Final Project

CS-UY 4563 - Introduction to Machine Learning
Spring 2023
Due Date: To Be Announced*

For your final project, you will work with one other student (from either class) to apply the machine learning algorithms and techniques you learned in class to a task/problem of your choosing. You and your partner will present your project to the class at the end of the semester.

You and your partner need to submit the following documents to Gradescope.

- Your presentation slides
- Your project write-up as a PDF
- Your project code (GitHub link is ok)
- Your dataset - if you made one from scratch

Please note that we will be passing your code and write-up through a plagiarism checker. We know there are ways to cheat on the final project. If we suspect you of cheating, you will receive 0 for your final project grade. See the syllabus for additional penalties that may be applied.

The following are some links you can search for a dataset. Feel free to use any other datasets or *make your own*:

- `https://vision.eng.au.dk/plant-seedlings-dataset/`
- `https://www.kaggle.com/competitions`
- `https://www.openml.org/`
- `https://paperswithcode.com/datasets`
- `https://registry.opendata.aws/`

*The projects will be due the evening before the first presentations are given. The date the project will be due depends on the number of presentations.

- https://opendatamonitor.eu/frontend/web/index.php?r=dashboard
- https://dataportals.org/
- https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research
- https://www.reddit.com/r/datasets/
- https://www.quora.com/Where-can-I-find-large-datasets-open-to-the-public

Please enter on Brightspace a link to the data set you will use and who your partner is by March 26th.

Guidelines for your writeup:

1. **Introduction:** You will briefly describe your data set and the problem you are trying to solve.

2. **Perform some unsupervised analysis:** Look to see if any interesting structure is present in the data. If you don't find any interesting structure, describe what you tried.

3. **Supervised analysis:** You must train at least three of the learning models discussed in the class (e.g., Logistic regression, SVM, Neural Networks).[1] For each model, you must try different *feature transformations* and different *regularization techniques*.[2] For example, try the linear, polynomial, and radial-basis function kernel if you use support vector machines in your project. Remember to illustrate (through graphs) how your feature weights and error changed when you used different parameters, regularizations, and normalizations.

4. **Table of Results:** You *must* create a table that contains the final results for your model. It would be useful to have a table that includes the *training* accuracy and the *validation* accuracy for every model you created. For example, suppose you're using Ridge Regression and manipulating the value of λ. In that case, your table should contain the training and validation accuracy for every lambda value you used. You should provide the precision and recall

5. **Why:** Your write-up should analytically discuss your experimental findings; and what conclusions you can draw from your work. You should appeal to the concepts discussed in class: *overfitting, underfitting, variance, bias, etc.*

You and your partner will give a six-minute presentation to the class. The final project presentations will be held during the last 1 or 2 class periods and during the final exam period for this class. You will be assigned a day for your presentation. If we run out of time the day you are to present your project, you will present the next day reserved for presentations.

You are required to watch the other students' presentations in class. A large part of your project grade will be based on your attendance for everyone else's presentation.

If you have a project idea that doesn't satisfy all the requirements mentioned above, please inform me, and we can discuss its viability as your final project.

[1] If you wish to use a model not discussed in class, you must discuss it with me first, or you will not receive any points for that model.

[2] Even if you get a very high accuracy, perform these transformations to see what happens.

If you use techniques not covered in class, you must demonstrate your understanding of these ideas.

Practical advice can be found in chapters 1 and 2 of Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, and in http://cs229.stanford.edu/notes2020fall/notes2020fall/CS229_ML%20advice_presented-slides.pdf Please Google appropriate topics. A quick first glance at some of these topics can be found here:

- Dealing with unbalanced datasets: https://www.svds.com/tbt-learning-imbalanced-classes
- Preparing your dataset: pages 62-69 Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow
- Working with time series: https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4.htm and search for time series
- Handling the missing features in the training set: see lecture 16 in https://web.stanford.edu/~lmackey/stats306b/ https://machinelearningmastery.com/handle-missing-data and https://machinelearningmastery.com/statistical-imputation-for-missing-values-in-ma