Proposal: Numerai Trading Predictions

Zhiliang Gong
zhiliang.gong@gmail.com

Domain background

More and more hedge funds are using machine learning to predict the price of financial products, such as stocks, bonds, options, and futures. Top players in this field includes D. E. Shaw, Two Sigma, Renaissance Technologies, etc [1]. These quantitative trading firms are mostly playing a zero-sum game, as gains of one firm often come from losses of other firms. The pressure to stay on top of other firms is a paramount struggle for any quantitative hedge fund.

The difficulty to compete effectively against other players in quantitative trading stems from the them sharing the same set of information, such as company reports, prices over time, trading volumes, etc. This set of information could enable algorithms of one firm to "guess" the "thinking" of algorithms of other firms as the algorithms evolve with fresh inputs from the open market. Here comes Numerai, a startup in quantitative trading, working to crowdsource predictions from high performance individuals [2], to compete against the in-house designed algorithms of the big quantitative funds.

Problem statement

Numerai releases a large pre-processed and encrypted financial dataset each week, which hides the meaning of each feature in the dataset, so that data scientists don't have to guess what the other data scientists might think about a given feature [3, 4]. In a given dataset, there are 50 features, and one binary output target (0 or 1). The distribution of the target is roughly 50% 0's and 50% 1's. In this project, propose to use machine learning models to predict the target with accuracy significantly higher than chance, i.e., 50%. I propose to prototype and train a machine learning model, improve the prototype model over fresh data, and then use the final model to make predictions for a validation set.

Datasets and inputs

The datasets will be one of the weekly datasets from Numerai. Each weekly dataset contains a training set with around 500k entries, a test and validation set around 200k entries, and a live set with around 100k entries without specifying the target. There are 50 input features, all numerical, and normalized between 0 and 1. The target is binary, 0 or 1, and the distribution of 0 and 1 are even. Presumably, the target represents the up or down of the price of a financial asset.

Solution statement

The solution to the problem is a multitude of machine learning methods in supervised and unsupervised learning. I anticipate to use clustering methods for dividing the datasets into feature sets, and try different combinations of regression and classification methods to refine predictions. To cover different biases, I would use a combination of ensemble and non-ensemble methods. The ensemble methods will include random forests, boosting, etc. Non-ensemble methods will include logistic regression, decision trees, support vector machines, naïve bayes, k-nearest neighbors, and neural networks. I'm particularly interested in using multiple layer neural networks with pooling mechanisms to work with the data.

Benchmark model

The benchmark for financial predictors are chance. In the current situation, it's the ability to predict the target at an accuracy higher than 0.5 over extended period of time. The higher the better. Keep in mind that even a slight advantage over chance can bring enormous amounts of profits.

Evaluation metrics

The metric to evaluate the model is the accuracy score, i.e., the percentage of the correct predictions for data from the validation set.

Project design

The project will contain three parts: training a model, testing the model, and validating the model.

In the training part, I'll use a combination of supervised and unsupervised learning to test different combinations of machine learning methods. Specifically, I would test grouping the data with unsupervised learning methods, such as k-nearest neighbors, and test different ensemble and non-ensemble supervised learning methods for predicting the target.

In the testing part, I hope to achieve a model with low testing errors. I'm also using the testing stage as the feedback for improving the model. So if the model does not perform well for the testing stage, I would circle back to the training stage, and try improving the model.

In the validation part, I will test the finalized model on this fresh set of data, and present the result as final.

References

1. http://www.streetofwalls.com/finance-training-courses/quantitative-hedge-fund-training/quant-firms/
2. https://techcrunch.com/2016/12/12/numer-ai-is-a-crowdsourced-hedge-fund-for-machine-learning-experts/
3. https://numer.ai/help

4. https://www.kaggle.com/numerai/encrypted-stock-market-data-from-numerai