Summary and discussion of: “A Fast Coordinate Descent Method for High-Dimensional Non-Negative Least Squares using a Unified Sparse Regression Framework” [*]

WONG, Yiu Kwan

15/12/2024

Contents

1 Overview of the paper **1**

1.1 Why is it Interesting 1

1.2 Challenges 2

1.3 Prposed methods 2

2 Result and Discussion **6**

2.1 Dataset 6

2.2 Result 6

3 Conclusion **8**

1 Overview of the paper

1.1 Why is it Interesting

This paper is interesting because it introduces a unifying framework that connects several well-known methods, such as NNLS, BVLS, and Lasso, under the general problem – polyhedron constrained least squares. In particularly, the paper shows that the problem admits locally

[*] The Latex document and the reproduced code can be found in `https://github.com/ykwongbb/MATH5472`

unique sparse solution in high dimensions. Also, the paper gives both theoretical understanding and applications on NNLS, BVLS, Lasso, etc. of these constrained regression problems. Moreover, the paper introduces a novel algorithm based on coordinate descent, which shown to be at least a 5x speed-up from the state-of-the-art solvers. In summary, the paper is interesting as it addresses both the theoretical aspect and the algorithmic aspects of constrained sparsed regression problems in high-dimensional spaces, which are important in statistical machine learning.

1.2 Challenges

High Dimensionality

In modern applications, the number of features p can be much larger than the number of samples n, leading to a high-dimensional setting. However, traditional methods only performs well when n is relatively large to p. This makes the solutions of the standard least squares become ill-posed or overfitted.

Local Uniqueness

In the context of high-dimensionality, solutions to constrained least squares problems are usually not unique and only locally unique in some subspaces. It is non-trivial to identify the necessary conditions for a subspace defined by the binding constraints.

Constraints

The coefficients in different problems are usually by some constraints, such as non-negativity or certain bounds. This increases the computational complexity of the optimization problem and makes it more difficult to find a general solution that can handle different constraints.

1.3 Prposed methods

Definition 3.1 A **polyhedron** is defined as

$$\mathscr{P} \equiv \{x \in \mathbb{R}^p : Ax \leq b\} \tag{1}$$

where $A \in \mathbb{R}^{m \times p}$ and $b \in \mathbb{R}^m$. An element $x \in \mathscr{P}$ binds the i-th constraint if $a_i^T x = b_i$. We say x binds k constraints if there exists a set $S \subseteq [m] = \{1, \cdots, m\}$ of size k such that x binds exactly these k constraints for the i-th constraint for every $i \in S$. Equivalently, we can express $\mathscr{P}$ as

$$\mathscr{P} \equiv \{x \in \mathbb{R}^p : A_{-S}x \leq b_{-S}, A_S x = b_S\} \tag{2}$$

where $S = \{i \in [m] : a_i^T x = b_i, \forall x \in \mathscr{P}\}$, A_S are the rows of A indexed by S and A_{-S} are the rows of A indexed by $[m] \setminus S$. As such, we have a non-empty interior $\{A_{-S}x \leq b_{-S}\}$ and a uniquely determined affine space $\{A_S x = b_S\}$. If $\mathscr{P}$ is non-empty, then we denote the dimension of $\mathscr{P}$ as $\dim(\mathscr{P}) = p - \text{rank}(A_S)$ and say S is the set of maximal affine constraints.

Definition 3.2 The **Hull $(Q, \mathscr{P})$** is defined to be

$$\text{Hull}(Q, \mathscr{P}) = \{x \in \mathbb{R}^n : x = Qw, w \in \mathscr{P}\}. \tag{3}$$

where $Q \in \mathbb{R}^{n \times p}$ and $\mathscr{P} \subseteq \mathbb{R}^p$. We say that $x \in \text{Hull}(Q, \mathscr{P})$ is a $\mathscr{P}$-combination of Q with $w \in \mathscr{P}$ if $x = Qw$.

The paper shows that there always exists a solution to the polyhedral regression

$$\min_{\beta \in \mathscr{P}} \frac{1}{2} \|y - X\beta\|_2^2, \tag{4}$$

that binds the constraints of A. Also, the paper shows show the such a solution is unique when it isrestricted to the affine subspace defined by the binding constraints, which is the local uniqueness property.

High-level description of the theoratical results

The paper first establishes Lemma 3.3 quantifies the number of binding constraints for polyhedrons with non-empty interior point, then it is extended to a more general case by removing the assumption of having non-empty interior in Therorem 3.4. From Theorem 3.4, we can immediately obtain the result od Collary 3.5, which is the polyhedral regression. Finally, the paper ends this session by stating the Theorem 3.6, which is the local uniqueness property. For the Lemma, Collary and Theorems below, we will use the following notation: let $Q \in \mathbb{R}^{n \times p}$ be any matrix, $\mathscr{P} \subseteq \mathbb{R}^p$ be any polyhedron represented by $A \in \mathbb{R}^{m \times p}$ and $b \in \mathbb{R}^m$. Also, let S be the set of maximal affine constraints and $x \in \text{Hull}(Q, \mathscr{P})$.

Lemma 3.3 If $\mathscr{P}$ has a non-empty interior and $\text{Null}(Q)\backslash\{\mathbf{0}\} \subseteq \text{Null}(A)^c$, then x can be written as a $\mathscr{P}$-combination of Q with an element $w \in \mathscr{P}$ that binds at least $\min(\text{rank}(A), (p-n)_+)$ constraints of A.

The lemma is particularly useful when $p > n$ since it views the facet of the polyhedron as a lower dimensional space. Intuitively, after restricting ourselves from $\mathscr{P}$ to Q, we remove $p - n$ degrees of freedom, as long as there are enough i.e. $\text{rank}(A) > p - n$ to be removed. Mathematically, $\text{Null}(Q) \setminus \{\mathbf{0}\} \subseteq \text{Null}(A)^c$ ensures that the element in $\text{Null}(Q)$ must not be ortgonal to the constraints of A, while the non-empty interior condition ensures the line intersects $\mathscr{P}$ at more than one point.

Theorem 3.4 (Existence of sparse representation) If $V \in \mathbb{R}^{p \times \dim(\mathscr{P})}$ is afull column rank matrix such that $\text{Null}(QV) \setminus \{\mathbf{0}\} \subseteq \text{Null}(A_S V)^c$, then $x \in \text{Hull}(Q, \mathscr{P})$ can be written as a $\mathscr{P}$-combination of Q with an element $w \in \mathscr{P}$ that binds at least $\min(\text{rank}(A) - \text{rank}(A_S), \dim(\mathscr{P}) - \text{rank}(QV))$ constraints of A_{-S}.

The theorem first takes away the assumption of non-empty interior stated in lemma 3.3, which is done by expressing the polyhedron in terms of the non-empty interior $\{A_{-S}x \leq b_{-S}\}$. After having the constrainted matrix $A_{-S}V$, with $\text{rank}(A_{-S}V) = \text{rank}(A) - \text{rank}(A_S)$, the theorem also shows QV is the full rank matrix of $A_{-S}V$, such that we can obtain $\dim(\mathscr{P}) - \text{rank}(QV)$ and follow the result in lemma 3.3.

Collary 3.5 (Polyhedral regression) Let $X \in \mathbb{R}^{n \times p}$ be a feature matrix and $y \in \mathbb{R}^n$ be the response vector. Take $\mathscr{P}$ such that the conditions of Theorem 3.4 hold with $Q \equiv X$. Also, let $V \in \mathbb{R}^{p \times \dim(\mathscr{P})}$ such that $A_S V = \mathbf{0}$. Then, there exists a solution $\hat{\beta} \in \mathbb{R}^p$ to (4) that binds at least $\min(\text{rank}(A) - \text{rank}(A_S), \dim(\mathscr{P}) - \text{rank}(XV))$ constraints of A_{-S}.

This collary can be immediately obtained from Theorem 3.4, since we have $X\beta = \Pi_{\text{Hull}(X,\mathscr{P})}(y) \in \text{Hull}(X, \mathscr{P})$, where the projection of the hull is closed and convex.

Theorem 3.6 (Local uniqueness property) Suppose the conditions of Theorem 3.4 hold and

$$\text{rank}(A) - \text{rank}(A_S) \geq \dim(\mathscr{P}) - \text{rank}(QV)$$

Let w be the sparse representation that binds at least $\dim(\mathscr{P}) - \text{rank}(QV)$ constraints of A_{-S}. Denote $(A_{-S})_T$ be the binding constraints, where T is the set of indices. Let $\mathscr{T} = \{w \in \mathbb{R}^p : (A_{-S})_T w = (b_{-S})_T\}$. Then, w is the unique element in $\mathscr{P} \cap \mathscr{T}$ such that $x = Qw$.

This theorem shows that the sparse solution w is unique when restricted to the subspace by Theorem 3.5. The ineqaulity in the condition usually holds in practice since we have $\text{rank}(A) \geq p$. This theorem shows that the solution is unique when the binding constraints are identified correctly, which is particularly useful in optimization algorithms as we can perform the active-set strategies rapidly and obtain the optimal solution upon convergence.

Collary 3.7 (NNLS solution) There exists a solution with at most $\text{rank}(X)$ positive value to the NNLS problem.

Take $A = -I$ and $b = \mathbf{0}$. Since the conditions of collary 3.5 hold, we can say there exists $\beta \in \mathbb{R}$ that binds at least $p - \text{rank}(X)$ constraints of A, which is the result of Collary 3.7.

Collary 3.8 (BVLS solution) There exists a solution with at most rank(X) non-boundary value to the BVLS problem.

Take $A = \begin{bmatrix} -I \\ I \end{bmatrix}$ and $b = \begin{bmatrix} -l \\ u \end{bmatrix}$. Since the conditions of collary 3.5 hold, we can say there exists $\beta \in \mathbb{R}^{|}$ that binds at least $p - \text{rank}(X)$ constraints of A, which is the result of Collary 3.8.

Coordinate Descent for BVLS

Let $X \in R^{n \times p}$ and $y \in R^n$ be the feature matrix and response vector respectively, and denote l and u as the lower and upper bounds. The NNLS problem is just setting $l = 0$ and $u = \infty$. The algorithm is intended to solve the BVLS problem in a high dimentional case i.e. $p >> n$, we have Collary 3.8 that guarantees the solution is unique and sparse. The algorithm is based on the coordinate descent, in which convergence is guaranteed [Tse01]. This is because coordinate descent is the state-of-the-art method in models like the lasso and group lasso [YH24]. Therefore, it is expected that the algorithm can efficient and effective solve the sparse solution to the BVLS problem.

High-level description of the BVLS algorithm

We first choose a subset of the features S, which requires adjustments the most. We will check whether the current β is optimal by evaluating the KKT conditions. If the KKT conditions are not satisfied, we will solve the BVLS problem using S. This process is repeated iteratively until the KKT conditions are met or S includes all features i.e. $S = [p]$. We will do this by computing the violation(δ_i) of the KKT conditions as follows:

$$\delta_i = \begin{cases} \max\{0, -\nabla_i f(\beta)\} & \text{if } \beta_i < u_i \\ \max\{0, \nabla_i f(\beta)\} & \text{if } \beta > l_i \end{cases} \qquad \forall i \in [p] \tag{5}$$

We then sort the δ_i in descending order and choose the large violations, i.e. $\delta_i > 0$ in S to update. Collary 3.8 shows that rank(X) is always the upper bound that guarantees the existence of a unique solution. However, if we do not know what rank(X) is, we can still estimate it by min(n, p). After checking the KKT conditions on S^c, if we can obatin the optimal solution, which has violation equal to 0, then we are done. Otherwise, we will solve the BVLS problem on S and update β. We first perform one coordinate descent on S to identify the active set. If we are converged, then we are done. Otherwise, we will do coordinate descent on the active set repeatedly until convergence. Also, in every iteration, we will remove the variables that have reached the boundary. We will set the initial value of β to be the vertex of the box that is closest to 0. This is because we want to simulate the shrinkage effect in the penalized regression methods.

2 Result and Discussion

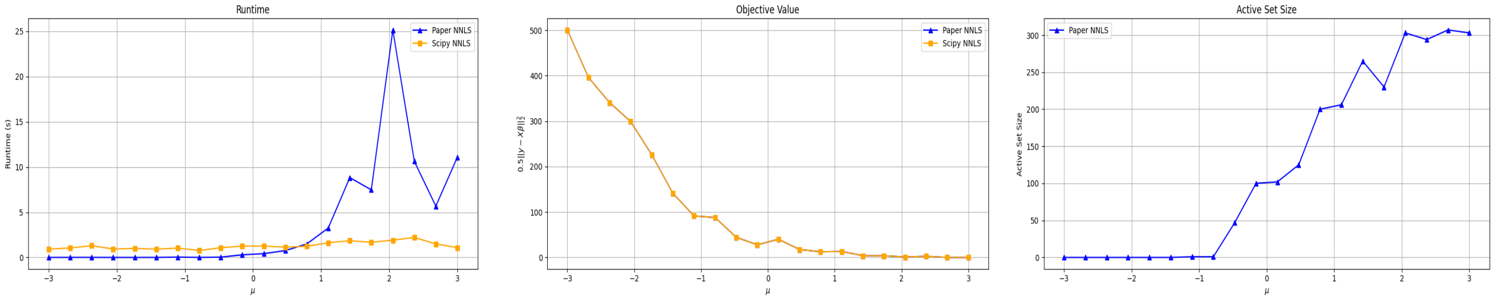
My implementation is based on the pseudocode of the BVLS algorithm in the paper and SciPy's implementation of the BVLS algorithm. In my implementation, I reproduce the result of paper's algorithm and try compare the performance of the two algorithms by running them on the same dataset.

2.1 Dataset

I first generate non-negative i.i.d matrix X uniformly on (0, 1) and set 80% of the entries to be 0. This ensures us getting pairwise othogonal vector in a large probability. Then, i generate vector $y \sim \mathcal{N}(\mu, \sigma^2)$, where 20 of the μ are equally chosen from $[-3\sigma, 3\sigma]$ with $\sigma = 1$. Then I set $p = 1000$ and $p = 10000$ to see how large the dimension need to be in order to be consider high-dimensional. For both cases, I set $n = 100$. For the NNLS problem, which is setting $l = 0$ and $u = \infty$. As for the BVLS problem, I set $l = 0$ and $u = 1$.

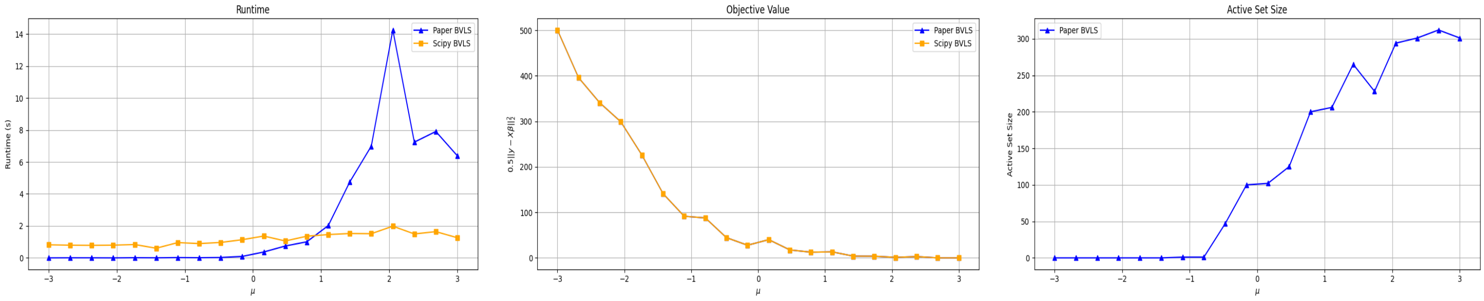
2.2 Result

The result is shown in figures below, we can see the comparison between the runtime (a), the objective value (b) and the size of the active set (c). For (a), we can see that in the case of $p = 1000$, the SciPy NNLS and BVLS always performs better than the paper's algorithm. This could be simply understood as the dimension is not high enough to see the advantage of the paper's algorithm. However, for $p = 10000$, we can see that the paper's algorithm is much faster than the SciPy's algorithm especially when μ is small. When μ is large, the paper's algorithm is still faster than the SciPy's algorithm, but the difference is not as significant as when μ is small. This is because when μ is large, the method of the paper will include more variables than usual does. For (b), we can see that the objective value of both algorithms are the same. For (c), we can see that the size of the active set for NNLS is fewer than that of BVLS, which is expected since the NNLS problem has a larger bound than the BVLS problem. Also, we may see that the size of the active set gradually increases as μ increases, which is expected since there are more $\min(n,\, p)$ active variables.



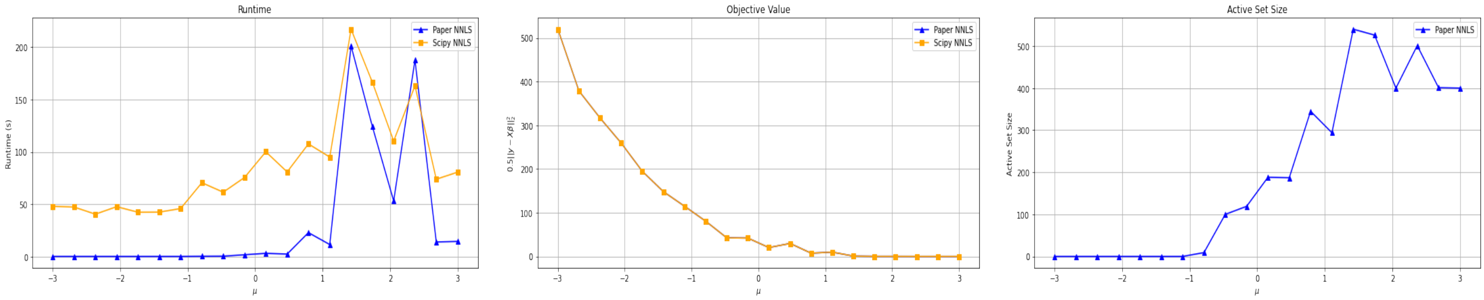
(a) (b) (c)

Figure 1: NNLS for $n = 100$ and $p = 1000$



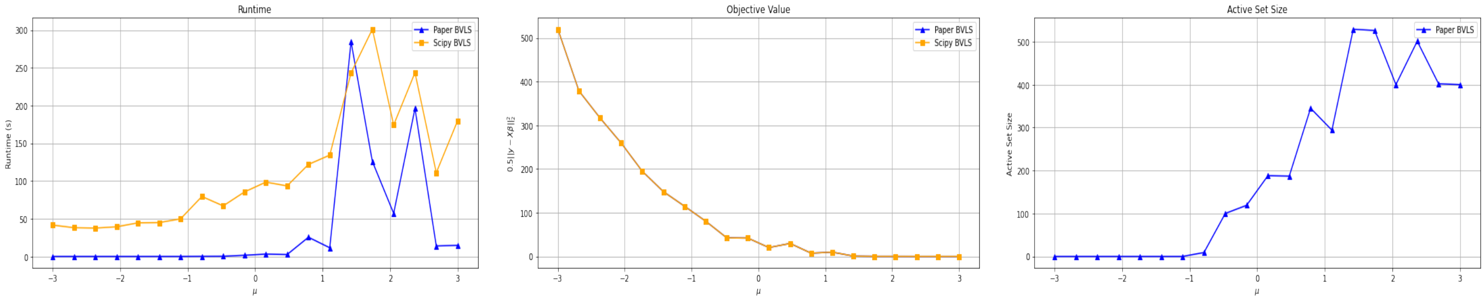
(a) (b) (c)

Figure 2: BVLS for $n = 100$ and $p = 1000$



(a) (b) (c)

Figure 3: NNLS for $n = 100$ and $p = 10000$



(a) (b) (c)

Figure 4: BVLS for $n = 100$ and $p = 10000$

3 Conclusion

In this paper, the authors introduce a unified framework for solving polyhedral constrained least squares problems. They show that the problem admits locally unique sparse solutions in high dimensions. The paper provides a theoretical understanding of the problem and presents a novel algorithm based on coordinate descent. The results are promising and suggest that the proposed algorithm can efficiently solve high-dimensional constrained regression problems.

References

[1] [Tse01] P. Tseng: *Convergence of a block coordinate descent method for nondifferentiable minimization.* Journal of Optimization Theory and Applications, 2001.

[2] [YH24] J. Yang, T. Hastie: *A fast and scalable pathwise-solver for group lasso and elastic net penalized regression via block-coordinate descent*, 2024.