Spring 2024 CSCI 567 Project Information:

For the course project you will work in teams of 3-4 students (preferably 4 students) on an open-ended project applying, investigating, or extending the ML algorithms discussed in class. Each team will be assigned to a TA mentor to help guide and scope your project. You will submit a project proposal, a midterm report, and a final project report throughout the semester to ensure everyone is on track.

Logistics

You will work in groups of 3-4 (preferably 4) of your choosing. If you would like to work in a smaller group, you can speak to course staff but we would strongly recommend working in groups of 3-4, and our grading criteria will remain consistent regardless of group size. We also encourage you to form groups as soon as possible so you can have enough time to discuss the project ahead of the proposal deadline.

The project consists of 3 graded portions: the project proposal, midterm report, and final report. You will also be required to meet twice with your TA mentor for an initial check-in and midterm check-in that will be scheduled later in the semester but of course our office hours will be open if there are any additional questions outside of those check-ins.

Each proposal and report should be submitted to Gradescope and formatted using this LaTex template. You can make a copy of the template and edit directly with an Overleaf account. You can keep the same document for proposal, midterm and final report and just add or edit the necessary sections.

If your project requires additional compute resources, we would recommend Kaggle, Google Colab, or Paperspace, which all provide access to some free compute resources including GPUs. If you need additional resources, we can make them available on USC CARC, contact your TA mentor if this is of interest (note: you will have to learn how to submit jobs on CARC using Slurm).

Project Types and Sample Topics

The project is open-ended but should relate to topics discussed in class. It should also include some form of research or application novelty whether that is applying an existing algorithm to a new domain or investigating an extension to an algorithm. To help you come up with an appropriate project, we have provided suggestions for project types and some sample topics for each. Feel free to use any of the sample projects or propose your own project. Your TA mentor will also help make sure your project is well-scoped and meets all the criteria.

1. Applying an existing ML method to a new interesting domain or dataset.
    a. Participate in a Kaggle competition, such as this Kaggle competition on predicting sales prices of houses. The competition website provides the dataset and evaluation metric. You should try multiple learning algorithms or variations of algorithms, describe what worked or did not work, and report your final results.
    b. You will probably enjoy the project the most if you pick an application or data that you are excited about. An example project in some application domain can be to predict flood risk in LA. For this particular project, the steps would look like: (1) collecting environmental sensor data such as wind, tide, river level, and water capacity information (this can be scraped online). (2) Collecting this data from the week before to the week after the recent storm. (3) build models to predict the potential flood hazard zone.
2. New empirical analysis on an ML algorithm or problem. This could be comparing different methods, regularizations, datasets, or seeing how a method varies with different hyperparameters, model size, etc.
    a. In deep learning, researchers have developed many forms of regularization including by changing the input data, changing the data labels, or changing the network structure (see this survey paper for an overview and list of recent methods). Pick an interesting dataset (or a few) and explore and compare more complex forms of regularization. How well do they generalize on smaller training datasets? Can different forms of regularization be combined? Are they better suited for certain datasets?
    b. Pick one ML algorithm from class and experiment with ensemble methods through boosting or bagging. How does performance change with (1) the number of models in the ensemble, (2) the accuracy or strength of the base model, (3) amount of training data you have access to? Generally, when are ensemble methods most useful?
3. Extending a research paper: either doing additional analysis or extending the algorithm or results. You should find a paper with existing open source code or is easy to implement, so you do not need to spend too much time reproducing the paper.
    a. K-means is a simple and popular unsupervised method that clusters data into K clusters. X-means (Dan Pelleg & Andrew Moore, 2000), overcomes the need to specify K by dynamically searching for the optimal K and resulting in better clusters. Design and implement an extension to this method and compare the results (this can be one of the extensions suggested in the conclusion of the paper). An extension could aim to improve the accuracy, efficiency, or tackle a new challenge that the original method does not.
    b. In overparameterized models (think large neural networks), generalization can be extra challenging because of the highly non-convex optimization landscape. In Sharpness Aware Minimization (Pierre Foret et al., 2021), the authors propose a method that minimizes loss value and loss sharpness to achieve more generalizable models. The authors provide results for image classification. Do the results extend to other ML problems or architectures? Can this method be

combined with other generalization techniques (they mention several in the introduction)?

c. We know that stochastic gradient descent (SGD) achieves good performances in deep learning. However, as we have shown in the homework problems, SGD may have a slower learning process than full gradient descent. SGD samples one data point per iteration and may suffer from high variance in the gradient, leading to a slower learning process. However, applying full gradient descent is too costly in many deep learning applications which work with large datasets. Stochastic Variance-Reduced Gradient Descent (Rie Johnson & Tong Zhang, 2013) gets the best of both worlds by proposing a method to reduce the variance of SGD to achieve fast convergence. The authors in (Sashank J. Reddi et al., 2016) further propose a mini-batch SVRG version which outperforms GD and SGD in many image classification datasets with simple fully connected neural networks. Do these results generalize to other datasets or neural network architectures? How does SVRG compare with other adaptive SGD methods? How does SVRG compare to SGD with different mini-batch sizes, learning rate, or weight initialization schemes? How should we balance the variance/efficiency trade-off?

4. Other ideas. We are also open to any other project ideas related to course content (for example a theory project), but please check in with a TA or the professor before the project proposal to make sure it is in scope.

Note: Getting negative results is a natural part of any open-ended or research project. We do not expect all projects to achieve positive results in order to get a good grade. It is more important for this course that you understand the results and practice applying ML algorithms. If you are presenting a negative or unexpected result in your final report, this could mean presenting analysis for why something did not work or discussing different approaches you tried and what worked or did not work for your problem.

Project Proposal (due on March 8, more details later):

The project proposal should be 1-2 pages (not including references) and should include the following sections:

- Introduction: Describe the project, research questions, and why this is interesting or novel.
- Related work: Cite relevant papers and explain how they relate to your project or differ.
- Experiment Plan: What kind of data are you collecting or what datasets will you use? What algorithms will you try? What codebases are you using and what will you need to implement? What experiments and analysis will you run? What should be accomplished by the midterm report?

Midterm Report (more details later):

The midterm report should be 3-4 pages (not including references) and should include some preliminary results which may be data collected or experiments run. It should also include what you plan to complete by the final report. If the project topic or direction has changed at all, you should also update the sections outlined in the project proposal.

Final Report (more details later):

The final report should be 5-6 pages (not including references) and should include all results, analysis, and conclusions from your project. This report will be the bulk of the grade for the project.