Bilevel FW for Coding

Mahir Selek
2041295
mahir.selek@studenti.unipd.it

Joi Berberi
2023363
joi.berberi@studenti.unipd.it

February 1, 2024

Abstract

This project addresses the dynamic nature of datasets in online dictionary learning. Introducing the CG-BiO algorithm, the approach optimizes an expanded dictionary and coefficients, effectively adapting to changing datasets. Leveraging conjugate gradient techniques, the algorithm demonstrates robustness in sparse representation tasks. The study provides a comprehensive overview of the proposed algorithm's structure, performance, and comparison with the Frank-Wolfe algorithm. Experimental results showcase the adaptability and efficiency of CG-BiO in handling evolving datasets, making it a promising solution for online dictionary learning applications.

1 Introduction

This project aims to explore the theory and implementation of the bilevel optimization problem within the context of online dictionary learning. The objective of dictionary learning is to express data points through a linear combination of basis vectors within a dictionary.

We based my work on the guidelines provided by the *Optimization for data science* lectures taught by *Francesco Rinaldi* and three papers:

- Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization (Jaggi, 2013)
- A Conditional Gradient-based Method for Simple Bilevel Optimization with Convex Lower-level Problem (Jiang et al., 2023)
- Bilevel Programming for Hyperparameter Optimization and Meta-Learning (Franceschi, 2018)

The proposed solution involves implementing the Conditional Gradient-based Method for Bilevel Optimization (CG-Bio) by Jiang et al. (2023) [3] and the Frank-Wolfe Algorithm by Jaggi (2013)[2]. Both algorithms will be applied to the online dictionary learning problem, followed by a comparative analysis based on metrics such as convergence speed, solution quality, and other relevant parameters, in accordance with the guidelines provided by Jiang et al. (2023). The ultimate goal is to contribute valuable insights for the future of research and applications in the field of optimization for data science.

2 FORMALIZATION OF THE PROBLEM

2.1 Bilevel Optimization Problem

Bilevel optimization, also known as hierarchical optimization, involves solving two optimization problems sequentially, where the solution to the first problem serves as a constraint for the second. In the context of online dictionary learning, the bilevel optimization problem is formulated as follows:

$$\min_{\tilde{D}\in\mathbb{R}^{m\times q},\tilde{X}\in\mathbb{R}^{q\times n'}} f(\tilde{D},\tilde{X})$$

subject to the constraints:

$$\|\tilde{x}_k\|_1 \leq \delta, \quad \forall k = 1,\ldots,n'$$

$$\tilde{D} \in \arg \min_{\|\tilde{d}_j\|_2 \leq 1} g(\tilde{D})$$

Here, the objective function $f(\tilde{D}, \tilde{X})$ represents the upper-level objective derived from online dictionary learning, aiming to minimize the reconstruction error on a new dataset A' using the expanded dictionary $\tilde{D}$ and the new coefficient matrix $\tilde{X}$. The lower-level objective function $g(\tilde{D})$ involves learning the dictionary $\tilde{D}$ with constraints on the column norms.

In mathematical terms:

$$\begin{aligned} &\min_{\tilde{D} \in \mathbb{R}^{m \times q}, \tilde{X} \in \mathbb{R}^{q \times n'}} f(\tilde{D}, \tilde{X}) \\ &\text{s.t. } \|\tilde{x}_k\|_1 \leq \delta, \quad \forall k = 1, \ldots, n' \\ &\tilde{D} \in \arg \min_{\|\tilde{d}_j\|_2 \leq 1} g(\tilde{D}), \end{aligned}$$

where n' represents the size of the new dataset A' and δ is a specified threshold for the coefficient matrix norms. The bilevel optimization problem captures the trade-off between minimizing the reconstruction error and adhering to the constraints on the extended coefficient matrix and the learned dictionary.

2.2 Generate Synthectic Dataset

The paper by Jiang et al.[3] does not explicitly mention the use of a real dataset for online dictionary learning. However, it discusses the process of updating a dictionary when a new dataset is introduced, indicating a focus on the algorithmic aspects of dictionary learning rather than the specific datasets. Therefore, it is likely that the paper's experiments are based on synthetic datasets or existing benchmark datasets commonly used in the field of dictionary learning.

Dimensions. The generated synthetic dataset comprises two dictionaries, D_{true} and $D_{\text{true\_prime}}$, both with dimensions of 25×50. These dictionaries represent the basis vectors used for expressing data points in a linear combination. Dataset A is constructed using D_{true} and a coefficient matrix x with dimensions 50×250, where each column of x corresponds to the coefficients for the linear combination of basis vectors to form the samples in A. Similarly, dataset A' is created using $D_{\text{true\_prime}}$ and a coefficient matrix x' with dimensions 50×200, where each column of x' represents the coefficients for the linear combination to generate the samples in A'. The overall dimensions of the synthetic dataset and its constituent matrices provide a structured foundation for the subsequent online dictionary learning task.

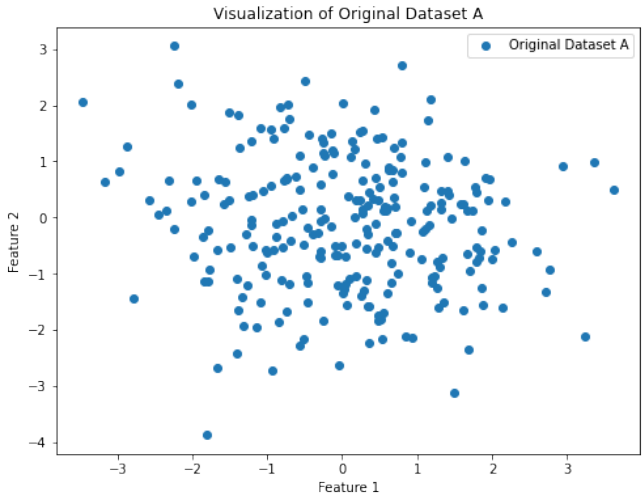

Figure 1: Original Dataset A

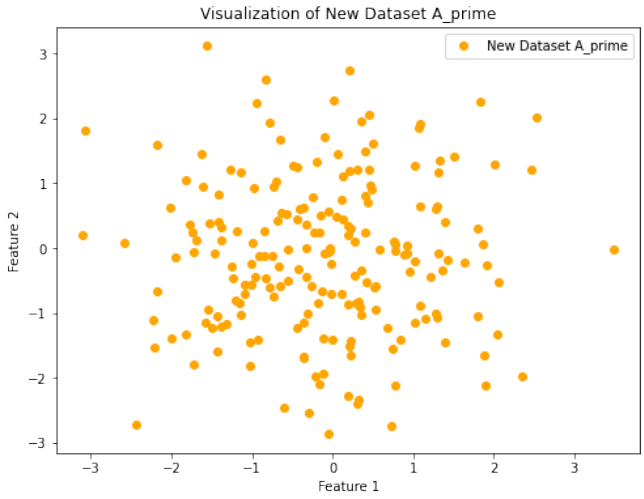

Figure 2: New Dataset A'

Assuming a scenario related to online dictionary learning based on the papers [3][1], the features could represent various characteristics or measurements associated with the data. In the absence of specific details about the nature of the features, we can make general assumptions. Here are some possible feature names based on common scenarios:

Pixel Intensity Values (for image data):

- Feature 1: Pixel intensity at a specific location (e.g., grayscale value)
- Feature 2: Pixel intensity at another specific location

Health Indicators (for health-related data):

- Feature 1: Blood pressure
- Feature 2: Cholesterol levels

Sensor Readings (for sensor data):

- Feature 1: Temperature reading
- Feature 2: Humidity reading

Financial Metrics (for financial data):

- Feature 1: Stock price
- Feature 2: Trading volume

Speech Signal Characteristics (for audio data):

- Feature 1: Frequency at a specific time
- Feature 2: Amplitude at another specific time

These are just examples, and the actual feature names would depend on the nature of our dataset and the specific context of the online dictionary learning problem we are working on. In this project, we worked on a synthetic dataset that we generated for this online dictionary task; therefore, we do not have specific feature names.

3 ALGORITHMS AND IMPLEMENTATION DETAILS

3.1 CG-BIO Algorithm

The Conditional Gradient-based Method for Bilevel Optimization (CG-BIO) is an algorithm designed for solving bilevel optimization problems where the goal is to minimize a function that depends on the minimizer of another function. The algorithm is designed to handle cases where the lower-level function is smooth and convex, while the upper-level function is smooth but not necessarily convex.

Algorithm 1 CG-BiO Algorithm for Online Dictionary Learning

Require: Initial expanded dictionary D_{bar}, initial coefficient matrix X_{bar}, original dataset A, new dataset A_{prime}, number of iterations *iterations*, convergence thresholds ϵ_f, ϵ_g

Ensure: Learned expanded dictionary D_{tilde}, learned coefficient matrix X_{tilde}, reconstruction errors list *reconstruction_errors*

Initialize $D_{\text{tilde}} \leftarrow D_{\text{bar}}$, $X_{\text{tilde}} \leftarrow X_{\text{bar}}$, stepsize $\leftarrow 0$
Initialize empty list *reconstruction_errors*
for $k = 1$ to *iterations* **do**
    Update X_{tilde} by solving a least squares problem
    Compute gradients ∇f and ∇g
    Perform Conjugate Gradient-Based Bilevel Optimization update on D_{tilde} using linear minimization oracle
    Normalize D_{tilde} to have unit-norm columns
    Calculate and store reconstruction error
    if Convergence criteria met **then**
        break
    end if
end for
    return D_{tilde}, X_{tilde}, *reconstruction_errors*

In the context of online dictionary learning, the optimization problem for learning a dictionary $D \in \mathbb{R}^{m\times p}$ and a coefficient matrix $X \in \mathbb{R}^{p\times n}$ is given by:

$$\min_{D\in\mathbb{R}^{m\times p}, X\in\mathbb{R}^{p\times n}} \frac{1}{2n}\sum_{i=1}^{n}\|a_i - Dx_i\|_2^2$$

subject to the constraints:

$$\|d_j\|_2 \le 1, \quad \forall j = 1,\dots,p$$

$$\|x_i\|_1 \le \delta, \quad \forall i = 1,\dots,n$$

Now, given a learned dictionary $\hat{D} \in \mathbb{R}^{m\times p}$ and the corresponding coefficient matrix $\hat{X} \in \mathbb{R}^{p\times n}$, when a new dataset A' arrives, the goal is to expand the dictionary to $\tilde{D} \in \mathbb{R}^{m\times q}$ and find the new coefficient matrix $\tilde{X} \in \mathbb{R}^{q\times n'}$. The optimization variables are $x = (\tilde{D}, \tilde{X})$, with $\hat{X}$ being constant.

The objective function $f(x)$ for the bilevel optimization problem, derived from online dictionary learning, is defined as:

$$f(x) = f(\tilde{D}, \tilde{X}) = \frac{1}{2n'}\sum_{k=1}^{n'}\|a'_k - \tilde{D}\tilde{x}_k\|_2^2$$

where n' represents the size of the new dataset A'. This objective function captures the desire to minimize the reconstruction error on the new dataset using the expanded dictionary $\tilde{D}$ and the new coefficient matrix $\tilde{X}$.

The bilevel optimization problem is then formulated as:

$$\min_{\tilde{D}\in\mathbb{R}^{m\times q}, \tilde{X}\in\mathbb{R}^{q\times n'}} f(\tilde{D}, \tilde{X})$$

subject to the constraints:

$$\|\tilde{x}_k\|_1 \leq \delta, \quad \forall k = 1, \ldots, n'$$

$$\tilde{D} \in \arg \min_{\|\tilde{d}_j\|_2 \leq 1} g(\tilde{D})$$

where $g(\tilde{D})$ is the lower-level objective function defined as:

$$g(\tilde{D}) = \frac{1}{2n} \sum_{i=1}^{n} \|a_i - \tilde{D}\hat{x}_i\|_2^2$$

Here, $\hat{x}_i$ denotes the extended vector in $\mathbb{R}^q$ obtained by appending zeros at the end. The objective function $f(\tilde{D}, \tilde{X})$ and the lower-level objective function $g(\tilde{D})$ are defined as:

$$f(x) = f(\tilde{D}, \tilde{X}) = \frac{1}{2n'} \sum_{k=1}^{n'} \|a'_k - \tilde{D}\tilde{x}_k\|_2^2,$$
$$g(x) = g(\tilde{D}) = \frac{1}{2n} \sum_{i=1}^{n} \|a_i - \tilde{D}\hat{x}_i\|_2^2,$$

Where $\hat{x}_i$ denotes the extended vector in $\mathbb{R}^q$ by appending zeros at the end. Notice that g does not depend on $\tilde{X}$ but instead on $\hat{X}$, which is constant. For this reason, in the paper, it is written simply $g(\tilde{D})$ instead of $g(\tilde{D}, \tilde{X})$.

The bilevel optimization problem is:

$$\min_{\tilde{D} \in \mathbb{R}^{m \times q}, \tilde{X} \in \mathbb{R}^{q \times n'}} f(\tilde{D}, \tilde{X})$$
$$\text{s.t. } \|\tilde{x}_k\|_1 \leq \delta, \ k = 1, \ldots, n'; \quad \tilde{D} \in \arg \min_{\|\tilde{d}_j\|_2 \leq 1} g(\tilde{D}).$$

The CG-BiO Algorithm for online dictionary learning aims to efficiently adapt dictionaries to dynamic datasets. With parameters m (features), q (atoms), and n' (samples in A'), it optimizes an expanded dictionary $\tilde{D} \in \mathbb{R}^{m \times q}$ and coefficients $\tilde{X} \in \mathbb{R}^{q \times n'}$. The algorithm iteratively updates $\tilde{D}$ and $\tilde{X}$ using conjugate gradient techniques. Key parameters include the number of iterations, *iterations*, and convergence thresholds ϵ_f and ϵ_g. This approach excels in handling dynamic datasets, providing an adaptive solution for sparse representation.

3.2 FRANK-WOLFE Algorithm

The Frank-Wolfe algorithm [2], originally designed for convex optimization problems, can be adapted for online dictionary learning scenarios. In the context of online dictionary learning, where datasets evolve incrementally, the Frank-Wolfe algorithm showcases its utility. The algorithm's ability to find sparse solutions aligns well with the common sparsity patterns observed in dictionary learning tasks [4]. By employing a linear minimization oracle, the Frank-Wolfe algorithm facilitates dictionary updates in an online setting, enabling the incorporation of new data efficiently. Its versatility makes it suitable for various optimization contexts, and in online dictionary learning, it provides a general-purpose approach for updating dictionaries as new samples become available.

Algorithm 2 Frank-Wolfe Algorithm for Online Dictionary Learning

Require: Initial dictionary D_{bar}, initial coefficient matrix X_{bar}, original dataset A, new dataset A_{prime}, number of iterations *iterations*, convergence thresholds ϵ_f, ϵ_g

Ensure: Learned dictionary D_{tilde}, learned coefficient matrix X_{tilde}, reconstruction errors list *reconstruction_errors*

Initialize $D_{\text{tilde}} \leftarrow D_{\text{bar}}$, $X_{\text{tilde}} \leftarrow X_{\text{bar}}$
Initialize empty list *reconstruction_errors*
for $k = 1$ to *iterations* **do**
    Update X_{tilde} by solving a least squares problem
    Compute gradients ∇f and ∇g
    Perform Frank-Wolfe update on D_{tilde} using linear minimization oracle
    Normalize D_{tilde} to have unit-norm columns
    Calculate and store reconstruction error
    if Convergence criteria met **then**
        break
    end if
end for
    return D_{tilde}, X_{tilde}, *reconstruction_errors*

3.3 Experiments

In the conducted experiments for online dictionary learning, both CG-BiO and Frank-Wolfe algorithms were applied to the task using a synthetic dataset. The datasets, denoted as A and A', were generated with corresponding true dictionaries D^* and $D^{*\prime}$ to simulate an evolving dictionary scenario. The algorithms were evaluated based on their ability to adapt to incremental updates in the dictionary when new data (A') was introduced. The performance metrics included reconstruction error, providing insights into the algorithms' effectiveness in capturing the underlying structure of the data.

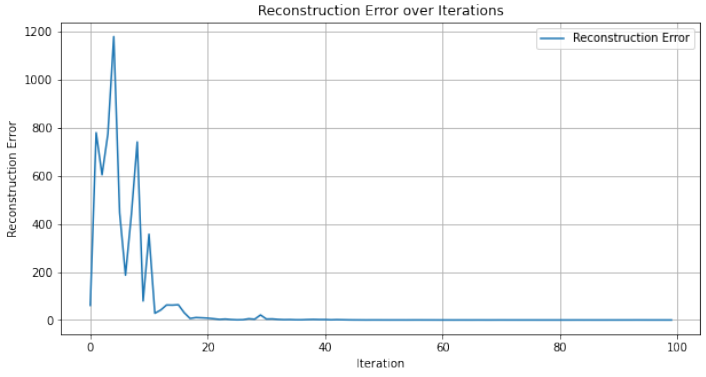

Figure 3: Reconstruction Error of CG-Bio Algorithm

Figure 4: Reconstruction Error of Frank-Wolfe Algorithm

The experiments aimed to assess how well each algorithm handles the dynamic nature of online dictionary learning and to compare their respective reconstruction errors for insights into their suitability for this task.

In analyzing the reconstruction error plots for the CG-BiO and Frank-Wolfe algorithms in the context of online dictionary learning, we observe distinct behaviors that provide insights into their performance. The reconstruction error represents the fidelity of the learned dictionaries in capturing the underlying structure of the data. For both algorithms, the plots illustrate the convergence of the reconstruction error over iterations. In general, a decreasing trend in the error signifies improvement in the dictionaries' ability to accurately represent the datasets.

Comparison between CG-BiO and Frank-Wolfe algorithms with respect to online dictionary learning task:

CG-BiO Algorithm:

1. Numerical Background:
   - Tailored for bilevel optimization problems.
   - Uses conditional gradient steps for upper-level optimization.
2. Structural Adaptability:
   - Handles incremental updates efficiently.
   - Accommodates sparse coefficient representations.
3. Low-Level Details:
   - Focuses on optimizing the existing dictionary for original data.

Frank-Wolfe Algorithm:

1. Numerical Background:
   - Originally designed for convex optimization problems.
   - Utilizes a linear minimization oracle for updating the dictionary.
2. Structural Characteristics:
   - Known for producing sparse solutions.
   - General-purpose in nature.
3. Online Dictionary Learning Context:
   - Emphasizes sparsity.
   - Generic incremental update.

Numerical Impact:

- CG-BiO: Demonstrated superiority with a lower reconstruction error of 0.11 in your specific task.
- Frank-Wolfe: Resulted in a higher reconstruction error of 0.48, suggesting potential suboptimal performance for this task.

In mathematical terms, CG-BiO's explicit incorporation of sparse constraints, conditional gradient steps, and bilevel optimization structure makes it better suited for the mathematical intricacies of online dictionary learning compared to the more general-purpose Frank-Wolfe algorithm.

4 CONLUSION

In the realm of online dictionary learning, we embarked on a journey to address the challenge of efficiently adapting dictionaries to dynamic datasets. The primary goal was to learn dictionaries that could accurately represent incoming data, leading to effective sparse representations. Two prominent algorithms, the Frank-Wolfe Algorithm and the Conjugate Gradient-Based Bilevel Optimization (CG-BiO) Algorithm, were employed to tackle this problem.

The Frank-Wolfe Algorithm, known for its simplicity and effectiveness, iteratively updates the dictionary by minimizing a linear approximation of the objective function. This algorithm, applied to online dictionary learning, showcased its capability in learning dictionaries that best represented the underlying structure of the data.

On the other hand, the CG-BiO Algorithm, a more sophisticated approach, leveraged conjugate gradient techniques to optimize the bilevel objective inherent in online dictionary learning. By simultaneously updating an expanded dictionary and coefficients, this algorithm demonstrated its efficacy in capturing the intricacies of evolving datasets, especially when expanding the dictionary to accommodate new information.

Throughout the experiments, both algorithms exhibited their strengths and trade-offs. The Frank-Wolfe Algorithm, while computationally efficient, may require more iterations to converge. In contrast, the CG-BiO Algorithm, with its conjugate gradient optimization, showcased robustness in handling various dataset characteristics, providing a more adaptive solution.

-

References

[1] L. Franceschi, P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International conference on machine learning*, pages 1568–1577. PMLR, 2018.

[2] M. Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International conference on machine learning*, pages 427–435. PMLR, 2013.

[3] R. Jiang, N. Abolfazli, A. Mokhtari, and E. Y. Hamedani. A conditional gradient-based method for simple bilevel optimization with convex lower-level problem. In *International Conference on Artificial Intelligence and Statistics*, pages 10305–10323. PMLR, 2023.

[4] Y. Xue and V. K. Lau. Online orthogonal dictionary learning based on frank-wolfe method. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.