Machine Learning Engineer Nanodegree

Capstone Proposal

Kevin Mann
June 22, 2017

Proposal

Domain Background

Technical analysis of asset values is predicated upon the recognition of patterns of quantitative asset measurements, often using price and volume time series of stocks, commodities, and foreign exchange. Patterns in quantitative asset behavior hypothetically represent relatively static phenomena that reveal the underlying valuation processes of investment pressures, and thereby serve as predictive indicators of asset behavior.

The history of technical analysis of asset behavior goes back more than a century in simple terms, while coming into widespread use amongst professional traders during the first half of the 20th century. Different forms and approaches exist, but all rely on dependable factors that contribute to recognizable and repeating patterns.

Problem Statement

The problem to be solved is predicting future stock performance to enhance investment returns throughout potential investment time horizons. In order to make predictions about future returns, the recognition of patterns portends to offer a workable solution. The field of technical analysis allows for a purely quantitative approach to predicting future returns, since the inputs into the analysis are the historical price and volume characteristics of the asset to be predicted. The actual returns compared to the predicted returns will provide a measure of the quality of the analysis, and the problem is applicable at every moment while the asset under analysis is offered on the market.

Datasets and Inputs

The data needed for the analysis are the price and volume history of the assets under analysis, because technical analysis is oftentimes solely based on these quantitative indicators. The data can be obtained from various services catering to investors and researchers, and is originally obtained from public market exchange data providers.

The data source that I have chosen to use is Yahoo Finance. Python libraries, including pandas, has functionality that will connect to the Yahoo Finance data source and retrieve results. The data retrieved will vary in number of samples because each stock has been trading for a different amount of time. The time series period can be daily, weekly, or monthly, and I anticipate possibly needing to test the model on each to determine accuracy, but it could be ideal if the model would select a time period based on the prediction's time horizon.

The model will focus on stocks, but not any single stock. Therefore, the model should be able to take in any time series of stock data and return a prediction.

Solution Statement

The greater solution to this problem is a prediction of future returns of the asset under analysis. Ideally, the prediction will be continuous on both the time and returns dimension, and provide a probability distribution showing the anticipated path of returns. This output of the analysis can be measured against actual asset performance to confirm or refute the validity of the analysis. A lesser solution would offer a single number representing the expected return over a specified period of time, and the least solution would be a binary indicator of direction of returns over a specified period of time. It is difficult to know at the outset what quality of solution is feasible beyond a binary predictor.

The algorithms that I think will be most useful will be decision trees, SVMs, and convolutional neural networks. The final solution will pick the best based on result. I anticipate having to iterate through the available algorithms to determine which is the most likely to produce the best result. After that, model tuning will be needed to enhance the results and maximize the accuracy.

Benchmark Model

There is no standard benchmark to compare a stock price predictor to, where actual returns are compared against predicted returns. However, the conventional expected return of stocks over the long-term is 7%, so this number will serve as a benchmark against which the predictor can be compared. This benchmark number should serve well for all but the binary predictor. For the binary predictor, I have chosen a benchmark of 60% accuracy in predicting direction over a given period of time. Picking at random would be expected to result in a correct prediction 50% of the time. I believe a rate of 60% would be both reasonable and profitable.

Evaluation Metrics

An evaluation metric to compare predicted returns against actual returns is the difference between the two, e.g. a predicted return of 9% and an actual return of 7% would give a difference of 2%. Using this metric, one would expect a distribution of deviances that is centered on zero for a working model. When a binary indicator is being used, predicting a stock's movement up or down, the percentage of correct predictions would serve to evaluate a model's performance.

Project Design

The workflow to determine a model that will serve as a predictor of a single stock's future performance would be to try various machine learning algorithms to see if one works well without data manipulation. After (and perhaps during) algorithm selection, I would analyze the data to determine if there might be derived fields that can be created from the existing data that can provide additional help to the algorithms in determining the desired output, e.g. a daily return field could be derived from the adjusted close field and may serve to clarify the algorithm's purpose. No initial analysis of the data should be required because it is a factual reckoning of the stock's behavior, but there are several time-series calculations which might be helpful to yield better predictions. Tuning the algorithm parameters, where available, would be an additional process to increase the accuracy of the model.