S&DS 631 Project Proposal

A Deep Dive into the Frank-Wolfe Method

Gonghan Xu

1 Problem Description and Background

The Frank-Wolfe method[1], also known as the conditional gradient method, is an iterative algorithm designed for convex optimization problems with compact, convex feasible sets. Originated in the 1950s by Marguerite Frank and Philip Wolfe, this method provides a unique approach by avoiding direct projections onto the feasible set, which can be computationally expensive or intractable for certain problems. Instead, it relies on linear approximations to the objective function, making it particularly useful for optimization over polyhedra or simple constraint sets.

In recent years, with the upsurge of large-scale optimization problems in various fields like machine learning, data science, and operations research, there has been a revived interest in the Frank-Wolfe method. This is due to its simplicity, adaptability, and efficiency, especially in scenarios where other optimization methods struggle.

2 Objectives

Theoretical Understanding: Delve deep into the convergence properties and complexity of the Frank-Wolfe method. This will involve understanding scenarios where the method excels, its limitations, and the factors that influence its convergence rates. Additionally, this theoretical investigation will shed light on why the method is well-suited for certain problems, especially those with challenging constraint sets.

Practical Understanding: Implement the Frank-Wolfe method using Python and test its performance on various optimization problems. This hands-on approach will offer insights into the nuances of the method and how practical challenges, such as the choice of step size, can affect its efficacy. Furthermore, a comparative analysis will be performed against gradient descent to contextualize the strengths and weaknesses of the Frank-Wolfe method.

3 Plan of Action

Literature Review: Begin with textbooks and research papers to acquire a comprehensive understanding of the method's history, evolution, and current relevance.

Theoretical Analysis: Study the convergence proofs, analyzing under which conditions the method converges and how different factors influence its rate of convergence.

Implementation: Develop a Python-based implementation of the Frank-Wolfe method, possibly leveraging tools like cvxpy for the linear minimization oracle.

Experiments: Design experiments on benchmark optimization problems to evaluate the method's performance. Compare its results with gradient descent, noting relative advantages and drawbacks of each method.

Report and Presentation: Consolidate findings into a comprehensive report and prepare a presentation to share results and insights on the topic.

By the conclusion of this project, a comprehensive understanding of the Frank-Wolfe method, from both theoretical and practical perspectives, will be achieved. This exploration will provide valuable insights into its role and relevance in optimization challenges.

References

[1] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110, 1956.