

On Lab Assignment Submissions, Ethics, and Getting the Most out of Lab

In the lab assignments for this course, we strive to solve hard problems that can have multiple, valid solutions. Typically, you will submit and be evaluated on the following items in each lab:

1. A post-lab write-up justifying your model.
2. A link to your GitHub Repository, so your TAs can execute your code.

This course is foundational to the rest of the program. The assignments in this course are carefully designed and tailored to give you a comprehensive and efficient course of study in machine learning and are designed from the perspective that your time is valuable. You will never be asked to complete busy work in this course.

The grade weighting breakdown for the laboratory assignments is as follows:

- Working Code: 50%
- Code Style: 20%
- Post-Lab Report (1 Page Maximum): 30%

Post-Lab Report

Describe your model and then justify why it is a good model for solving the problem. Provide a rationale for each decision you made in designing your model (why you incorporated x , y , z into your model, why you chose a particular value for a parameter, etc). and provide appropriate evidence to support your claims (accuracy measurements, figures, etc.).

Make this argument as brief and as accessible as possible. Always write any technical documentation with your audience in mind. In this and all lab assignments, you should strive to evaluate your writing and figures against the metric of, "if someone who had taken this course reads this document, would they be able to easily understand everything that happened after a quick scan of the document?" Your ability to communicate clearly and succinctly is prized and will be prioritized in these lab assignments. It is not sufficient to simply do the technical analysis--you must also be able to communicate your analysis and ideas accessibly to others.

If a lab assignment has reflection questions, ensure that they are also answered succinctly in your report.

Your Code

You should complete your programming assignments in the Coursera Labs environment. You will generate a read-only link to your lab environment that will allow TAs to read and run your code by clicking on a button on the lab home page.

It is critical that you prioritize the readability of your code and that you annotate your code thoughtfully. Your evaluation will consider the readability of your write-up, any figures you make, and your code.

These labs are designed to expose you to a deeper understanding of machine learning. As a result, you should not use Python libraries like Scikit-Learn (sklearn), PyTorch, or TensorFlow to solve these problems unless otherwise directed. Later in the course, you can use these libraries--but only once you have demonstrated that you understand their underlying operation. We will let you know explicitly when it is ok to use them to solve a problem.

As always, if you reference any external sources, cite them appropriately. In the interest of furthering your understanding of the material, refrain from using generative AI to code for you. Your work should be your own, and you should feel comfortable justifying each design decision you make.

Criteria for Success

All lab submissions are individual; every item you submit should reflect your work. While your work is your own, we understand that it can be helpful to collaborate with your peers, ranging from high-level discussion of a problem to implementation and debugging. Having others look at your code encourages you to write code with readability in mind. In practice, you will never work in a silo, and being able to discuss these topics with others well is a valuable skill. When you collaborate with another student, please cite them appropriately and be respectful of not sharing too much. You should never copy someone else's code.

While there are diminishing returns when struggling with a problem for too long, there is value in struggling with tricky concepts on your own for a while. As always, the Academic Honor Principle applies and should be your guiding principle in working with others. Try to make progress on the problem yourself before discussing it with others.

These lab assignments are complex and may require significant time debugging. Having bugs in your code or conceptual questions is normal, so you should budget your time accordingly. You will succeed more in the labs if you start early enough to get help from your peers, your TAs, and your course instructor when you need it. Ask questions early and often.

Finally, **please have fun with these labs!** We do some of our best learning when we can deeply engage with the material. While we strive to have a deep understanding of the underlying mechanics of machine learning, we hope that you find it still magical to see your algorithms work to solve complex problems. I hope that by the end of this course, you feel the same quiet reverence and awe for the underlying mathematics--and that you are excited for your future progression in this program as you consider classes like Applied Natural Language Processing and other courses that build on this material.