Predicting Stock Liquidation Days Using RNNs

Team Members: Andrew Schreiber (ajs2409), Shahana Nandy (sn3031)

Goal

In the margin lending business, brokers lend money for clients to trade stocks with. This allows clients to trade with more money than they have, potentially increasing their returns. Just like where in a mortgage, the bank uses the asset that you bought with loan (the house) as collateral for the loan and can sell it if you do not meet the terms of the loan, brokers use the stock portfolio that the client buys with loan as collateral and will liquidate the portfolio. However, the value of stock portfolio fluctuates much more than the value of home, so unlike where for a mortgage there is a fixed down payment which serves to mitigate the banks risk due to home value loss, in margin lending there is a variable down payment, called margin, which is updated every day based on the current and future projected value of the portfolio. Margin is recomputed daily, and if the client fails to meet an increase in the required margin the broker may choose to liquidate the portfolio in order to attempt to make their loan payment back.

One important challenge when deciding whether to liquidate a portfolio or not is how long do you expect it to take you to liquidate it. Institutional clients often times have large positions in a particular stock which might be a sizable percentage of, if not greater than, the total number of shares that might trade for that stock on any given day. If the broker were to try to sell this all at once, it would significantly bring down the price of the stock and hence cause losses to the broker. This means it can regularly take multiple days (often times weeks), to fully liquidate a portfolio in a way that will not significantly bring down the value of the portfolio. Its important to note that there is also a time crunch here in that the longer it takes to liquidate the portfolio, the more the underlying portfolios value could change, bringing extra risk to the broker. So the challenge of liquidation is to liquidate as quickly as possible, without bringing down the value of the portfolio to significantly.

This brings in the fundamental problem we are trying to model. Given a position on a stock (a stock, quantity pair), and the current date, how long do expect it to take to liquidate that position without significantly moving the market. In order to do this we will make the assumption that there is a certain percentage of the volume of trading on a given day that we can use that will not impact the price of the security greatly. That being said **what we want to model is given a position on a stock, predict the 25th and 75th percentiles of the volume of the stock per day over the next 3 week window**. We pick 3 weeks, as we assume if it takes longer than 3 weeks to liquidate a position we not extend margin on that position to the client (i.e. its too risky and not worth our business) and we pick 25th percentile as we want to be conservative in our measurements. Given this information we could then use in another model that would help us figure out what to set the margin levels at for a portfolio.

Challenges

The first major challenge is the raw dataset we want (stock prices, volumes and metadata over the last 5 years) does not exist anywhere for free in its raw form. However there appears to be ways you can construct this dataset using various free APIs like Finnhub and yfinance. Particular to the stock market, many stocks have over a 30 year history, it will be challenging to decide how much of that to include and is relevant, it also add scale to the data.
Another challenge is to decide which metadata is helpful in the model, for example there are data points like Sector, is a tech stock, finance, etc…, total number of shares, the market capitalization of the stock, and many more that potentially could be useful in the model but it will take effort to determine how much value they add. Also each additional reference data point might need additional effort to find that data, given there is not one whole dataset we're using. This means that mostly it will be up to us ahead of time to think through which reference datapoints will be most useful and make sure we fetch those and add them to the dataset.
Finally its worth nothing that the stock market is in general unreliable as a whole and can be very challenging to make accurate predictions, so it will be interesting to see how well we can do.

Approach/Techniques

1. Experiment with different kinds of RNNs and hyperparameters, implemented in PyTorch, to see which is most effective for the problem
2. Use quantile regression (1 & 2) to create a prediction interval for the volume. Alternatively, Kernel Density Estimation can also be utilised to approximate the prediction interval for the volume.
3. Explore techniques for discovering feature importance to help with identifying new datapoints that would be valuable to add to the model
4. Use Optimisation techniques such as Quantisation and Pruning to improve the performance metrics provided by the PyTorch Profiler.

Implementation Details

For the dataset, the plan is to construct a dataset of relevant details using the free data at the Finnhub and yfinance APIs. This should be able to give historical data over the past 5 years for all symbols on the Nasdaq and NYSE, as well as reference data, like sector, per ticker. Roughly there about 8000 stocks and 1250 business days in the past 5 years, approximately 10 million datapoints worth of history. In terms of hardware we plan to train my model on the Habanero cluster using GPUs for training efficiency.

On obtaining the predicted total volume from our RNN, Quantile Regression or Kernel Density Estimation can be used to estimate the prediction interval of the total volume traded. Quantile Regression would fit a separate regression model for each quantile of interest (in this case, the 25th and 75th percentile) using a suitable loss function such as the pinball loss. Once the models are trained, using the error as the response variable and any relevant predictor variables (e.g., day of the week, month of the year), you can use them to predict the 25th and 75th percentile values for the predicted values from the RNN.

Alternatively, KDE uses the predicted total volume values to compute the kernel density estimate of the probability density function. The kernel function we choose will determine the shape of the density estimate. A possible choice for the kernel function is the Gaussian kernel,
where we determine a bandwidth parameter. The bandwidth parameter controls the smoothness of the estimated density function. Once we have the estimated density function, we can estimate the 25th and 75th percentile values by computing the inverse cumulative distribution function (CDF) of the estimated density function. We can do this using numerical methods such as Newton's method or bisection method. Once we have estimated the inverse CDF values for the 25th and 75th percentiles, we can use them to estimate the corresponding total volume values.

Finally, we optimise for performance using pruning of the RNN branches, or by quantisation to better the performance metrics we obtain from the PyTorch Profiler.

Presentation

The main goal of the project is to hopefully show and that this model can outperform the industry standard which is to just assume that the average over the last 21 days will be the average over the next 21 days. We also hope to show which type(s) of RNN and hyperparameters worked best for this performance, and present on feature importance of features in the dataset. The feature importance could help us research other datapoints that might be helpful in enriching the model.

References

1. https://medium.com/the-artificial-impostor/quantile-regression-part-1-e25bdd8d9d43
2. https://medium.com/the-artificial-impostor/quantile-regression-part-2-6fdbc26b2629
3. https://towardsdatascience.com/building-rnn-lstm-and-gru-for-time-series-using-pytorch-a46e5b094e7b
4. https://www.exxactcorp.com/blog/Deep-Learning/5-types-of-lstm-recurrent-neural-networks-and-what-to-do-with-them