Final Project Guidelines for STATS 315 Statistics and AI

Yixin Wang
University of Michigan

August 26, 2022

The final project is an integral part of this course. There is no better way to learn about deep learning then by applying it to a research question of your own. The goal of the final project is for you to implement an entire deep learning pipeline yourself: data loading, model design, training, evaluation, and analysis.

You have a lot of freedom in choosing a topic for your final project. The only criterion is that it deeply involves applying deep learning models to a real-world problem. You choose a dataset, an interesting question about it, and address it with deep learning.

The project involves three assignments: project proposal, project milestone, and project report. For each, please use the latex template, a 12-point font, and 1-inch margins. **Page limits are without figures; include as many pages of figures as needed.**

Project Proposal

The project proposal is an abstract that imagines the completed project. We understand that your project will evolve and change over the semester, but writing an abstract early is a good way to plan and think about what it would mean to successfully complete it. (Kaggle is a good place to find interesting datasets.)

We encourage you to refer to computer science conferences (such as *Neural Information Processing Systems* and *International Conference of Machine Learning*) or journals (such as *The Annals of Applied Statistics*, *Journal of the American Statistical Association*, *Journal of Machine Learning Research*) to get a sense of how to write an abstract.

Project Milestone

The project milestone describes the problem you are addressing and discusses some preliminary results. Include what you have completed and what you plan to finish by the end of the semester. As a guideline, the project milestone should (at least) answer the following questions.

- Clarify the purpose of the prediction task:
  - What is the problem of interest?
  - What is the final goal you want to achieve?
- To the best of your knowledge, explain the data as thoroughly as possible:
  - How is the data collected?

  - Is the data a simple random sample from some population?
  - What is the potential defect in the data?
  - If you were able to collect the data, what would you have done differently?
  - Are you going to account for the data collecting process in your analysis?
  - Describe simple facts about the data: size, data structure & source, etc.
  - If you are not going to use real data, explain how you are going to generate "realistic" synthetic data that represents your knowledge of the problem.

- Draft a plan for data analysis:
  - What deep learning model are you going to use for the prediction?
  - Have you done any preliminary analysis of the data?
  - What is your time line for (1) propose and fit a preliminary deep learning model, (2) examine initial results and adjust models further if needed, (3) summarize and write a final report.
  - What are the (potential) difficulties with your data analysis?

The project milestone is 1–3 pages long.

Project Report

The project report is due at the end of the semester. The report should have a similar format to that of an academic paper. Notebooks or markdowns with annotated code which generates the results from the final project should also be submitted as supplementary materials. You can further include a set of appendices (of any length) to which you can banish any details. (We may not read the appendix when grading your work.)

The project report should (at least) include

- **Introduction:** Clear description of the problem. Describe the problem you are trying to solve, why it is important or useful, and summarize any important pieces of prior work that you are building upon.
- **Dataset:** Clear description/visualization of the data. Describe the dataset or datasets you are working with. Show examples from your dataset to give a sense for the types of data it contains. . If you collected or constructed your own dataset, explain the process you used to collect the images and labels, and why you made the choices you did in the data collection process.
- **Method:** Clear and thorough description of statistical analysis. Describe the method you are using; this may also contain parts of the implementation of your model, loss function, or

other components along with sanity checks to ensure that those components are correctly implemented.

You are encouraged to compare multiple architectures. You may need to experiment with different optimizers, learning rates, L2 regularization strengths, and data augmentation strategies. You should train at least 6 different models, and report their accuracy on both the train and validation sets.

You should show training curves for your models. You should plot the training loss per iteration, as well as the accuracy of the model on the train and val sets every epoch (pass through the training data). You may try using additional techniques like test-time augmentation or model ensembling to improve upon the prediction performance of your individual models.

You should also include a table that summarizes the final train and val accuracies of all your models. You can be creative in exactly what models you choose to train.

- **Experiments:** Clear and thorough explanation of experiments and interpretation of results. Describe the experiments you did, and key results and figures that you obtained. This may interleave explanations of the experiments you run and figures you generate as a result of those experiments.

  You should analyze the models you trained to try and gain insight into how well they are working and where they make mistakes. You should analyze at least two different models: your best-performing model and at least one other of your choice.

  For example, you may show qualitative examples of datapoints that were both correctly classified/predicted and incorrectly classified/predicted. You can show both datapoints hand-picked by you to showcase interesting features of your model, and randomly-chosen datapoints to give a better sense of the average performance of your model.

  You may also show confusion matrices of your predictions if you do classification, a 2D matrix of predicted category vs ground-truth category, where each entry shows the fraction of val-set datapoints falling into this situation; this result demonstrates the kinds of mistakes that your model makes.

  You may also look up and use one or more neural network interpretation tools to give examples of what portions of datapoints are used by the model for prediction decisions.

  After all your analysis, you should run your best-performing model on the test set. If your model performs very differently on the val and test sets then you should attempt to explain why that might be the case.

- **Conclusion / Future work:** What did you learn in doing this project? What are the shortcomings or failure cases of your work? If you had more time or resources, how would you continue or expand upon the work you have already done?

You should turn in a .zip file containing both your final report and the notebooks containing all the code and the generated results (tables, figures etc) that are included in the report. You must run all cells in your notebook to receive credit; we will not rerun your notebook.

The project report is 3–5 pages long, excluding figures or references.

The deadline of the final project report is strict, and late days cannot be applied. All late final projects receive a score of zero. In case of requesting exceptions due to severe medical reasons, a doctor's note and a signature from your graduate (or undergraduate) advisor is needed.

Project Evaluation

We evaluate the project on ambition, significance, originality, technical depth, results, relevance, and writing quality. Two good books about writing are Strunk Jr and White (2007) and Williams (1990).

The following grading criteria for the final project report is based on 100% maximum.

- Solid understanding and description of problem (**+5%**).
- Quality of data analysis.
  - Appropriateness of the training of deep learning models (**+25%**).
  - Clarity of comparisons and analyses about the deep learning models trained (**+25%**).
- Presentation of results, including visualization (**+10%**).
- Proper interpretation of results (**+15%**).
- Well-organized and clean code (**+10%**) .
- Clarity of written report (**+10%**).
- Irrelevance with course materials (**-40%**).
- Plagiarism of code or written report (**-100%**).

Policy about Open Source Code

While there are many open source implementations for the problems described below, you should not use them for your implementation. You may refer to existing open-source implementations if you are confused about a specific detail, but you should neither import from nor copy code directly from existing implementations. Your implementation should only use standard libraries for scientific computing in python (e.g. numpy, scipy, matplotlib, pytorch, torchvision, tensorflow, JAX, etc). If you have questions about whether you should or should not use an existing library, please ask about it on Piazza.

References

Strunk Jr, W. and White, E. B. (2007). *The Elements of Style Illustrated.* Penguin.

Williams, J. (1990). Toward clarity and grace. *Chicago: The University of Chicago.*