CS 475/675 Machine Learning: Project

This semester you will complete an independent project to learn how machine learning can be applied in practice. You will work in groups of 3-4 students, and all students in the group will receive the same grade. The final project will be worth 20% of your final grade (200 points).

You have a choice between two types of projects.

- **Application**: These projects will apply machine learning methods to a dataset.
- **Methods**: These projects will conduct an analysis of two or more machine learning algorithms.

Additionally, each project will select a specific area: either an application domain or a subfield of ML methods. Each team will be assigned to a TA/instructor who will provide feedback and review project deliverables.

Project Structures

Your first goal will be to form a team of 3-4 students. You will share your grade with your team members, so pick people with whom you can work well. We recognize that given the physical distance between students, and time zone issues, some students may be unable to form teams of 3-4 students. We will consider a limited number of exemptions that allow students to work by themselves or with only a single partner. Contact cs475@cs.jhu.edu for an exemption.

The final project has several milestones and deliverables.
1) Project proposal (20 points): Due **Thursday November 5th (11:59pm)**
Your project proposal should be about 1-2 pages. Use the provided latex template here.

2) Project Update (10 points): Due **Thursday November 19th (11:59pm)**
Your project update should be about .5 pages. Copy your proposed list of deliverables and write a 1-2 sentence update on your progress for each deliverable. We expect that at this point you've at least processed and explored your data. Hopefully, you will have completed at least one of your “must accomplish” deliverables.

3) Final presentation (60 points): **Either Mon December 7 or Wed Dec 9** (in class)
Each team will present a ~10 minute presentation on their project in a breakout room during class time. Students in timezones incompatible with the live lecture will submit recordings of their presentations. Include an update on your deliverables: are you on track to accomplish everything you proposed?

4) Project git repo and Jupyter Notebook (110 points): **Monday December 14 (11:59pm)**
The final deliverable for your project will be a git repository containing all of your code and a Jupyter Notebook containing the final writeup. The git repo can be stored on Github.com or Bitbucket.com, and you will provide a link to the repository. You will also provide a link to a Jupyter notebook. The notebook can be stored directly in the git repo, or hosted on Google

Colab. The notebook should be structured as a writeup, with text explanations mixed with a step by step walkthrough of the project, including summaries of the methods, goals, figures, and results. We will provide a Notebook template you should follow.

Application Project

For the application project, you will select one of five application domains. We will require you to evaluate on a dataset from that domain, but you are encouraged to expand beyond the provided dataset. Your goal will be to propose an application that requires machine learning using the dataset. The novelty lies in the pipeline that you develop, and you will be focusing on the various aspects of developing a complete machine learning solution to automate a real-life need. This will include, formulating a task, selecting data, feature engineering, determining the best features for your model, implementing a model, determining hyperparameters and testing your model for problematic behavior. You can use any software library for this task. The way you process your data, and the choice of models will depend on the domain that you're working on. You will be judged based on the work you put into the project beyond the available software libraries. You can think of the application project as an expanded Lab 1 homework. While in the lab we asked you to develop a dataset and test various properties of it, in the application project you'll go well beyond this by developing the application and trying different solutions. Unlike the lab, you do NOT need to make your own dataset.

Each group should select a dataset from one of the following domains.

1. Medical informatics and Genetics
   - UK Biobank https://www.ukbiobank.ac.uk/data-showcase/
   - Genotype-Tissue Expression https://www.gtexportal.org/home/datasets
   - Protein Classification http://scop.mrc-lmb.cam.ac.uk
2. Healthcare delivery and public health
   - COVID cases and deaths: https://ourworldindata.org/covid-deaths
   - WHO Statistics: https://apps.who.int/gho/data/node.resources
   - CDC Statistics: https://data.cdc.gov/browse
3. Textual analysis
   - Wikipedia articles: https://en.wikipedia.org/wiki/Wikipedia:Database_download
   - Sentiment prediction: http://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences
   - 20newsgroups: https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html
   - Translation of Canadian Parliament: https://www.isi.edu/natural-language/download/hansard/
   - Hate speech on Twitter https://data.world/thomasrdavidson/hate-speech-and-offensive-language
4. Image analysis
   - Facial recognition: http://vision.ucsd.edu/content/yale-face-database

    - Art objects (paintings, sculptures, drawings etc) from the Rijksmuseum museum in Amsterdam: https://figshare.com/articles/Rijksmuseum_Challenge_2014/5660617
    - Satellite Imagery for Natural Disasters: https://www.digitalglobe.com/ecosystem/open-data
5. Finance and economics
    - World Bank Open Data: https://data.worldbank.org/
    - International Monetary Fund Data: https://www.imf.org/en/Data
    - Quandl: https://www.quandl.com/ (Make sure to select free data)
    - Stock prices from pandas_datareader

Methods Projects

We have learned about several algorithms and the assumptions behind them. We also observed that these observations are not valid in real data. For example, the assumption of independent and identically distributed observations does not hold for time series data. Further, it is possible that there are behaviors other than high prediction accuracy that a real world application might need. For example, you want your classifier to be fair to individuals who belong to a minority group. Real data also requires coming up with smart ways to deal with noise, missingness and unknown confounders. Additionally, you might care about other model requirements like interpretability: why did the model predict a certain outcome?

A methods project will select from one of the following topics. Your task will be to modify a method, such as creating a new loss function, constraints, data preprocessing, etc. with the goal of solving one of the tasks. You will demonstrate that your model improved on the desired task, and can be used to achieve your stated goal, on selected datasets, either simulated or real data depending on your project needs. You **must** evaluate your method on one of the simple datasets contained within sklearn.datasets, but you are encouraged to also use a real-world dataset to explore a more interesting application. The exact evaluation will depend on your proposed method. You do not need to implement the methods from scratch; you will be allowed to use existing software libraries. However, we will judge you based on the amount you do beyond existing implementations. Additionally, you should compare your method to an existing method, such as a classifier from Scikit Learn.

A methods project will be similar to some of the breakout rooms we did in class. Those breakout rooms focus on explorations of a method, and tested it in various ways. Your methods project will do something similar but go beyond these limited tests to propose a specific method for one of the topics below, then explore that method and show its efficacy.

A methods project should select from one of the following topics.

6. Fairness in Machine Learning

    - A project that demonstrates fairness aspects of a model or dataset, and proposes methods to improve fairness in a measurable way

7. Interpretable Machine Learning
    - Methods for interpreting trained ML models
8. Graphical models / Structured prediction
    - Develop a new graphical model, or develop a model for a structured prediction task
9. Robust Machine Learning
    - Learning in the presence of outliers or adversarial examples. For example, see https://jerryzli.github.io/robust-ml-fall19.html
10. Private machine learning
    - Develop an algorithm that respects the privacy of users whose data appears in the training set. For example, see https://medium.com/dropoutlabs/privacy-preserving-machine-learning-2019-a-year-in-review-123733e61705