Udacity Machine Learning Nanodegree Capstone Project Proposal: Stock Prediction
Date: 09-19-20
Author: John Farrell

Domain Background

Prediction of stock trends has long been of interest to investment and trading firms around the world. Entire professions are dedicated to analyzing stock trends and valuations in order to predict which direction that stock will move in the next day, month, or year(s).

Problem Statement

Stock prediction is difficult. With the advent of machine learning techniques coming to the forefront in the last decade, we can now attempt to use computational power to better predict the direction of stock movements. In this project I will be using Amazon's DeepAR algorithm to attempt to predict stock prices of a select few stocks for a given forecasted time period.

Specifically, I will only be focused on predicting adjusted close prices for stocks. Adjusted close prices are closing stock prices at the end of each trading day that account for any corporate action such as stock splits, dividends, or rights offerings. Thus, we can compare daily closing stock prices of a stock like General Electric (GE) that pays regular dividends and has split 7 times since it started trading publicly.

Datasets & Inputs

I will be using an API to capture data provided by quandl. Free stock prices of over 2,000+ stocks are provided up until 2018. Open, high, low, and close prices, adjusted prices, volumes, and split ratios are all provided. As noted above, we'll only be focused on adjusted closing prices for this project. We'll use our API key to pull the data from quandl's database. The pulled data is already tabularized into a dataframe for us. See below for an example of an output of the data:

Date	Open	High	Low	Close	Volume	Ex-Dividend	Split Ratio	Adj. Open	Adj. High	Adj. Low	Adj. Close	Adj. Volume
1962-01-02	75.00	76.2500	74.25	74.75	21600.0	0.0	1.0	0.329505	0.334997	0.326210	0.328407	2073600.0
1962-01-03	74.38	74.3800	73.75	74.00	14800.0	0.0	1.0	0.326781	0.326781	0.324014	0.325112	1420800.0
1962-01-04	74.00	74.6200	72.50	73.13	18400.0	0.0	1.0	0.325112	0.327836	0.318522	0.321290	1766400.0
1962-01-05	73.13	73.2500	70.00	71.25	27300.0	0.0	1.0	0.321290	0.321817	0.307538	0.313030	2620800.0
1962-01-08	71.25	71.2500	69.00	71.13	31000.0	0.0	1.0	0.313030	0.313030	0.303145	0.312503	2976000.0
...	...	...	...	...	...	...	...	...	...	...	...	...
2018-03-21	13.66	13.9600	13.57	13.88	64989359.0	0.0	1.0	13.660000	13.960000	13.570000	13.880000	64989359.0
2018-03-22	13.75	13.7900	13.32	13.35	70929333.0	0.0	1.0	13.750000	13.790000	13.320000	13.350000	70929333.0
2018-03-23	13.40	13.4499	13.02	13.07	82930120.0	0.0	1.0	13.400000	13.449900	13.020000	13.070000	82930120.0
2018-03-26	13.23	13.2395	12.73	12.89	101095809.0	0.0	1.0	13.230000	13.239500	12.730000	12.890000	101095809.0
2018-03-27	12.92	13.7200	12.82	13.44	153476613.0	0.0	1.0	12.920000	13.720000	12.820000	13.440000	153476613.0

Solution Statement
I will run the Amazon Deep AR algorithm on a portfolio of stocks to predict future prices at a prediction length of 1 to 30 days. First, we'll train our model using data from previous years (e.g. 2015 to 2017) leaving out the last *x* days, where x is the prediction length. This will be our 'test' data set that the model can validate the training job against. The model will then be used to test against a "future" time series (e.g. the first trading days in 2018). I will use the benchmark model and evaluation metrics detailed below to determine the accuracy of our model's prediction at that point.

Benchmark Model
For this project, I'll use a Naive method to compare our model against. A Naive method follows a model that forecasts for every new time period correspond to the last observed value. It is described by the following equation[1]:

$$\hat{Y}(t+h|t) = Y(t)$$

Evaluation Metrics
I will use the mean absolute percentage error (MAPE) to evaluate the accuracy of my model and the Naive method. The MAPE provides a suitable indication of how well the model predicts the output over the entire prediction time horizon. It is calculated as follows[2]:

$$\mathrm{M} = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|,$$

where A_t is the actual value, F_t is the forecasted value, and n is the number of periods.

Outline of Project Design
In general, I will be looking to obtain historical stock data using an API through quandl's databases. This data will then need to be preprocessed in order to extract the adjusted closing prices and affiliated timestamps. From there, I will develop time series for each stock and convert into a JSON object, as that is the data format that the DeepAR algorithm accepts. I'll likely play around with prediction length, context length, and hyperparameter variables to settle on a model that predicts with the highest accuracy. I can ultimately use SageMaker's hyperparameter tuning capability to identify the "optimal" hyperparameters for this particular model.

Using the model's deployed predictor, I'll then predict outputs using our provided time series.These predictions will have to be decoded from JSON to a meaningful predicted adjusted stock price over time. I can display the mean predicted quantile of each stock of interest along with 10% and 90% quantile bounds over the prediction window.

Lastly, I'll then have to test the accuracy of the model further by testing it against data that it hasn't seen before. I'll measure the MAPE of the output as well as the Naive method to compare the accuracy of our model.