CSE 422: Toolkit for Modern Algorithms

Final Project Guidelines

1 Overview

The final project is worth 30% of your course grade and has three parts: a short proposal, a 5–10 page report, and a 15-minute in-person presentation to Jerry. You may work alone or in a team of two, but **we recommend working in a team of two**.

The goal is to go deeper on one tool from this course than the homework allows. You will either use a technique from class on a real problem (an implementation project) or read and explain a research paper that builds on ideas from class (a theory project). Either way, the report should show that you understand why the method works, not only that it does. Here's roughly a breakdown of what we expect for each type of project:

	Implementation project	**Theory project**
Core task	Take one of the methods discussed in class, and implement it in a real-world setting (perhaps scaled down). Compare how it works with state-of-the-art methods, or perhaps other methods discussed in class.	Understand a theoretical paper's main result and present at least one of its key proofs in your own words. You are not expected to produce a full proof, but you should summarize the key ideas and concepts in the paper. Alternatively, prove something new!
Course connection	Explain which class tool the method relies on (e.g., JL, SVD, multiplicative weights) and why it is the right tool.	Explain which class result the paper builds on and how it goes beyond what we proved in lecture.
Extra submission	A single `.zip` of your code with a README that regenerates every figure and table.	None.

We recommend implementation projects for most students! One high-level goal of this course is to get you some hands-on experience with using these techniques in real-world settings. For both projects, it is not our expectation that you must achieve state-of-the-art numbers or results. A good project is one where both the students and the instructors learn some new insight from reading it/listening to the presentation.

If you are struggling to come up with candidate projects, we recommend taking a look at some of the papers from recent ML/theory conferences such as STOC, FOCS, NeurIPS, ICML, or ICLR. Also feel free to ask the instructors (perhaps at office hours) for suggestions and/or feedback!

2 Deliverables

You submit three things on Canvas (a proposal, a report, and code for implementation projects) and give one presentation.

Proposal (1 page). List your team, the paper you chose, and the specific result you will reproduce or prove. Describe your plan, the extension or added value you intend, and for implementation projects the datasets and compute you will use. Name one risk and your fallback if it happens.

Report (5–10 pages). Use 11-point font, single column, and 1-inch margins; LaTeX is recommended. References do not count toward the page limit. End the report with a short contribution statement for two-person teams and an AI usage note as required by the course AI policy.

Presentation (15 minutes, in person). Plan for about 12 minutes of talk and a couple minutes of questions. Both partners should present. Slides are generally encouraged, however, if you wish, you can also do a whiteboard talk. Expect a few questions at the end.

Code (implementation only). Submit one `.zip` containing your code and a README with setup steps and the command that produces each figure. Do not include large datasets or model weights; link to them instead.

3 Timeline

Proposals are due by 11:59 PM on November 13th and the report is due on 11:59PM on Dec. 11th. Once proposals and teams have been finalized, teams should reach out to Jerry to schedule a time for their presentation during the week of Dec. 7th or the week after.

4 Grading rubric

The project grade is 10% proposal, 45% report, and 45% presentation. You will be graded on the following related criteria:

Correctness and understanding
The report and presentation should demonstrate clear understanding of the material being presented. For implementation projects, the method should be described and implemented correctly, and meaningful experiments and ablations should be done. For theoretical projects, the main conceptual steps and key lemmas are explained clearly.

Depth and contribution
The topic should be interesting and non-trivial, given what we've learned in this class. For implementation projects, this means that the topic is on something timely, and has a clear conceptual connection to what we've covered in class. For theory projects, the topic should be a clear and interesting research-level extension of a topic covered in class.

Writing and clarity

The written report should be written and organized well. The report should clearly specify the problem considered, the motivation, and the proposed method. For implementation projects, the experimental design should be carefully specified and justified, and for theory projects, the main steps of the proof should be clearly spelled out. You may omit the details of the proof, but you should clearly specify the main ideas. You should also clearly cite and discuss the most relevant related work.

Presentation

The presentation should be clear and on time, with both partners speaking and questions on the method (or, for theory projects, the proof) answered well.

For two-person teams, both partners normally receive the same grade unless the contribution statement or presentation shows a large imbalance.

5 Policies

The course collaboration and AI policies on the course website apply to the project.