{2k} - 1, 1 - 2\mu\alpha_{2k}\} + 4\mu\frac{\beta_{4k}-\alpha_{4k}}{2})$, β_i is the isometry constant associated with i-sparse vectors

Proof: Let $v = w_t - \mu\nabla f(w_t)$ and $S_\star = supp(w_{t+1}) \cup supp(w^*)$. Applying the triangle inequality yields

$$\|w_{t+1} - w^*\|_2 \leq \|w_{t+1} - \Pi_{S_\star} v\|_2 + \|\Pi_{S_\star} v \ - w^*|_2$$

Notice, by construction, ${}_{t+1}, S_\star^c\rangle =^*, S_\star^c\rangle = 0$ Focusing on the first term

$$\begin{gathered}
\|w_{t+1} - \Pi_{S_\star} v\|_2^2 = \|w_{t+1} - v + \Pi_{S_\star^c} v\|_2^2 \\
= \|w_{t+1} - v\|_2^2 + \|\Pi_{S_\star^c} v\| + 2{}_{t+1} - v, \Pi_{S_\star^c} v\rangle \\
= \|w_{t+1} - v\|_2^2 + \|\Pi_{S_\star^c} v\| + 2\langle -v, \Pi_{S_\star^c} v\rangle \\
\leq \|w^* - v\|_2^2 + \|\Pi_{S_\star^c} v\| + 2\langle -v, \Pi_{S_\star^c} v\rangle \\
= \|w^* - v\|_2^2 + \|\Pi_{S_\star^c} v\| + 2^* - v, \Pi_{S_\star^c} v\rangle \\
= \|w^* - v + \Pi_{S_\star^c} v\|_2^2 \\
= \|w^* - \Pi_{S^*} v\|_2^2
\end{gathered}$$

Where the first equality is due to the equivalence of subtracting values from some indices $i \in S$, and subtracting all values and adding back indices $i \in S^c$, and the inequality is due to the projection $w^* \in C_k \cap R^+$ and the fact that the projection $w_{t+1} = \Pi_{C_k \cap R^+} v$ is done optimally. Substituting our the first term with this inequality yields

$$\|w_{t+1} - w^*\|_2 \leq 2\|\Pi_{S_\star} v \ - w^*|_2$$

Expanding v and denoting the optimal error $\epsilon = \Phi w^* - y$

$$\begin{gathered}
v = w_t - \mu\nabla f(w_t) \\
= w_t - \mu(2\Phi^T(\Phi w_t - y)) \\
= w_t - \mu(2\Phi^T\Phi(w_t - w^*) + 2\Phi^T(\Phi w^* - y)) \\
= w_t - 2\mu\Phi^T\Phi(w_t - w^*) - 2\mu\Phi^T\epsilon
\end{gathered}$$

Where the second equality is due to the expansion of $\nabla f(w_t)$, and the third equality is due to the equivalence between the total error in our current estimate and the error from our current estimate to the optimal value w^*, added to the optimal error. Substituting the expansion of v into the previous inequality yields

$$\begin{gathered}
\|w_{t+1} - w^*\|_2 \leq 2\|\Pi_{S_\star}(w_t - 2\mu\Phi^T\Phi(w_t - w^*) - 2\mu\Phi^T\epsilon) - w^*\|_2 \\
= 2\Big\|\Pi_{S_\star}(w_t - w^*) - 2\mu\Pi_{S^*}\Phi^T\Phi(w_t - w^*) - 2\mu\Phi^T\epsilon\Big)\Big\|_2 \\
\leq 2\Big\|\Pi_{S_\star}(w_t - w^*) - 2\mu\Pi_{S^*}\Phi^T\Phi(w_t - w^*) + 4\mu\Phi^T\epsilon\Big)\Big\|_2 \\
= 2\|\Pi_{S_\star}(w_t - w^*) - 2\mu\Pi_{S^*}\Phi^T\Phi I(w_t - w^*) + 4\mu\Phi^T\epsilon)\|_2
\end{gathered}$$

Where the first equality is due to the rearrangement of terms and the fact that $\Pi_{S^*} w^* = w^*$. Expanding the identity matrix as $I = \Pi_{S_*} + \Pi_{S_*^c}$ yields

$$\begin{gathered}
\|w_{t+1} - w^*\|_2 \leq 2\|(I - 2\mu\Pi_{S_*}\Phi^T\Phi\Pi_{S_*})\Pi_{S_*}(w_t - w^*)\|_2 \\
+4\mu\|\Pi_{S_*}\Phi^T\Phi\Pi_{S_*^c}(w_t - w^*)\|_2 \\
+4\mu\|\Pi_{S_*}\Phi^T\epsilon\|_2)
\end{gathered}$$

We will now use the previous lemmas to bound each of these terms. Let us begin with the first term $2\|(I - 2\mu\Pi_{S_*}\Phi^T\Phi\Pi_{S_*})\Pi_{S_*}(w_t - w^*)\|_2$. Let $\lambda(A)$ denote the eigenvalues of a matrix A. We know $S_* \leq 2k$, which allows us to apply Lemma 3.1, which yields

$$\alpha_{2k} \leq \lambda(\Pi_{S_*}\Phi^T\Phi\Pi_{S_*}) \leq \beta_{2k}$$

which implies

$$1 - 2\mu\beta_{2k} \leq \lambda(I - 2\mu\Pi_{S_*}\Phi^T\Phi\Pi_{S_*}) \leq 1 - 2\mu\alpha_{2k}$$

which further implies

$$\begin{aligned} 2\|(I - 2\mu\Pi_{S_*}\Phi^T\Phi\Pi_{S_*})\Pi_{S_*}(w_t - w^*)\|_2 &\leq 2\max\{2\mu\beta_{2k} - 1, 1 - 2\mu\alpha_{2k}\}\|\Pi_{S_*}(w_t - w^*)\| \\ &\leq 2\max\{2\mu\beta_{2k} - 1, 1 - 2\mu\alpha_{2k}\}\|(w_t - w^*)\| \end{aligned}$$

To analyze the second term, let us denote $S' = supp(w_t) \cup supp(w^*)$. We can then write the second term as $4\mu\|\Pi_{S_*}\Phi^T\Phi\Pi_{S_*^c}\Pi_{S'}(w_t - w^*)\|_2$, and given the fact that $|S' \cup S_*| \leq 4k$, we can apply Lemma 3.2 directly which yields

$$\begin{aligned} 4\mu\|\Pi_{S_*}\Phi^T\Phi\Pi_{S_*^c}\Pi_{S'}(w_t - w^*)\|_2 &\leq 4\mu\frac{\beta_{4k} - \alpha_{4k}}{2}\|\Pi_{S'}(w_t - w^*)\|_2 \\ &\leq 4\mu\frac{\beta_{4k} - \alpha_{4k}}{2}\|(w_t - w^*)\|_2 \end{aligned}$$

Finally, we can bound the third term as follows, where $\mathcal{X} = \{x \in R^n \mid \|\|x\|_2 = 1\}$

$$\begin{aligned} \|\Pi_{S_*}\Phi^T\epsilon\|_2 &= \max_{x\in\mathcal{X}}\langle\Pi_{S_*}\Phi^T\epsilon, x\rangle \\ &= \max_{x\in\mathcal{X}}\epsilon^T\Phi\Pi_{S_*}x \\ &= \max_{x\in\mathcal{X}}\langle\epsilon, \Phi\Pi_{S_*}x\rangle \\ &\leq \max_{x\in\mathcal{X}}\|\epsilon\|_2\|\Phi\Pi_{S_*}x\|_2 \\ &\leq \sqrt{\beta_{2k}}\|\epsilon\|_2 \end{aligned}$$

Combining the above inequalities yields the following iterative invariant

$$\begin{aligned} \|w_{t+1} - w^*\|_2 \leq 2\max\{2\mu\beta_{2k} - 1, 1 - 2\mu\alpha_{2k}\}\|(w_t - w^*)\| \\ +4\mu\frac{\beta_{4k} - \alpha_{4k}}{2}\|(w_t - w^*)\|_2 \\ +4\mu\sqrt{\beta_{2k}}\|\epsilon\|_2 \end{aligned}$$

D Experiment Details

For the first experiment, construct a synthetic dataset with $N = 1000$ samples with $M = 400$ features each, such that each feature value is sampled from a univariate Gaussian distribution. We construct the dataset such that 80% of the input vectors contain features sampled from some univariate Gaussian distribution with mean μ_1, and standard deviation σ_1, and the remaining 20% of the data contains features sampled from a different univariate Gaussian distribution with mean μ_2, and standard deviation σ_2, such that $c\mu_2 = \mu_1$ and $c^2\sigma_2 = \sigma_1$. Essentially 80% of the data has features that are sampled from some Gaussian distribution, and the remaining 20% of the data has features sampled from a scaled down version of the same Gaussian distribution.

In the second experiment, we observe the cost of the objective function using the hueristic solution to the group sparsity problem across iterations, and compare this to the cost of the objective using our proposed algorithm. We take the average across 10 trials, and the mean objective cost of approach is represented by the solid lines, and the shaded regions represent the standard deviations. Note that while the hueristic approach does produce a feasible solution to the group sparsity problem, the points selected to be in the sample differ from the points selected by our proposed algorithm. In the future, we hope to show empirically that the sample points selected by our algorithm are qualitatively superior to the sample points selected by the hueristic. This intuitively makes sense as our algorithm takes into account the full posterior, while the huerisitic only considers the group-wise partioned posterior