EE 516: Mathematical Foundations of Machine Learning **Winter 2023**

Course Project: Project Topic
Due: March 24, 2023, 11:59PM PT

Student Names: Student Names

Use this template for your project. Delete my description text below when you write your report, and **include your names above!**

1 Paper Descriptions

All individual paper descriptions go in this section. Details on what should be in each section are in Section 1.1. Note that I am including the `bibliography.bib` file so you can learn to cite papers like this [1]. The easy way to do this is to go to Google Scholar, search the paper name, and click the quotation mark icon. There is a link to the BibTeX entry that you can simply copy and paste into your `bibliography.bib` file.

1.1 Paper 1

Paper Title: Title goes here
Student Name: Student name goes here

1.1.1 Problem Description & Formulation

Each team member should write their own problem description. You may (and should) discuss the problem among yourselves, but I want each of you to write your own individual description of the problem.

- Be sure to give both a verbal/intuitive problem description as well as a mathematical description and formulation of what the proposed algorithm solves.
- Talk about why this problem is important, either from a practical or theoretical perspective (or both).
- Discuss what gaps exist in the literature and how this paper fills those gaps. What is the novel contribution of this paper, and why is it important?
- Do **not** simply copy the problem description from your selected paper. This is plagiarism. Your description should be your own.
- In this and all sections, be sure to define every variable used, as well as any non-standard sets, objects, or functions/operations.

1.1.2 Algorithm Description

In this section, you should give an intuitive explanation of the algorithm. Below are some prompts that may be helpful.

- What type of algorithm is this? Does it use iterative methods? Optimization? Geometric intuition?
- Why should this algorithm work? Provide intuition beyond what the theoretical results can tell us.
- What tools or methods does the algorithm leverage? Is it applying a known formulation to this specific problem? Is it leveraging some geometric insight to simplify the problem?

- Does the algorithm perform well empirically? In what cases?
- What (if any) topics from class are used in the algorithm, either in its development or in the actual pseudo-code?

1.1.3 Theoretical Results

In this section, describe the theoretical results that accompany the algorithm (if any). Note that I do not expect you to understand the proofs (or even the more technical results) in detail. The main goal is to get a clear picture of what can be said about the algorithm.

- What assumptions are made for the theoretical results to hold?
- Do the results guarantee that the problem of interest is solved? Under what conditions?
- Do the results guarantee that the algorithm converges? To a local or global minimum?

1.1.4 Relation to Course Material

In this section, you should discuss *any* aspects of the paper that relate to the course material. Consider the introduction, problem statement, algorithm, theoretical results, empirical results, and any conclusions.

- What (if any) topics from class are used in the algorithm, either in its development or in the actual pseudo-code?
- What (if any) topics from class are used in the theoretical results?
- Are there any portions of the paper that you could not have understood prior to taking this course that now make (at least some) sense to you?
- What aspects of the paper do you still not understand? Are these beyond the scope of ECE 516, or are they parts of the class that you still don't understand? Be honest please :)

1.2 Paper 2

Paper Title: Title goes here
Student Name: Student name goes here
Same subsections as Section 1.1.

1.3 Paper 3

Paper Title: Title goes here
Student Name: Student name goes here
Same subsections as Section 1.1.

2 Comparison of Algorithms

In this section, your group should compare the following merits of the papers.

- Interpretability. Is any single algorithm simpler to understand or implement? This may seem silly, but it is often a determining factor in which algorithms are actually used in practice.
- Theoretical guarantees. Does any paper have the strongest theoretical guarantees? Does one algorithm work in certain cases, while another works in other cases?

- Empirical guarantees. Does any paper have the strongest empirical results? Does one algorithm work in certain cases, while another works in other cases? Consider both synthetic and real/benchmark datasets if available.
- Computational complexity. Discuss the computational complexity of each algorithm. What problem parameters does it scale with? Are there regimes in which one algorithm should be chosen over another?

3 Algorithm Implementation & Testing

Instructions below. Do not feel obligated to run exactly the same tests as in the paper. These often take a large amount of computation time.

- Clearly state which algorithm you've implemented and why you chose this algorithm.
- Include your algorithm code either as an appendix at the end of the paper or in some digital perform if preferred (e.g., a Jupyter notebook or GitHub repository).

3.1 Results on Synthetic Data

In this section, include a few plots of performance on synthetic data. You may choose to simulate scenarios similar to those in the corresponding paper. Groups of three should do at least one plot. Groups of four should do at least two and try to find the limits of where the algorithm performs well. Some plots of interest may be

- Performance vs. noise level
- Performance vs. problem size (heatmap if applicable)
- Performance vs. tuning parameters (heatmap if applicable)
- Run time vs. problem size

3.2 Results on Benchmark Data

Choose at least one benchmark dataset where it makes sense to apply your algorithm. You may choose one from among the papers your group has read or find a different dataset. Be sure to cite the appropriate authors or location for your dataset.

- Report the performance of your implemented algorithm.
- Report the run time for your algorithm on this dataset.
- You may either report the best performance you were able to obtain and the accompanying algorithm parameters or plot the performance vs. tuning parameters.
- Does your performance match what is reported in the paper? The answer will almost certainly be no. Why do you think this is?

References

[1] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 2765–2781, Nov. 2013.

A Appendix

This is an appendix where you may include your code.