COMP 432: Machine Learning, Concordia University

**PROJECT PROPOSAL FORM**

This document serves as a template that you can either fill out and submit, *or* use as a guideline / checklist for writing your own project proposal document. Remember, this is just a proposal to demonstrate that you have put time into planning, but you are allowed to change your plans for the final project if you run into trouble or change your mind. The guidelines (“Aim for X sentences”) are flexible, not hard constraints.

Group: \_\_\_\_\_\_ (e.g. G00)

Student 1: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student 2: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student 3: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (only by permission of instructor)

Date submitted: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Propose a title for your project.** If you had to guess what title you would give your project, what would it be? A good title provides a clue to the main goal and/or main method that the project focuses on. For example, the paper [*Intriguing properties of neural networks*](https://arxiv.org/abs/1312.6199) (Szegedy *et al*. 2014) has a title that, although a little vague, at least suggests that the nature project was an investigation of some kind, focusing on “neural networks,” and that the results are surprising. As another example is the paper [*The fastest pedestrian detector in the West*](https://authors.library.caltech.edu/94253/) (Dollar *et al*. 2010), whose title indicates that the goal is “pedestrian detection” and that the nature of the contribution is “speed.”

[Type your answer here.]

**Describe the goal of your project.** What are you trying to achieve? What “main question” are you trying to answer, or at least to provide evidence for? Secondary goals are OK, but you should still have a clear “main goal” or “main question.” From your description, it should also be clear whether your project is about: making better predictions for some application? speeding up training and/or predictions? simply comparing predictive performance and/or speed of several methods? assessing or comparing interpretability? understanding failure modes or sensitivities of some methods? Etc.

[Type your answer here. Aim for 3-6 sentences.]

**Describe the data you plan to use.** One of the hardest steps for a good machine learning project is to find data that is truly suitable for your goals. Finding good data not the most fun part, but it’s one of the most important—after all, for machine learning it is “garbage in, garbage out”. Here are some things you should ideally know:

* What are the ‘modalities’ that apply to the data? (images, video, speech, text, tabular, categorical, numerical, time series, experimental measurements, etc.)
* What does an input look like? (show an example if possible, like an image, or a sound wave, or some features, or at least try to describe)
* For an example input, what does a desired output look like? (show an example if possible, or at least try to describe)
* How many training and testing samples will there be?
* How are the training and testing data to be split? (randomly shuffled, by some grouping, by time period, etc.)
* Will the data need preprocessing before you can feed it into a training algorithm? (It is OK if you are not sure, but try to guess)
* Is the data small enough to train models on your computer, or is there a risk of scalability/engineering difficulties?

[Type your answer here. Be concise. Aim for half a page.]

**Describe how you will measure “success.”** You should explain how you will know whether you have achieved the goal(s) that you described earlier. What does “success” look like? What does “failure” look like? Keep in mind that your project can still succeed (in the sense of a good grade!) even if the experimental results are bad—what is important is that your experimental results are *conclusive*! A bad project is one in which you cannot even tell whether the goal was achieved or not.

[Type your answer here. Aim for 3-6 sentences.]

**Describe how work will be divided.** It is very important for everyone to have a meaningful role in the project. If one person (the most experienced person) does all the programming or writing, then everyone else in the group loses this important chance to gain experience. For example, if there is no way to “happily divide” the work because two group members want to work on the same part, that is totally OK and no one should feel guilty for wanting that; both group members can do their own version of that part of the project, and then the final report can say “two group members each implemented did this part, and their results {matched, didn’t match}” When two people attempt and come to different conclusions, that is interesting and a chance for everyone to learn!

[Type your answer here. High-level description only, like “Angela will train the neural networks, and Seyyed will preprocess the data and train the SVM. Both will write the report.”]

**List the main Python packages you expect to use.** PyTorch? TensorFlow? Scikit-learn? Special packages for working with your data? (It is OK if this list is incomplete or changes for the final project.)

[Type your answer here.]

**Frequently asked questions.** Below are some questions students have asked.

*Q: Can we use a pre-trained model, or do we have to train a model ourselves?*

A: This question applies mainly to computer vision or natural language processing. Your project must involve training one or more models. However, you are definitely allowed to incorporate pre-trained models in that effort. For example, if you download a pre-trained model and then use it to convert your raw training data into more a more useful representation, then that is OK, but (a) you should still train new models on top of that representation for the task you are trying to solve and (b) you should consider training simple baselines (e.g. a linear model, a random forest) on your raw data, to demonstrate that the pre-trained model was important for performance.

*Q: Are bigger groups expected to do more ambitious projects?*

A: No. But in bigger groups there is a higher chance that the least-experienced group member will be “left out” of important programming or writing activities. So please be conscious of this and give everyone a chance to learn.

*Q: Can we try machine learning algorithms beyond what we learn in the course?*

A: Yes absolutely. If you want to try reinforcement learning, that is OK. But you should still try your best to apply some of the methods we’ve learned about—even if they are not a natural fit to your “task” and you expect them to perform poorly, you should try. This can also be useful for dividing work among group members: some can try to apply fancier methods outside the course, where other group members can try to apply the basic methods even if the results are not expected to be “state of the art.”

*Q: How do we find data?*

A: Here are some thoughts:

* Google something you are interested about, like “climate change datasets for machine learning” and you may get lucky.
* You can try to look at machine learning data set repositories, such as the UCI machine learning repository or the OpenML repository.
* You can use or create synthetic data, generated by simulators or other software that you are capable of running on your computer. For example, if the main question of your project was “can we use ML to approximate the output of <insert software here>” then you may pursue this approach. (However, notice that basic machine learning algorithms tend to produce fixed-dimensional outputs, whereas most software produces variable-length outputs, so defining a good project along these lines takes time!).
* You can look for “challenges” or “competition tracks” that have been hosted as part of machine learning conferences, or as part of a Kaggle challenge.

Be careful about the size of the data sets. If you choose to train a model directly on a huge data set (a huge text corpus, or 3D medical images, or the ImageNet dataset), then it is very likely that some group members cannot participate in the training because they do not have GPUs, and so most group members do not have a good experience or are stuck waiting.