Instructions for Assignment 1
ECE 310 - Machine Learning Engineering Practicum
Fall 2023-2024

1 The Dataset

Your dataset appml-assignment1-dataset-v2.pkl contains[1]:

- historical exchange rates between various currencies and the US dollar on a major trading platform (precisely midpoints between the bid and ask),
- during the trading day trading prices for exchange traded funds that attempt to track stock markets in the associated countries
- an index that tracks a collection of US treasuries, and its volatility, and the S&P 500 index

For each such index/ETF/exchange rate, each of the following are provided hourly

- The "open" price – which is the price the item began the hour at
- The "high" price – which is the highest price the item reached during the hour
- The "low" price – which is the lowest price the item reached during the hour
- The "close" price – which is the price the item ended the hour at

NOTE: When an item was not trading during an hour the information from the most recent hour it traded is included. These items are included in the pandas dataframe X. In addition to these the pandas dataframe y contains the high for the (midpoint between bid and ask) for the Canadian dollar exchange rate (i.e. how much 1 USD is in Canadian dollars) from the subsequent hour.[2]

2 Your Task

Leveraging the scikit-learn library, you are to train a model to predict the next hour's high of the Canadian dollar exchange rate y from only the previous hour's data X. In particular, you must create and submit code to create some basic descriptive plots, and model generation code. The model generation code must include

1. Performing a train-test split of the data.
2. Building a pre-processing pipeline which takes as its input data of the form of X and performs all of the following steps
   (a) Fills in any missing values in the numerical features in X and standard scales them to have zero mean and unit variance.
   (b) Replaces the date feature in X with two categorical features - a 7-valued categorical feature indicating the day of the week, and a categorical feature indicating the hour of the trading day.
   (c) One-hot encodes the newly augmented categorical features.
3. Create a model to predict y from the output of your pipeline, fit it on the training data, and evaluate it on the testing data.
4. Save the trained model and your pipeline into files as specified below.

Additional, the basic descriptive plot generation code must

1. load one of your saved pipelines and the data from the pickle.
2. process the entire original data with your pipeline

[1] Use pandas' read_pickle method to read the pickle then extract the variables X and y from the python dictionary it returns.
[2] There are a ton of obvious things that this leaves out – notably (1) sovereign debt yields in the foreign countries, (2) some measure of corporate debt yield, (3) prices of related swaps and derivatives. This is not a proper economics assignment.

3. create a new dataframe containing the day of week and hour of day variables and the difference between the CAD-close attribute (representing the previous hours' closing value) and y, the CAD-high from the next hour (i.e. subtraction).

4. Use pandas' group by and agg functions with appropriate functions from pandas.Series to evaluate the mean, 5th, 10th, 25th, 50th, 75th, 90th, 95th percentiles, and root-mean-square values of the newly created difference attribute as a function of the (day of week, hour of day) pair. Plot these on a common axis using matplotlib.pyplot, including descriptive axis labels, axis ticks, and a legend. Save this plot to a png file with the name cad-change-stats.png.

2.1 Key considerations and other details

- The dataframes y and X are already aligned so that that in a given row, y contains the next hours high (i.e. the label to predict), and X contains the current hours information. Whatever features you calculate to predict a row in y must come exclusively from the same row in X. None of the operations in your code should be reordering the rows in X and y.
- Your pipeline may, at your option, include other steps in an attempt to improve your model's performance, for instance by augment with additional computed features, but the steps above at the required ones.
- Your pipeline must be fit on the training portion of your data and must transform both the training and testing portions.

3 How you will be graded

Some of the data I have not shared with you. As explained in *how to submit your assignment* below, you can submit up to two models, both of which must be accompanied with the data transformation pipeline you have fit to your data. I will first transform the heldout data using the .transform method of your supplied transformation pipeline, then predict the associated CAD-high for the next hour for it using the .predict method of your supplied model. I will measure a mean squared error, repeating this process for both of your two supplied (model, transformation pipeline) pairs, keeping the lower of the two mean squared errors. I will map this MSE to a score for your model's relative performance by comparing it to the MSEs obtained by your peers - 30 out of 100 points will be this score. The remaining 70 points will be generated by assessing whether you are compliant with the instructions in *How to submit your assignment* below, and by reviewing your code to determine if it correctly performs each of the steps identified in the section *Your task.* above.

4 How to submit your assignment

Your submission should be gzipped or zipped archive titled **abc123-lab1.tgz** or **abc123-lab1.zip** (with abc123 your drexel username). The archive must contain at least the following 6 files with the following names

- code1.py – a simple text file containing the python code you used to create your first transformation pipeline model (cut and paste from your jupyter notebook as necessary)
- pipeline1.pkl and model1.pkl – a joblib/pickle for your first transformation pipeline and first model respectively.
- code2.py – a simple text file containing the python code you used to create your first transformation pipeline model (cut and paste from your jupyter notebook as necessary)
- pipeline2.pkl and model2.pkl – a joblib/pickle for your second transformation pipeline and second model respectively.
- cad-change-stats.png – the plot showing descriptive statistics of the change between the previous hours close and the next hour's high of the CAD-USD pair as a function of the hour of day and day of week.
- plotGeneration.py – a simple text file containing the python code you used to create and save the descriptive plot cad-change-stats.png (cut and paste from your jupyter notebook as necessary).

You must submit the zipped or gzipped file within BBlearn before the assignment's due date. NOTE: I must be capable of regenerating your model using the code you supplied.