

Capstone Proposal: Investment and Trading

Domain Background

This project is from the investment and trading field. Since securities were traded, people have been attempting to predict prices. There are reams of academic papers using statistics and other techniques to predict stock prices. A few of them, specifically pertaining to machine learning, are listed below:

- https://papers.ssrn.com/sol3/papers.cfm?abstract_id=893141
- https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2416494
- https://papers.ssrn.com/sol3/papers.cfm?abstract_id=893141

I'm particularly interested in this area because it overlaps with my professional career in financial services and as a Quantitative Researcher and Developer.

Problem Statement

We want to predict securities prices in the future using historical data. However, we will go beyond just prices and try to also predict volume and volatility. The reason is that prices are notoriously difficult to predict because most simple predictions have been arbitrated away. Additionally, the ability to predict volume could give a sense of how much market impact there is value in also understanding likely volume and volatility numbers.

To summarize, we want to predict the following for a particular security:

- Price: on a one week and one month time horizon.
- Volume: on a one week and one month time horizon.
- Volatility: on a one week and one month time horizon.

Datasets and Inputs

We will use public data from either Quandl or Yahoo. We will need to get market cap numbers as well as open, high, low, close and volume because market cap will give us information about turnover (volume / market cap).

- Quandl: Available publicly and for free. Has an API that can be accessed through Python.
- Yahoo: Also available publicly and for free. There are wrappers that can access the API using Python.

Solution Statement

The solution will be a web application that takes in a set of ticker symbols as well as metrics (price / volume / volatility) to make predictions on. The output will be the predictions after running the models.

Benchmark Model

Because of the theory of efficient markets and the difficulty of predicting prices, the benchmark model will simply be whether our model can do better than a naive model that extrapolates the most recent price / volume / volatility measures.

Evaluation Metrics

We will use ROC scores as well as a profitability metric (one where we take hypothetical trades based on the predictions of the model).

Project Design

- 1) Gather data. The first step will involve connecting to the services to gather some experimental data. During this process, it will become more evident what data is available and for what periods. We would want data for many tickers and for at least five years of history.
- 2) Build specs for user interface. Next, we'll want to figure out what the app should look like. We will take the problem defined earlier in this proposal and map out a user interface that can answer these questions.
- 3) Experiment with core. The core analysis framework will involve a lot of trial and error. We will try several machine learning algorithms -- linear regression, k-nn, and some sort of boosting -- to obtain results. We will process and validate the data for inclusion into the model. We will set aside the most recent year as our out of sample set. For the rest of the data, we will use it to train and test our algorithms and tune parameters.
- 4) Connect the back and front. Once the core engine is built, we'll need to connect the core to the front end