CS 181 – Practical Logistics

What should I turn in via Canvas? The main deliverable of this practical is a three-to-four page typewritten document in PDF format that describes the work you did and what your approach was. This document should contain the following headings:

1. **Technical Approach:** How did you tackle the problem? Credit will be given for
   - Diving deeply into a method (rather than just trying off-the-shelf tools with default settings) and making tuning and configuration decisions using thoughtful experimentation.
   - Exploring several methods, perhaps going beyond those we discussed in class.

   In either/both routes, thoughtfully iterating on approaches is key. If you used existing packages or referred to papers/blogs for ideas, you should cite these in your report.
2. **Results:** Did you create and submit a set of predictions? Did your methods give reasonable performance? Credit will be given for quantitatively reporting (with clearly labeled and captioned figures and/or tables) on the performance of the methods you tried compared to your baselines. You must have *at least one plot or table* that details the performances of different methods tried. You will not be graded in proportion to your ranking; we'll be using the ranking to help calibrate how difficult the task was and to award bonus points to those who go above and beyond. *However*, you must at least clear any sample baseline scores shown on the Kaggle leaderboard to earn full points.
3. **Discussion:** Do you explain not just what you did, but your thought process for your approach? Credit will be given for
   - Explaining the your reasoning for why you seqentially chose to try the approaches you did (i.e. what was it about your initial approach that made you try the next change?).
   - Explaining the results. Did the adaptations you tried improve the results? Why or why not? Did you do additional tests to determine if your reasoning was correct?

How will my work be assessed? This practical is intended to be a realistic representation of what it is like to tackle a problem in the real world with machine learning. As such, there is no single correct answer and you will be expected to think critically about how to solve it, execute and iterate your approach, and describe your solution. The upshot of this open-endedness is that you will have a lot of flexibility in how you tackle the problem. You can focus on methods that we discuss in class, or you can use this as an opportunity to learn about approaches for which we do not have time or scope. You are welcome to use whatever tools and implementations help you get the job done. Note, however, that you will be expected to *understand* everything you do, even if you do not implement the

low-level code yourself. It is your responsibility to make it clear in your writeup that you did not simply download and run code that you found somewhere online.

Bonus Points: The top three teams will be eligible for extra credit. The first place team will receive an extra five points on the practical, conditioned on them giving a five-minute presentation to the class at the next lecture, in which they describe their approach. The second and third place teams will each receive three extra points, conditioned on them posting an explanation of their approach on Piazza.

What language should I code in? You can code in whatever language you find most productive. We will provide some limited sample code in Python. You should not view the provided Python code as a required framework, but as hopefully-helpful examples.

Can I use {**scikit-learn** | **pylearn** | **torch** | **shogun** | **other ML library**}**?** You can use these tools, but not blindly. You are expected to show a deep understanding of the methods we study in the course, and your writeup will be where you demonstrate this.

Can I have an extension? There are no extensions to the Kaggle submission and your successful submission of predictions forms part of your grade. Your writeup can be turned in up to a week late for a 50% penalty. There are no exceptions, so plan ahead. Find your team early so that there are no misunderstandings in case someone drops the class.