Machine Learning Engineer Nanodegree

Capstone Proposal

Johnathon Schultz
September 3rd, 2018

Proposal

Domain Background

Predicting stock performance is a task most individuals and corporations have attempted to achieve at some level of accuracy since the initiation of exchanges. The challenge of such prediction is the speed with which data stagnates. For example, a piece of information may only be valid short term, until a new piece of information is received. To help mitigate this problem, folks have looked to correlate data available at a similar large volume and stagnation rate to that of stock price performance. One such data source is news analytics.

The ubiquity of data today enables investors at any scale to make better investment decisions. The challenge is ingesting and interpreting the data to determine which data is useful, finding the signal in this sea of information. By analyzing news data to predict stock prices, we have a unique opportunity to advance the state of research in understanding the predictive power of the news. This power, if harnessed, could help predict financial outcomes and generate significant economic impact all over the world.

Here are some example attempts at tackling this problem:

- https://medium.com/mlreview/a-simple-deep-learning-model-for-stock-price-prediction-using-tensorflow-30505541d877
- https://towardsdatascience.com/stock-prediction-in-python-b66555171a2
- https://www.quantinsti.com/blog/machine-learning-trading-predict-stock-prices-regression/

Problem Statement

I will predict a signed confidence value, $\hat{y}_{ti} \in [-1, 1]$, which is multiplied by the market-adjusted return of a given assetCode over a ten day window. If I expect a stock to have a large positive return–compared to the broad market–over the next ten days, I will assign it a large, positive confidenceValue (near 1.0). If I expect a stock to have a negative return, I assign it a large, negative confidenceValue (near -1.0). If unsure, I assign it a value near zero.

Datasets and Inputs

For this project I have two data sources:

1. Market data (2007 to present) provided by Intrinio–contains financial market information such as opening price, closing price, trading volume, calculated returns, etc.
2. News data (2007 to present) provided by Thomson Reuters–contains information about news articles/alerts published about assets, such as article details, sentiment, and other commentary.

Each asset is identified by an assetCode (note that a single company may have multiple assetCodes). Depending on what you wish to do, you may use the assetCode, assetName, or time as a way to join the market data to news data.

To prevent lookahead bias, the data will be split into two sets: Train (2007 - 2017) and Test (2017 - present).

Solution Statement

In an effort to make a reasonable prediction for future stock prices, I am leveraging stock trading data in addition to news and commentary. By correlating a dataset similar to stock trading data in terms of volume, frequency and stagnation rate, I hope to improve the predictability from that of a naïve model.

Benchmark Model

As a benchmark, the first model will employ a naïve approach of future price prediction. The algorithm I intend to use for benchmarking will be to predict uncertainty for the future timeslots for all input values. Thus, the output of the model for all assetCodes over a ten day window will be a confidenceValue of 0.0, indicating uncertainty of

positive or negative return, on a range of [-1.0, 1.0]. Such a prediction would indicate the market price for the stocks remain unchanged over the prediction period, which is what we will use as a baseline, or benchmark model.

Evaluation Metrics

For each day in the evaluation time period, calculate:

$$x_t = \sum_i \hat{y}_{ti} r_{ti} u_{ti},$$

where r_{ti} is the 10-day market-adjusted leading return for day *t* for instrument *i*, and u_{ti} is a $^0/_1$ universe variable that controls whether a particular asset is included in scoring on a particular day.

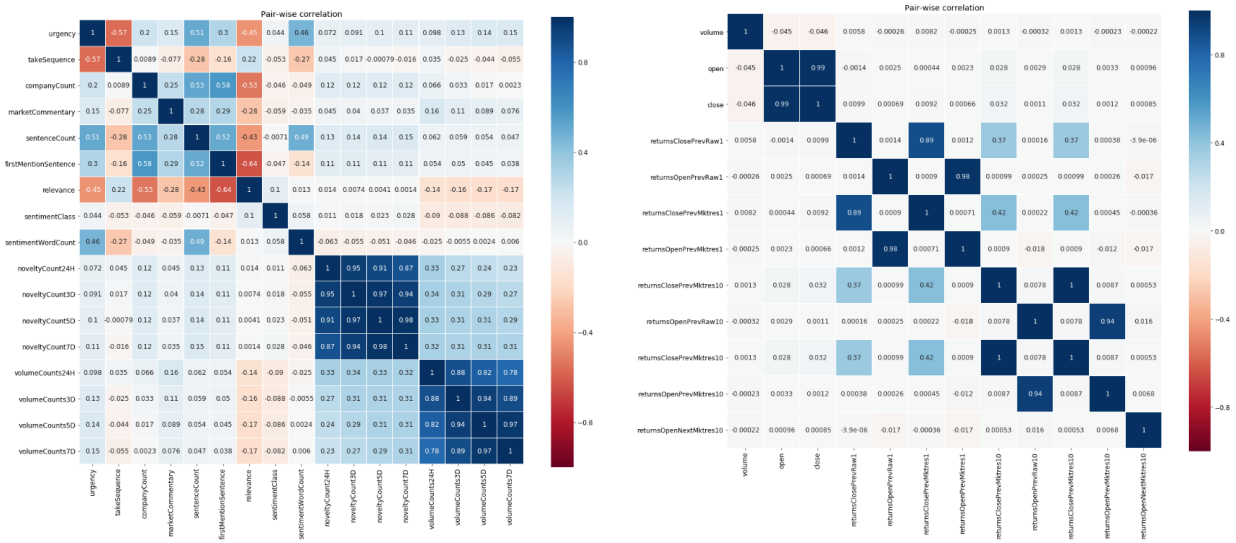
The model score is calculated as the mean divided by the standard deviation of the daily x_t values:

$$\text{score} = \frac{\bar{x}_t}{\sigma(x_t)}.$$

If the standard deviation of predictions is 0, the score is defined as 0.

Project Design

For this project, first, some effort will be put forth into feature selection to mitigate correlating features. Some features are highly correlated to others in the dataset (seen below).



Next, I will layout functions for calculating model performace. This allows for reliable and repeatable evaluation of the benchmark model and target models.

Then I'd move on to defining the naïve benchmark model in code to produce the initial results for the project. The performance of this model will be used to compare to the performance of the target model. The goal is for the target model to outperform the benchmark model.

Finally, I'd move on to implementing a deep neural network to take in stock trading data as well as data points from news sources such as sentiment, urgency, etc. Since time is a large component of the problem, the architecture I would likely use would be that of a recurrent neural network (RNN). Specifically, I will be implementing a Long-Short Term Memory (LSTM) network for the benefits it posses in carrying forward relevant information from one time-step to future time-steps, as well as 'forgetting' irrelevant information that isn't predictive in the time-series.