CSE 6242 Project: Stock Portfolio Recommendation System

Prepared by:

Kimi Gala (*kgala7*)

Darnesh Narla (*dnarla3*)

Somrita Sarkar (*ssarkar61*)

Daniel Mantica (*dmantica3*)

Abhirup Mishra (*amishra90*)

Beenapreet Kaur Singh(*bsingh46*)

1 Introduction - Motivation:

To invest successfully in the world of stocks, it is important to understand and keep track of multiple factors. The complexities of these factors along with the required time commitment often act as deterrents for many potential investors.

2 Problem Definition:

We plan to suggest a portfolio based on user inputs which have an objective of minimizing risk given the approximate tolerance on the return.

3 Survey:

There are 3 significant phases in our proposed method:

3.1 Phase-1 Selection of Stocks:

Score based selection: From user inputs, we can create a subset of stocks which match the criteria using the quality score[1]. Fama, French 5-factor model[10] can then be used to predict stock returns. The performance of the model is not sensitive to the specifics of the way its factors are defined. This model can be made better and robust by adopting the previously mentioned quality metric as opposed to the size factor as suggested by Asness et al[2].

Predicted Returns based selection: Several types of models have been developed to forecast future stock returns. Linear[3] or logistic[19] regression models are common for stock forecasting. Support vector machine techniques[18] can be adopted to classify stocks as low or high return[24] or predict future returns[22]. Neural network models[4][8][9][14] have been developed to try to capture the nonlinear relationship between stock returns and features. As model parameters, a combination of technical and fundamental factors can be used[12][13][20][21][23]. Dimensionality reduction techniques[25] like principal component analysis[24] or techniques like the fuzzy Delphi method[17] can be used to capture the best features for prediction.

3.2 Phase-2 Optimisation:

Portfolio Optimisation: One way to design the optimization problem is to focus on minimizing value at risk first while a second step dynamically maximizes the value of the stock portfolio.[11]

Another algorithm draws upon the genetic theory of evolution where we start with a population of stocks, select "best" stocks based on certain selection method, generate random weights for these stocks and create a new population. The process is repeated until we have the optimized the weights to come up with the best subset of stocks.[16]

3.3 Phase 3: Web-based UI

There are existing pieces of works that focus on providing visualization tools that help customers compare different portfolios.[15]

4 Methodology

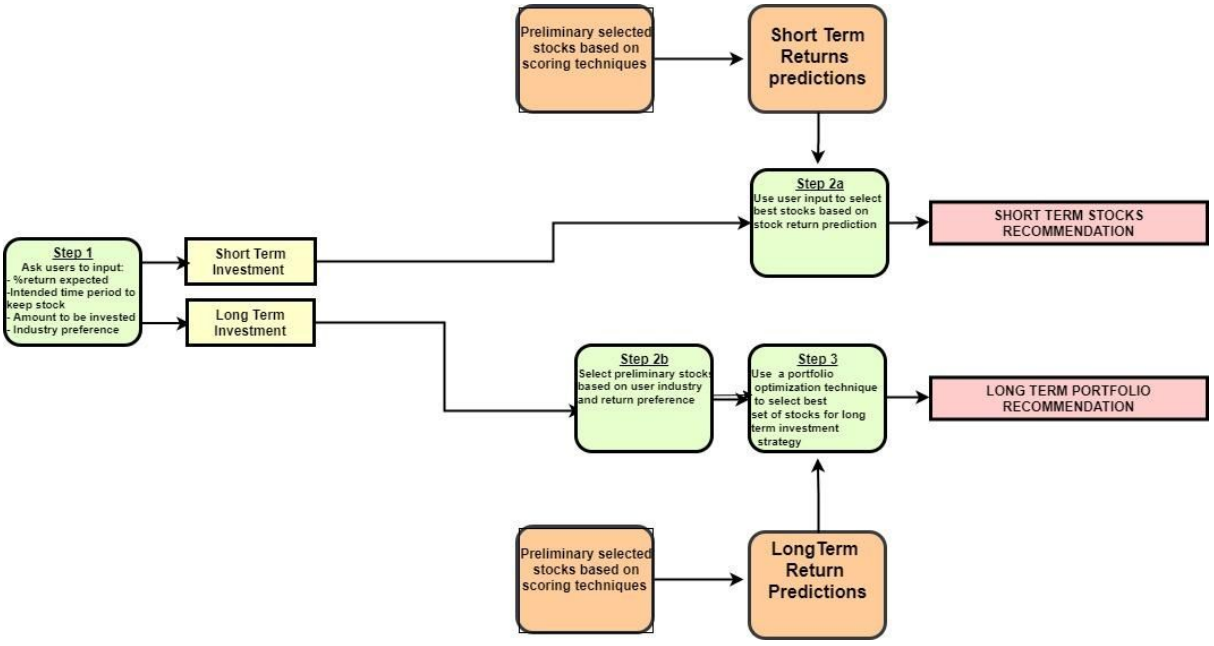


Fig 1: Flow of methodology adopted

4.1 Approach

4.1.1 Phase 1: Selection of Stocks

Quality Score:

We developed a quality score, that all else equal, an investor should be willing to pay a higher price for: stocks that are profitable, growing, safe and well managed. High quality stocks generally have a higher risk adjusted return. Fundamental data obtained from COMPUSTAT database is used to compute the score. It gives equal weightage to each of the factors namely "profitability", "growth", "safety" and "payout".

To put each measure on equal footing and combine them, each month each variable was converted into ranks and standardized to obtain a z-score. More formally, let “x” be the variable of interest and “r” be the vector of ranks,

$$r_i = rank(x_i)$$

Then the z-score of x is given by $z(x) = (r - \mu)/\sigma$, where μ and σ are the cross sectional mean and standard deviation of r.

Profitability: It is profits per unit of book value. It can be measured in several way such as in terms of gross profits, margins, earnings and accruals.

$$\text{Profitability} = z\,(z_{\Delta gpoa} + z_{\Delta roa} + z_{\Delta cfoa} + z_{\Delta gmar} + z_{\Delta acc})$$

gpoa = growth profit over assets

roe = return on equity

roa = return on assets

cfoa = cash flow over assets

gmar = growth margin

acc = accruals

Growth: It is measured as prior five year growth in each of our profitability measures.

$$\text{Growth} = z\,(z_{\Delta gpoa} + z_{\Delta roa} + z_{\Delta cfoa} + z_{\Delta gmar} + z_{\Delta acc})$$

Safety: We consider both return based measure of safety (market beta and volatility) and fundamental based measures such as low leverage, low volatility of profitability and low credit risk. Safe securities are defined as companies with low beta (BAB), low idiosyncratic volatility (IVOL), low leverage (LEV), low bankruptcy risk (o-score and z-Score) and low ROE volatility (EVOL)

$$\text{Safety} = z\,(z_{bab} + z_{ivol} + z_{lev} + z_o + z_z + z_{evol})$$

Payout: It is a fraction of profits paid out to shareholders. It can be seen as a measure of shareholder friendliness.

Payout = z (z_{eiss} + z_{diss} +z_{npop})

eiss = Net equity Issuance

diss = One year percent change in total debt

npop = Net payout over the past 5 years/Profits over the past 5 years

Once we get different quality factors to measure the quality of a stock, we give equal weightage to each of these factors and again to put each measure on equal footing, we take the z scores of the sum of those factors

Quality = z (Profitability + Growth +Safety + Payout)

Return Prediction:

Short-Term Return Prediction: The recurrent neural network (RNN) is a type of artificial neural network with self-loop in its hidden layer(s), which enables RNN to use the previous state of the hidden neuron(s) to learn the current state given the new input. RNN is good at processing sequential data. Long short-term memory (LSTM) cell is a specially designed working unit that helps RNN better memorize the long-term.

$$\mathbf{s_t} = \sigma(\mathbf{W}\mathbf{x_t} + \mathbf{U}\mathbf{s_{t-1}} + \mathbf{c})$$

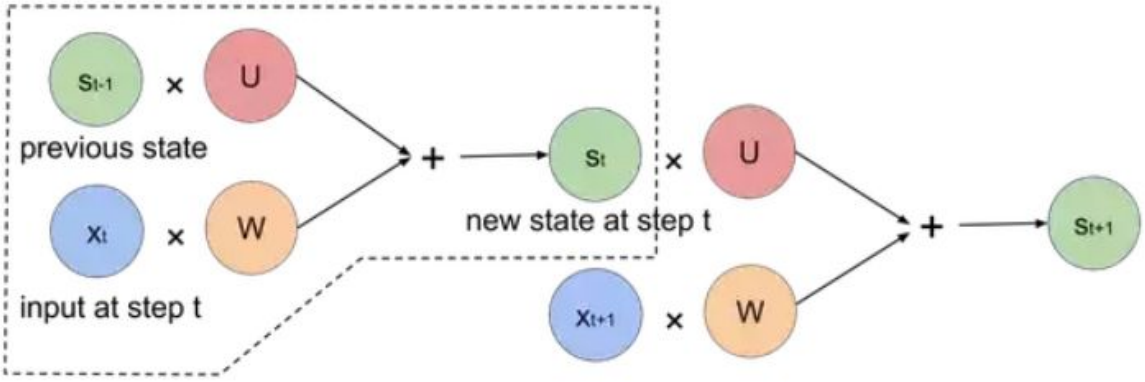


Fig 2: Shows how the final state(s(t+1) is the summary of the whole sequence given by x(t+1) and s(t) , ***s(t)=sigma(W*x(t)+U*s(t-1) +c***

Data Preparation: The stock prices is a time series of length N, defined as $p_{o,}\ p_{1,}p_{2\ ....}\ p_{n-1}$ where p_i is the average of the Open, High, Low and Close price of a stock for 1 day. We use a sliding window approach of size 5. We consider time series data with features t,t-1,t-2, t-3 and t-4 to predict OHLC average prices for time 't'.

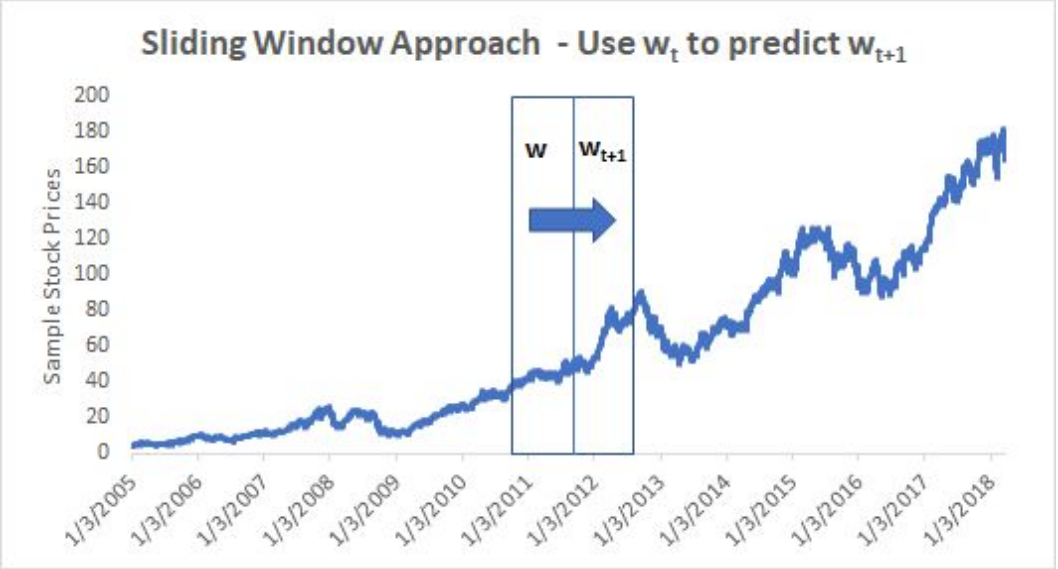


Fig 3 : Moving window approach for RNN-LSTM time series (left). Recursive Approach to predicting prices of a stock (right)

To solve the out-of scale issue the OHLC prices are first normalized, therefore the task now is to predict the relative change rates instead of the absolute prices.
The sequence of prices are first split into non - overlapped small windows where:

***number of samples**= number of training data (daily prices of upto past 15 years)*
***step_size (also called look back argument)** = 5 (t, t-1, t-2, t-3 and t-4)*
***number of features** = 1 (Since we're using only time series data)*

A recursive approach was used to predict the next 5 days prices for a particular stock.

Long-Term Return Prediction: For long term stock return prediction we implemented both regression and time-series models.
Regression Models: We first built a couple of linear regression models: one with 6-months out price of stocks as dependent variable and another with 12-months out price of stocks as independent variable. The independent variables considered were a combination of fundamental variables (for the present financial quarter), variables used for quality scoring

and price: Present Price, Earnings per share (EPS), Market Value, Returns on assets, Returns of equity, Gross Margin, Income, Long Term Debt, Gross Profit Over Assets, Cash FLow Over Assets, Accruals, Leverage, Idiosyncratic volatility, and Beta of a stock.

LASSO was used for variable selection. A cross-validation was performed on scaled data to obtain a value of lambda that minimized error metric (MSE used in this case). Based on the optimum value of lambda, LASSO chose 6 key variables: Present Price, EPS, Gross Margin, Returns on Assets, Five year growth in Returns on Assets, Accruals.

These variables were then used in a linear regression. Finally, based on significance and sense-making, 5 key variables were chosen to predict expected price 6 & 12 months (two separate models): Present Price, EPS, Gross Margin, Five year growth in Returns on Assets, Accruals

The model was built with data from 2013 onwards. It was later scored on 2012 data and used for backtesting.

Time-Series Models: Apart from a regressing price, we deemed it important to take the temporal component of the data into account for long-term prediction as well. With this in mind, we also implemented two time series models - Vector Autoregression (VAR) and ARIMAX.

For VAR, our dependent variable was 6-month and 12-month price (for each quarter of the training period). Selected independent variables were long term debt, market value and income generated by the corresponding firm as they granger cause stock price. We considered unrestricted VAR for our analysis.

For simplicity, let's consider bivariate $VAR(1)$ model,

$$y_{1t} = \pi_{10} + \pi_{11}\, y_{(1t-1)} + \pi_{12}\, y_{(2t-1)} + \epsilon_{1t}$$
$$y_{2t} = \pi_{10} + \pi_{11}\, y_{(1t-1)} + \pi_{12}\, y_{(2t-1)} + \epsilon_{2t}$$

where,

y_{1t} represents historical quarterly price

y_{2t}represents long term debt

$\epsilon_{1t}, \epsilon_{2t}$ is white noise

$cov(\epsilon_{1t}, \epsilon_{2t}) = \sigma_{12}$ for t=s and 0 otherwise

VAR assumes that not only do all predictors affect the dependent variable price but are also affected by price. Additionally, VAR does not include moving-average (MA) terms and approximates any existing MA patterns by extra autoregressives lags. Hence, we also implemented ARIMAX which is a less parsimonious solution than directly including MA terms as in an ARIMAX model. Also, ARIMAX is based on the assumption that predictors only impact the time series in question i.e stock prices and are not impacted by them.

For ARIMAX, our dependent variable was same as that of VAR. Independent variable was historical quarterly stock price of the firm. In addition, we used three exogeneous variables - *they help in predicting the dependent variable but are not affected by them* - long term debt, market value and income generated by the corresponding firm.

For simplicity, let's consider an ARIMAX (1,0,1),

$$y_t = \beta_1 x_{LTDebt} + \beta_2 x_{MktValue} + \beta_3 x_{Income} + \phi y_{t-1} + \Theta \epsilon_{t-4}$$

where,

y represents price
$\beta_1, \beta_2, \beta_3, \phi$ and Θ are estimated coefficients

We evaluated both the models on mean absolute percentage error against the test data. It was observed that for 95% of the stocks ARIMAX performed better than VAR, hence we selected ARIMAX along with the linear regression model described above for phase 2.

4.1.2 Phase 2: Optimization

We used an optimization procedure to determine the percentages of investment capital (portfolio weights) that should be invested in each stock selected in through the quality scoring. This procedure, called Mean-Variance Optimization, is done by finding the vector of weights that minimizes the objective function (total portfolio variance), subject to constraints that the weights must sum to one and that the expected portfolio return must equal a given target return. The expected portfolio return is taken from the predictions made by the short-term and long-term prediction models. To ensure that the user will not be overwhelmed by having too many stocks to invest in, we added an additional constraint limiting the number of the stocks in the portfolio to ten or less.

4.1.3 Phase 3: Web-based UI

We created an interactive, web-based UI that allows users to enter their portfolio preferences and then see what our system recommends for their optimal portfolio. On the first page, the user enters the amount they want to invest, their desired return on investment, and the time horizon over which they hope to generate that return. They are then taken to a page displaying the stocks and allocations they should invest in to create an optimal portfolio, as well as a visualization for what our models predict for this portfolio's

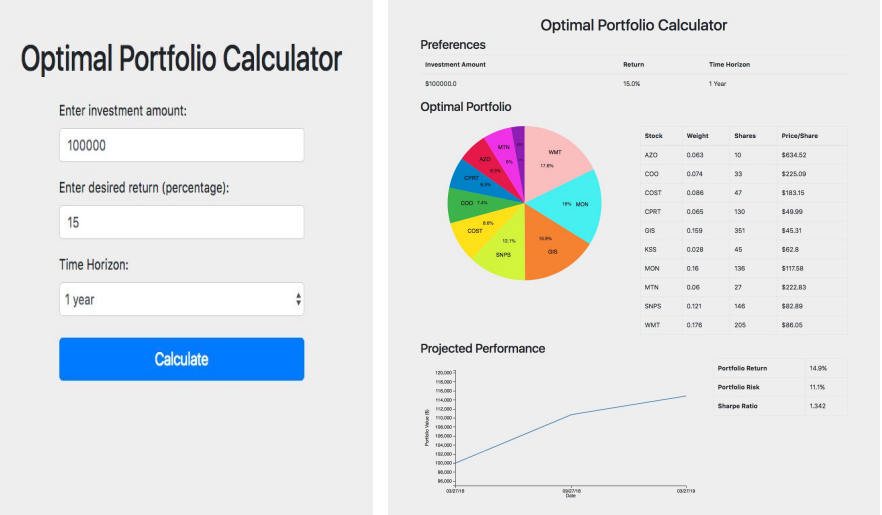


Fig 4: Screenshots of UI

performance over time.

4.2 Innovation

4.2.1 Combining prevalent rule-based algorithms with newly researched machine learning algorithms

There is a scarcity of ensemble approaches that take the best of rule-based and statistical algorithms. We plan to calculate quality scores to select a subset of stocks while use ML models to select another subset and combine both to get the final set of selections

4.2.2 Predicting long term 1 year returns for interested users

While it is indeed difficult to predict years far out into the future given market volatility we are leveraging on certain engineered features and fundamentals of stocks to predict 1-year returns

4.2.3 Building different models for short term and long-term predictions

To account for different factors that are at play in daily price fluctuations and long-term price fluctuations, we are creating two different models. For short term prediction our focus is on past price trend (open, high, low, close) only while for long term we are looking at fundamentals of the stocks as well.

5 Experiments/Evaluation

We plan to evaluate the accuracy of our return predictions based on back-testing and cross-validation. We also plan to conduct a user survey to gauge the usability and aesthetics of the GUI.

5.1 Findings: Evaluation

5.1.1 Short-Term RNN Model:

Optimizer	Epochs	MSE(Train)	MSE (Test)
Adagrad	20	0.5	0.555
Adagrad	**50**	**0.01**	**0.02**
Adagrad	100	0.0092	0.06
SGD	50	0.4	0.8
Adam	50	1.5	1.7

Fig 5: Shows the Hyperparameter tuning Results for the RNN-LSTM model with 2 hidden layers

We have stuck to 2 Hidden Layers as the accuracy of the test set - starts to decrease as the model fits more of the noise in the training set. We've used Adagrad Optimizer, with 2 hidden LSTM layers and 1 dense output layer, and 50 epochs to train our stock data as it performs best on the **TEST DATA** as shown above

Average Train Score: 0.01 MSE
Average Test Score: 0.02 MSE

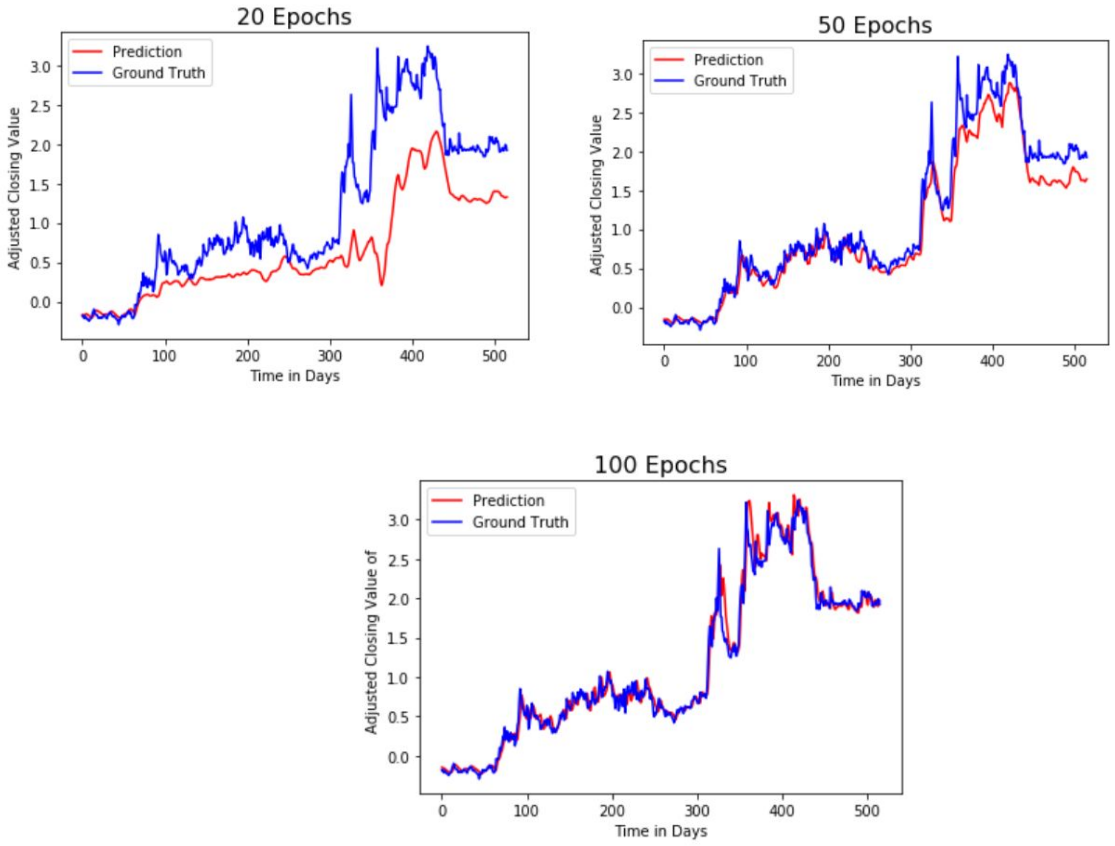


Fig 6: Shows the results of the Training Data of the Adagrad Optimizer with increase in number of epochs

5.1.2 Long-Term Model:

Regression Models:

R-squared - 6 months model: 88.7%

R-squared - 12 months model: 93.5%

Adj. R-squared - 6 months model: 88.1%

Adj. R-squared - 12 months model: 93.1%

MAPE - 6 months model (tested on 2012 data): 20.8%

MAPE - 12 months model (tested on 2012 data): 12.5%

Time Series Models:

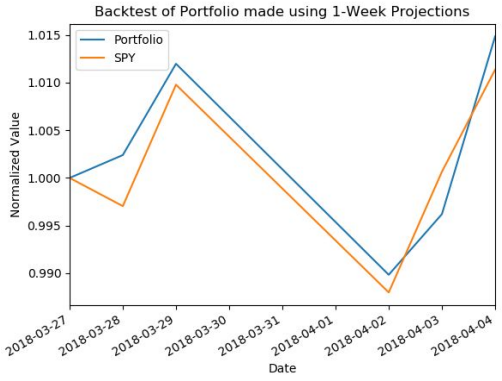
Average MAPE - 6 months model (tested on 2017 data): 4.2%

Average MAPE - 12 months model (tested on 2017 data): 6.5%

5.2 Findings: Experiment

For each model projection (One week, six months, and one year) we created a portfolio using our full recommendation system, then tested how that portfolio would have performed in the market. For the one week model we tested for March 27, 2018 and for the six month and one year models we tested for January 3, 2017; in making the predictions and performing the optimization we only used data that was available before these dates. The results for the performance of these portfolios, compared to the performance of the SPY index, are shown below:

1-week

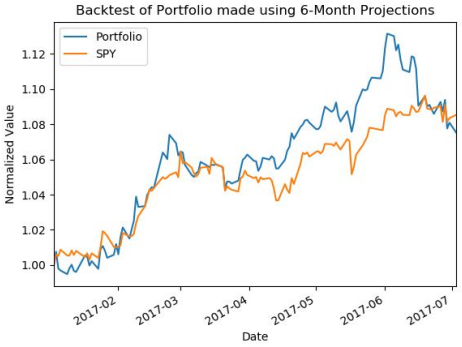



```
Backtest of Portfolio made using 1-Week Projections

Portfolio Stats:
Target Return: 0.005
Cumulative Return: 0.0148437891602
Average Daily Return: 0.00304385382363
Risk (Standard Deviation): 0.0371643110908
Sharpe Ratio: 0.491415619065

Benchmark (SPY) Stats:
Cumulative Return: 0.0113583727239
Average Daily Return: 0.0023508595586
Risk (Standard Deviation): 0.036517273259
Sharpe Ratio: 0.386259873554
```

6-months

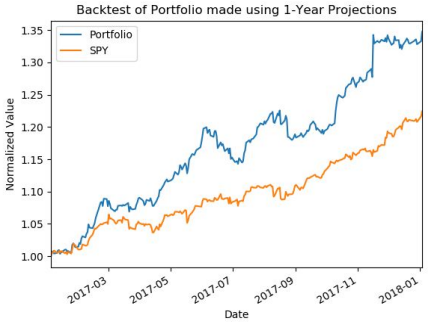



```
Backtest of Portfolio made using 6-Month Projections

Portfolio Stats:
Target Return: 0.2
Cumulative Return: 0.075322722212
Average Daily Return: 0.000591728234089
Risk (Standard Deviation): 0.0624099499559
Sharpe Ratio: 1.20412667824

Benchmark (SPY) Stats:
Cumulative Return: 0.0852869031688
Average Daily Return: 0.000664778258094
Risk (Standard Deviation): 0.0499087359608
Sharpe Ratio: 1.67830458751
```

1-year




```
Backtest of Portfolio made using 1-Year Projections

Portfolio Stats:
Target Return: 0.4
Cumulative Return: 0.347395499228
Average Daily Return: 0.00120406834258
Risk (Standard Deviation): 0.101518283111
Sharpe Ratio: 3.00073328013

Benchmark (SPY) Stats:
Cumulative Return: 0.224153245757
Average Daily Return: 0.000811902467293
Risk (Standard Deviation): 0.067616246045
Sharpe Ratio: 3.03789897014
```

6 Conclusions

We have tried to introduce a quantitative framework for selecting stocks as well as built a system that suggests the optimal quantity that one needs to buy of those “high quality” stocks in our recommendation engine. Our process has outperformed the S&P 500 index in backtesting for both long-term and short-term horizons. Thus, through our project, by using quantitative techniques, we have a created a portfolio recommendation engine for an average investor.

7 Heilmeier Questions Answered

1. Suggest optimum portfolio of stocks for investment to the user based on his preferences.
2. The prevalent approaches for predicting stock prices/returns are either rule-based or machine learning algorithm based. But there is a scarcity in the market to try and combine the best of both approaches. Also, most approaches focus on predicting daily prices/returns. From investors’ perspective it is also important to have long-term predictions.
3. Our approach will combine rule-based algorithms and machine learning based algorithms to come up with a robust selection of stocks that provide the best returns. On top of it, we would have an optimization algorithm to create an optimal portfolio for a user which would take into account the users risk appetite (proxied by the user provided input of expected returns), long/short options, preference for sectors, and investment budget. Additionally, we are building different models for short term prediction (5 days) and long-term predictions (1 year).

4. Any investor who wants to earn more return for his investment than the average return that can be availed from the stock market
5. We can measure the impact by conducting user surveys regarding how accurate and relevant the predictions were for them, once their duration of investment is completed

6. Risks: Providing inaccurate predictions due to unexpected volatility in a particular stock or the entire stock market

` Payoff: It will reduce the time investors spend researching for their investment decisions and will help them gain better returns than the average stock market return.

7. All technologies that will be used are free and hence there will be no cost

8. We expect to complete the project in 7-weeks.

9. There are 2 measures of success:

    a. Stock selection model based on quality score and predicted returns. This will be the base of our portfolio recommendation project

    b. Developing a portfolio optimization model that effectively incorporates the users stock return criteria, sector preference and time horizon

The midterm check can be conducted in another 1.5 weeks to determine the robustness of our stock return prediction model. The final check can be conducted the week later to confirm that the optimization algorithm along with execution of trade is effectively implemented and GUI is built.

8 Activity Plan

Activity	Status	Members Involved
Literature Review	Completed	All
Proposal, Presentation, Progress Report	Completed	All
Data Collection	Completed	Abhirup, Darnesh
Phase 1.1 Quality Scoring	Completed	Abhirup, Darnesh
Phase 1.2a Short Term Prediction	Completed	Beena
Phase 1.2b Long Term Prediction	Completed	Kimi, Somrita
Phase 3 Optimization	Completed	Daniel
Phase 4 GUI	Completed	Daniel, Beena
Experiment/Evaluation	Completed	Somrita, Kimi
Final Report & Poster	Completed	All

All team members have contributed similar amount of effort

References:

[1] Asness, C. S., Frazzini, A., & Pedersen, L. H. (2014). Quality minus junk.

[2] Asness, C. S., Frazzini, A., Israel, R., Moskowitz, T. J., & Pedersen, L. H. (2017). Size matters, if you control your junk.

[3] Bini, B. S., & Mathew, T. (2016). Clustering and Regression Techniques for Stock Prediction. *Procedia Technology*, *24*, 1248-1255.

[4] Dagli, C. (2011). Stock Market Prediction with Multiple Regression, Fuzzy Type-2 Clustering and Neural Networks

[5] David Easley, Marcos Lopez de Prado, Maureen O'Hara (2012). Discerning information from trade data. Journal of Financial Economics

[6] David Easley, Marcos Lopez de Prado, Maureen O'Hara (2012). Flow Toxicity and Liquidity in High Frequency World. *Review of Financial Studies, Vol. 25, No. 5, pp. 1457-1493, 2012*

[7] David Easley, Marcos Lopez de Prado, Maureen O'Hara (2011). The Microstructure of the "Flash Crash": Flow Toxicity, Liquidity Crashes, and the Probability of Informed Trading

[8] Eakins, S., & Stansell, S. (2003). Can value-based stock selection criteria yield superior risk-adjusted returns: an application of neural networks

[9] Enke, D., & Thawornwong, S. (2005). The use of data mining and neural networks for forecasting stock market returns. *Expert Systems with applications*, *29*(4), 927-940.

[10] Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, 116, 1–22.

[11] Guennoun, Z., Hamza, F., & El hachloufi, M. (2012). Stocks portfolio optimization using classification and genetic algorithms. *Applied Mathematical Sciences*, *6*(94), 4673-4684.

[12] Guerard, J. B., Chettiappan, S., & Xu, G. (2010). Stock-selection modeling and data mining corrections: Long-only versus 130/30 models. In Handbook of Portfolio Construction (pp. 621-648). Springer, Boston, MA.

[13] Hargreaves, C. A., Dixit, P., & Solanki, A. (2013). Stock portfolio selection using data mining approach. *IOSR Journal of Engineering (IOSRJEN)*, *3*(11), 42-48.

[14] Quah, T. S., & Srinivasan, B. (1999). Improving returns on stock investment through neural network selection. *Expert Systems with Applications*, *17*(4), 295-301.

[15] Savikhin, A., Lam, H. C., Fisher, B., & Ebert, D. S. (2011, January). An experimental study of financial portfolio selection with visual analytics for decision support. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on* (pp. 1-10). IEEE.

[16] Sinha, P., Chandwani, A., & Sinha, T. (2015). Algorithm of construction of optimum portfolio of stocks using genetic algorithm. *International Journal of System Assurance Engineering and Management*, *6*(4), 447-465.

[17] Thakur, G. S. M., Bhattacharyya, R., & Sarkar, S. (2016). Stock portfolio selection using Dempster–Shafer evidence theory. *Journal of King Saud University-Computer and Information Sciences*.

[18] Trafalis, T. B., & Ince, H. (2000). Support vector machine for regression and applications to financial forecasting. In *Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on* (Vol. 6, pp. 348-353). IEEE.

[19] Upadhyay, A., Bandyopadhyay, G., & Dutta, A. (2012). Forecasting stock performance in indian market using multinomial logistic regression. *Journal of Business Studies Quarterly*, *3*(3), 16.

[20] Wang, H., Jiang, Y., & Wang, H. (2009). Stock Return Prediction Based on Bagging-Decision Tree

[21] Weng, B. (2017). Application of machine learning techniques for stock market prediction.

[22] Yang, H., Chan, L., & King, I. (2002, August). Support vector machine regression for volatile stock market prediction. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 391-396). Springer, Berlin, Heidelberg.

[23] Yin, J., Si, Y. W., & Gong, Z. (2011, June). Financial time series segmentation based on Turning Points. In *System Science and Engineering (ICSSE), 2011 International Conference on* (pp. 394-399). IEEE.

[24] Yu, H., Chen, R., & Zhang, G. (2014). A SVM stock selection model within PCA. *Procedia computer science*, *31*, 406-412.

[25] Zhong, X., & Enke, D. (2017). Forecasting daily stock market return using dimensionality reduction. *Expert Systems with Applications*, *67*, 126-139.