CIVE 7100 – Time Series and Geospatial Data Sciences

Final Project Report

Title: Stock Portfolio Diversification using Time Series Analysis and Machine Learning.

Kedar Ghule

NUID: 001040571

Department of Electrical and Computer Engineering

College of Engineering

Northeastern University, Boston

Acknowledgement

I would like to take this opportunity to thank Prof. Auroop Ganguly for providing me the opportunity to apply concepts and methods learnt during the coursework in this project. I would also like to thank the teaching assistant Ms. Puja Das for her invaluable inputs and for all her help during the course which made me understand the concepts even better. I am also thankful to Mr John Wesley, Lead Internal Auditor at Nokia, and a former finance student at Northeastern University for guiding me with the financial knowledge aspect of the project. Lastly, I would like to thank Northeastern University for providing the required facilities, Internet access and important access to different literatures.

Kedar Ghule

Contents

Note: The link to the source code of this project is mentioned in the References.

Abstract

With the recent swings in the stock market, especially observed during the Covid-19 pandemic, it is imperative to understand the companies but also the sectors that show uncertainty in the stock market. Stock market is important for companies that want to raise funds for their expansion. This, in turn, helps them be profitable and launch new products, features or pay their debt. From an investors point of view, they need to have a stock portfolio such that they can maximize their profits. This way an investor can expect a good return on his investment. The key problem lies in the fact that most investors fail in the stock market. Most of them do not study the company or the sector before investing their money. An investor should study the company, the stock's behavior over time and carry out an in-depth fundamental analysis like studying the company's balance statements, debt, etc. and/or technical analysis like price movement and volume. One of the ideas of diversifying one's portfolio is to have stocks such that they cancel each other's volatility. Here, volatility can be referred to as uncertainty. This is the focus of the project. First, the project aims to group stocks with similar weekly percentage returns. After this, the project calculates realized volatility for the stock and uses classical volatility models and machine learning models to forecast the realized volatility of the stock.

Big Picture Summary

The abstract covers the 'why?' aspect of the project. This part of the report covers the big picture of the project. Portfolio diversification is done to reduce the risk. Portfolio diversification involves an investor to include different types of securities (stocks, bonds, etc.) from different issuers and sectors. It's the perfect example of the phrase – 'Do not put all your eggs in one basket.' The idea is to invest in multiple areas and sectors such that even if one fails, the overall portfolio remains secure, and this added security can ensure profits. There are many methods that are employed in portfolio diversification namely – determining correlation between the investments, diversifying across asset classes (bonds, stocks, properties, etc.), diversifying within asset an asset class (like investing in multiple industries), diversifying by location, determining risk tolerance and so on.

In this project, methods like clustering are employed to profile stocks with similar patterns. This can help in the investor's portfolio diversification as they can focus on stocks that exhibit certain behavior. Grouping stocks into different groups can provide valuable insights on a stock's overall risk. The whole point of diversifying your portfolio is to reduce risk and volatility however, some investments or stocks may have a greater impact than others. Given the volatile nature of stock market data, it is beneficial for potential investors to know about the volatility of the stocks before investing in them.

However, it is important to note that even high volatility can generate great profit and that investment strategy is not done solely on a stock's volatility estimate but also some fundamental analysis and/or technical analysis as mentioned in the abstract.

Another point to note in the big picture is that an investor should always rebalance their portfolio regularly. Markets are uncertain and no one can predict what war will start tomorrow or which political party will take power in a country. Certain industries are more volatile than the others and certain companies even if they are in the same sector are more volatile than the others in the same sector.

Focused Project Problem

Below is the problem statement this project intends to answer.

How can we diversify an investor's portfolio such that it can be profitable and has stocks with a security worth investing? How do we present information to an investor looking to diversify their stock portfolio, maximizing their profits such that they can study the company and make an informed decision before investing?

One of the answers to this question is forecasting the stock's closing price. However, it is very difficult to predict something that depends on the future which is uncertain and unpredictable. Stock data is also very volatile as you can see below in Figure 1. As you can see, the stock closing price trends look like that of a random walk.

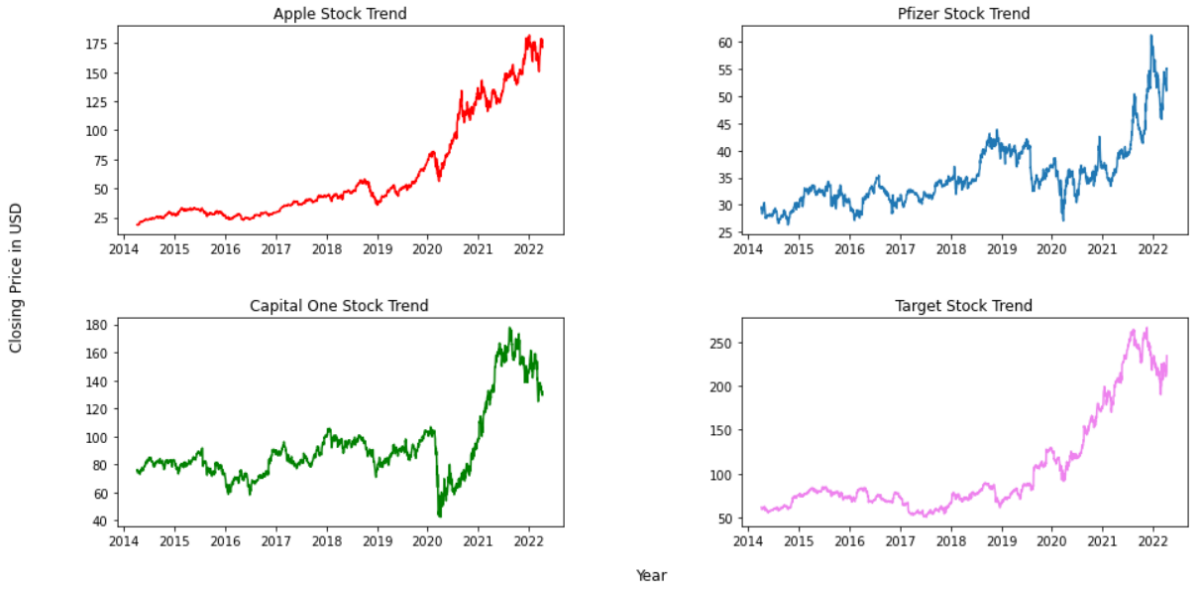

Figure 1. Stock Trends of 4 companies in the S&P 500.

Keeping the above in mind and multiple ways the problem can be approached, this project intends to focus on volatility and how stock's observed historical volatility and prediction can help diversify an investor's portfolio. The level of volatility is an important factor for an investor as higher volatility creates greater risk and uncertainty of the portfolio's performance at any given time. The increased uncertainty also reduces the ability to predict performance and use financial models [1]. The idea of diversifying a portfolio is to have securities in one's portfolio such that they cancel out each other's volatility.

Literature Review

Below is a brief literature review of the references used before the implementation of the project.

1. **An Empirical Evaluation in GARCH Volatility Modeling: Evidence from the Stockholm Stock Exchange**
   Published in 2017, this paper drew my attention to methods like GJR-GARCH and EGARCH for volatility modelling. The paper talks about the disadvantages of the asymmetric GARCH model where an asymmetric effect is usually observed and is registered from a different instability in the case of good and bad news. The paper also talks about how GARCH will always have an asymmetric response in positive and negative periods since all terms in a GARCH model are squared. The paper then focuses on the GJR-GARCH models and EGARCH and how they improve on GARCH. They were observed to better formulate the different responses to different past shocks and to explain conditional volatility [6]. The paper was also crucial in selecting RMSE, MSE and MAE as the preferred metrics for the evaluation of model performance in this project.

2. **Forecasting Financial Returns Volatility: A GARCH-SVR Model**
   This Springer article was referred to while exploring different methods for volatility modelling and inspired from this, the project implements two versions of the SVR-GARCH model – one with the linear kernel and one with the RBF kernel. The idea is to estimate GARCH parameters using SVR. In the paper, the parameter selection of the SVR model is done using a validation procedure based on grid search and sensitivity analysis to select the two free parameters ϵ (loss function parameter) and C (regularization parameter) and one kernel coefficient γ [4].

3. **Prediction and Risk Management: An SVR-GARCH Approach.**
   Like the above literature, this paper explores using the SVR-GARCH approach and compares its performance with the GARCH family models. This paper is even so more relevant to this project as the paper compares the performance of SVR-GARCH with selected 30 stocks listed on the S&P 500 index. The inference from the paper is that the SVR-GARCH model performs way better than the GARCH family models.

4. **Forecasting Volatility of the U.S. Oil Market**
   The paper talks about the HAR-RV model and uses the CBOE Crude Oil Volatility Index (OVX) when forecasting realized volatility in the WTI futures market. During forecasting, the paper concludes that volatility models based on realized volatility can be improved by including implied volatility and other variables and this is done so using the HAR-RV (Heterogenous Autoregressive of Realized Volatility) model [5]. The model considers observable market variables like volume, daily returns, weekly returns, monthly returns and so on.

Methodology

1. **Data Acquisition.**
   The data (symbol, name, sector, and sub-sector) of companies in the S&P 500 index is scraped from the following Wikipedia page mentioned in the references under [2]. The individual stock data for the company and for the S&P 500 index is gathered using the YFinance library in Python. The YFinance library offers to download market data from Yahoo Finance. The below graph shows the trend of the adjusted closing price of the S&P 500 index.

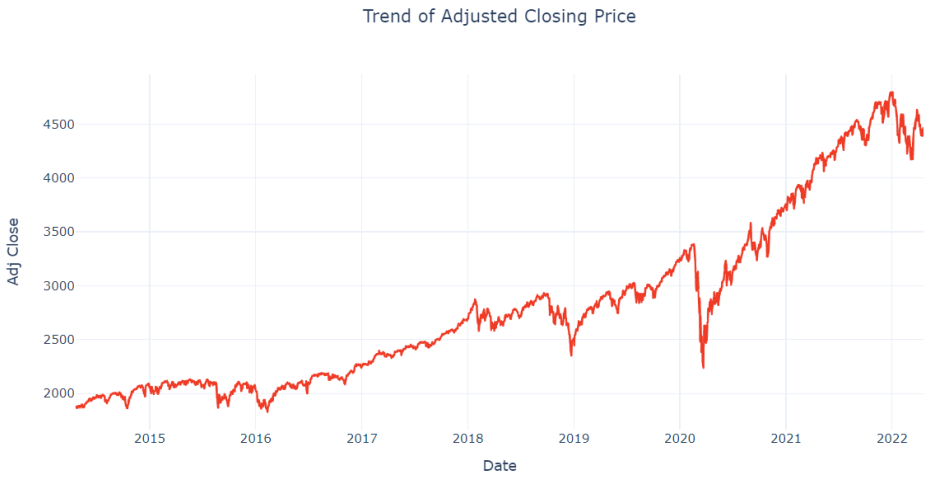

Figure 2. Trend of the Adjusted Closing Price of S&P 500 index.

2. **Preparing and Preprocessing Data.**
   Data wrangling was performed using the pandas library in Python to make the datasets ready to be used in modelling. Other aspects of this step involved data cleaning operations like handling missing values, changing the data type of certain columns, and adding new features like weekly percentage returns.

3. **Modelling and Engineering.**
   This step consists of two parts. The first part focuses on clustering stocks based on their weekly percentage returns. The second part focuses on volatility clustering and prediction.

3.1 K-Means Clustering.

K-Means Clustering is used to profile stocks with similar trends in weekly percentage returns. K-Means Clustering is an unsupervised machine learning technique in which the algorithm clusters the data in an iterative manner. It is used to cluster or profile unlabeled data to predict the class of observations without a target vector. The 'K' value here denotes the number of clusters.

In this project, percentage of weekly returns of each stock was calculated. Next, the elbow method was utilized to find the optimal value of number of clusters – 'K'. The elbow method aims to find that value of 'K' such that there is not substantial decrease in the within cluster sum of squares (WCSS) value. The below figure is the graph generated through the yellowbricks python library which shows the optimal value of 'K' selected.

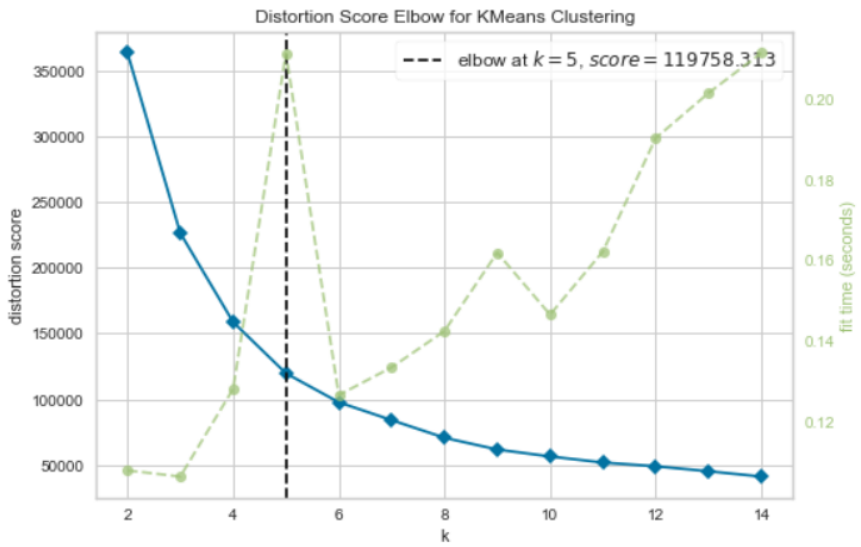

Figure 3. Elbow Method to determine the value of 'K'.

As you can see, the optimal number of clusters is 5.

Finally, we use silhouette analysis to determine the degree of separation between the clusters and this is documented in the results.

3.2 Volatility Clustering and Prediction.

Modelling volatility is essentially modelling uncertainty. When forecasting volatility, we are forecasting realized volatility or return volatility. Realized volatility or return volatility is the square root of sum of squared returns. The formula for realized volatility is given below.

$$\widehat{\sigma} = \sqrt{\frac{1}{n-1}\sum_{n=1}^{N}(r_n - \mu)^2}$$

Where,
r is the returns,
μ is the mean of the returns and,
n is the number of observations.

Below is the graph showing the realized volatility for the S&P 500 index.

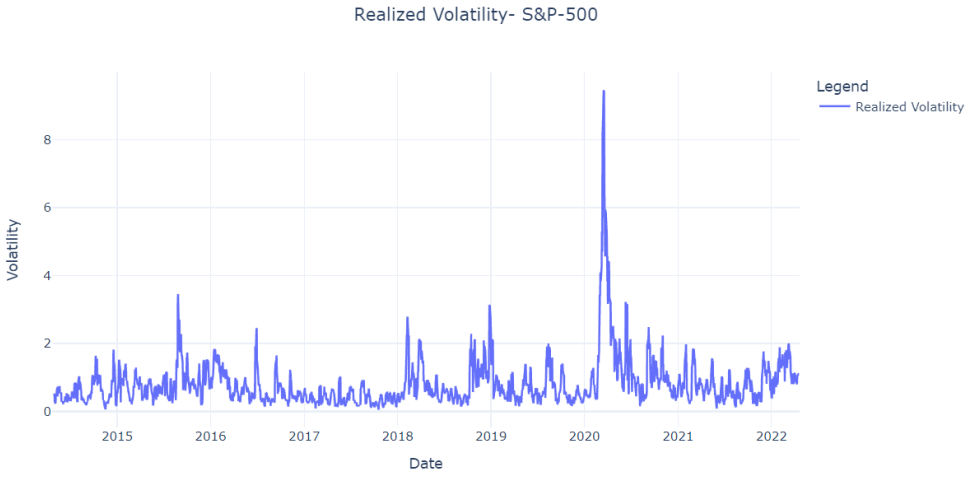

Figure 4. Realized Volatility of S&P 500 index.

Following this, the daily, monthly, and annual volatility of the stock is computed. Next, the volatility clustering is computed and finally we use the realized volatility data as input to the models.

The project explores 8 volatility models out of which 5 are classical volatility models while 2 are machine learning approaches.

This part of the project is prototyped on the S&P 500 index data.

3.2.1 Classical Volatility Models

Below are the classical volatility models that are implemented in this project.

3.2.1.1 ARCH Model

Autoregressive conditional heteroskedasticity or ARCH is a univariate and nonlinear statistical model that is used for time series data that describes the variance of the current error terms as a function of the actual sizes of the previous term periods' error terms. One aspect of the ARCH model is that large changes in the volatility clustering tend to be followed by large changes of either sign, or small changes in the volatility clustering are followed by small changes of either sign.

To determine the parameter p (also known as the lag) of the ARCH model, Bayesian Information Criteria (BIC) is used. We chose our model and lag value such that the BIC value is minimized. BIC is used since we have many samples and is also proven to be a reliable metric for model selection.

3.2.1.2 GARCH Model

Generalized AutoRegressive Conditional Heteroskedasticity or the GARCH model is a statistical model used for time series data and is an extension of the ARCH model but with lagged conditional variance. The GARCH model has parameter p which is the number of lagged squared returns and parameter q which is the number of lagged conditional variance.

We select our GARCH model in a similar way as the ARCH model. The model with parameters p and q is selected such that the BIC value is minimized.

3.2.1.3 GJR-GARCH Model

GJR-GARCH or the Glosten, Jagannathan and Runkle GARCH model is a GARCH-type model which performs well in modelling asymmetric effects of announcements such that bad news has a larger impact than good news [1].

We select our GJR-GARCH model in a similar way as the GARCH model. The model with parameters p and q is selected such that the BIC value is minimized.

3.2.1.4 EGARCH Model

EGARCH or Exponential GARCH was a model proposed to overcome the weakness of the GARCH model in handling financial time series. EGARCH is expressed in logarithms of the conditional variance volatility [7]. The GJR-GARCH model is selected in a similar way as the GARCH model with parameters p and q such that the BIC value is minimized.

3.2.1.5 HAR-RV Model

HAR-RV Model or the Heterogenous AutoRegressive of Realized Volatility model was designed to parsimoniously capture the strong persistence typically observed in realized volatility [3]. The model is estimated using realized volatility through the method of ordinary least squares (OLS). In this method, we are forecasting realized volatility of one day by considering the realized volatility of the previous week and previous month.

3.2.2 Machine Learning Approaches

Below are the machine learning approaches implemented in this project.

3.2.2.1 SVM-GARCH with Linear Kernel and RBF Kernel

Support Vector Machines is a supervised machine learning technique in which the aim is to find a line that separates two classes. Since many lines can fit, the algorithm tries to find the optimal line. This line is called a hyperplane. In regression problems, the algorithm aims to a hyperplane such that the error is minimized, and the margin is maximized. This method is called support vector regression (SVR) and this method is applied to the GARCH model.

The SVR-GARCH model is basically SVR based on GARCH. In the parameter selection part of the SVR model, a validation procedure based on randomized search is used to select the two free parameters ϵ (loss function parameter) and C (regularization parameter) and one kernel coefficient γ. The project draws inspiration from the approach used in [4] but uses randomized search over grid search for the parameters.

The project implements two approaches of the SVM-GARCH – one with the linear kernel and the other with the RBF (radial basis function) kernel.

3.2.2.2 Neural Networks

Neural networks are building blocks of deep learning which are inspired by the human brain. Using Keras and Tensorflow, the project specifies two hidden layers of 256 and 128 neurons each with each hidden layer having the ReLU activation function. One output neuron was used since volatility is of continuous type. This model was compiled using MSE as the loss function and the RMSprop optimizer. To select the best model, we pick the model with the lowest RMSE. Best model is selected by varying the batch size and epochs from 100 to 400.

4. Visualization and Deployment

The results of the performance of the above models is visualized and the workflow is implemented for each company in the S&P 500 index and the models are deployed on Streamlit. The link is found here - https://share.streamlit.io/kedarghule/stock-portfolio-diversification-using-clustering-and-volatility-prediction/main/main.py.

Note - Please use Mozilla Firefox only for the above link. This is due to some rendering issues of Plotly with other browsers.

The source code of the project can be found in the References section at [9].

Results

Below are the results obtained from this project.

1. **K-Means Clustering:** The below figure shows the weekly percentage of returns for stocks grouped together in a cluster.
    a. **Cluster 1 –** 305 stocks (e.g. Bank of America, AT&T).
    b. **Cluster 2 –** 6 stocks (e.g. AMD, Tesla).
    c. **Cluster 3 –** 40 stocks (e.g. Apple, Moderna).
    d. **Cluster 4 –** 1 stock (NVIDIA).
    e. **Cluster 5 –** 152 stocks (e.g. Google, Texas Instruments).

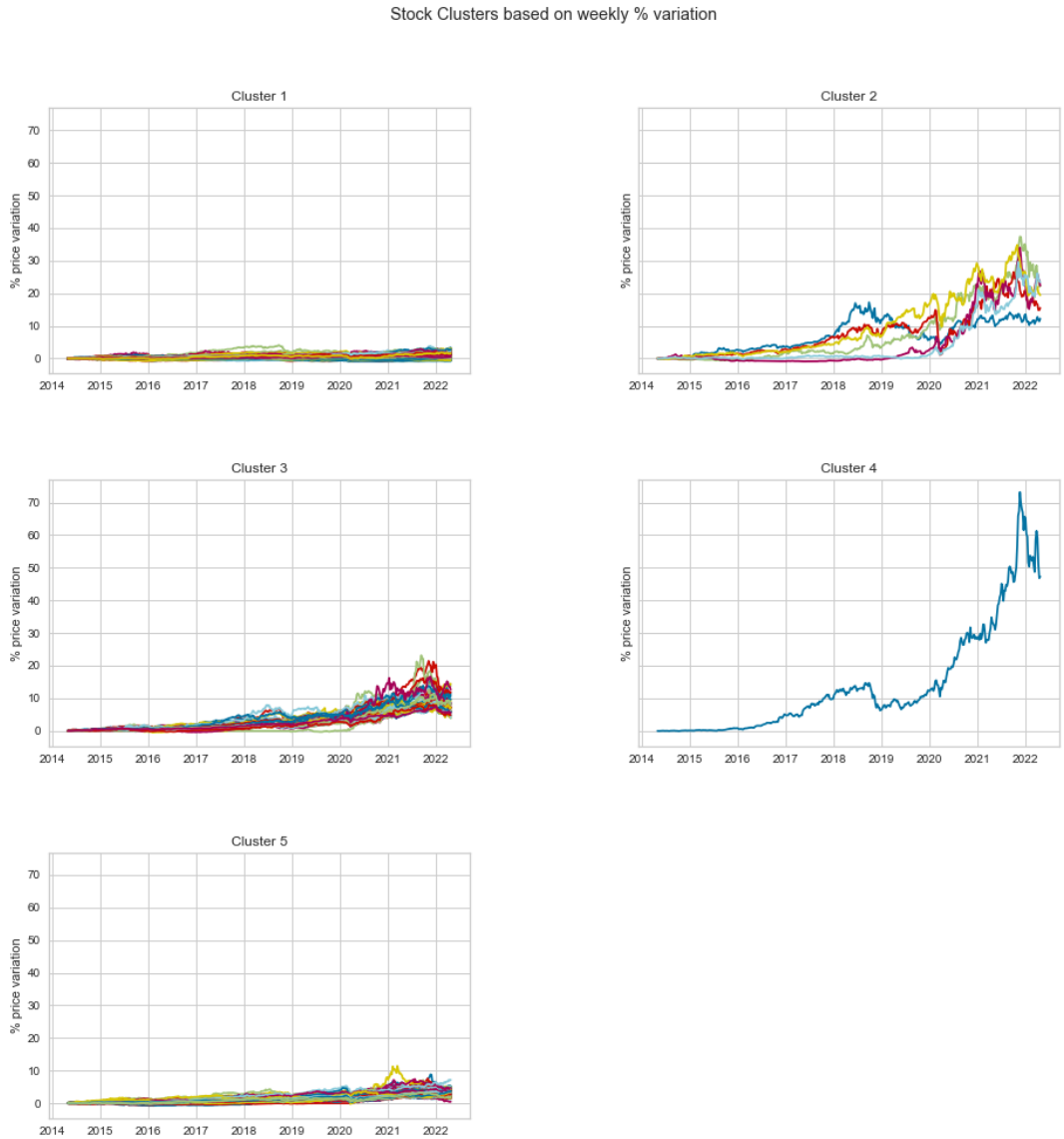

Figure 5. Stock Clusters based on weekly percentage variation after K-Means Clustering.

The Silhouette score of 0.4568 was achieved by this clustering algorithm.

One major inference from this is that Stocks that are volatile for a long time will remain volatile while stocks that are non-volatile for a long time, will remain non-volatile.

Below is a group of pie charts showing which sector is prominent in what cluster. This is essential to get insights as to which sector is more volatile or not.

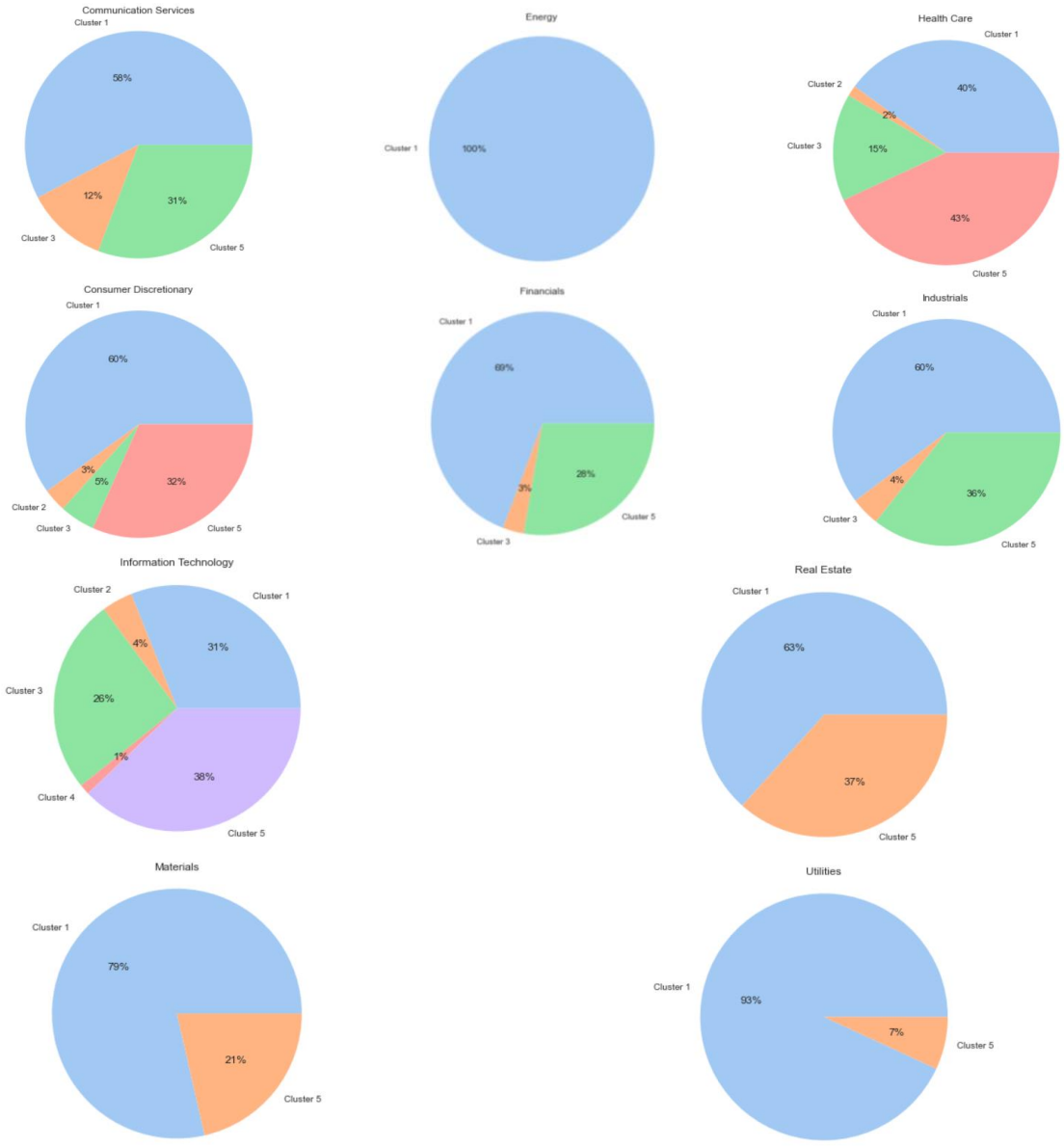

Figure 6. Pie-charts showing prominence of clusters in each sector.

2. Volatility Clustering and Prediction.

Below is a graph showing the volatility clustering of the S&P 500 index.

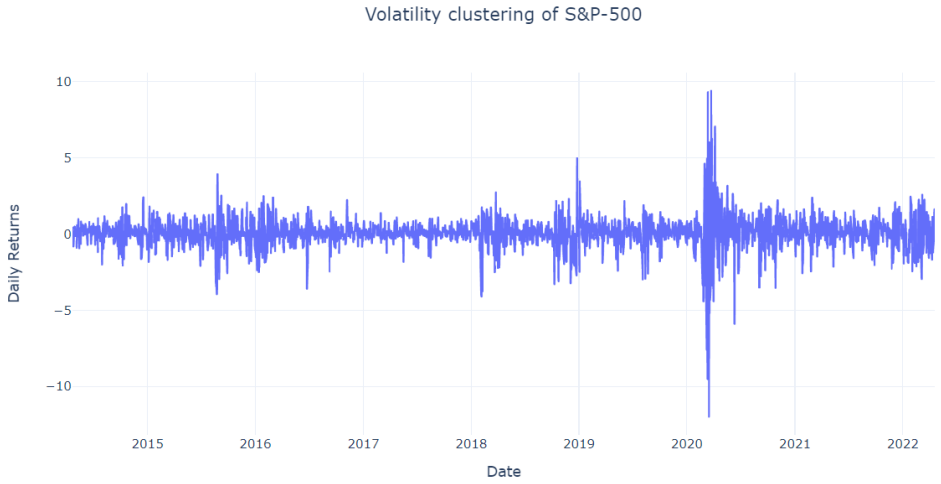

Figure 7. Volatility Clustering of S&P 500 index.

Below is a table showing the percentage volatility for each of the periods -daily, monthly, and annual.

Period	Percentage Volatility
Daily Volatility	1.11
Monthly Volatility	5.07
Annual Volatility	17.56

Figure 8. Percentage volatility table.

Following were the best parameter values for the classical volatility models with low BIC.

- ARCH with parameter p =4.
- GARCH with parameter p =1 and q =1.
- GJR-GARCH with parameter p = 1 and q = 1.
- EGARCH with parameter p = 1 and q = 1.

Below is a table showing the results of the volatility prediction models on the realized volatility of the S&P 500 index data.

Model	RMSE Value	MSE Value	MAE Value
ARCH	0.0887229	0.0078718	0.0842505
GARCH	0.0868563	0.007544	0.0830895
GJR-ARCH	0.0889648	0.0079147	0.0833105
EARCH	0.0897355	0.0080525	0.0860906
SVR-GARCH (Linear)	0.000388	2e-7	0.0002953
SVR-GARCH (RBF)	0.0007149	5e-7	0.0005929
Neural Networks	0.0006601	4e-7	0.0004904
HAR-RV	0.002458	0.000006	0.0018505

Figure 9. Percentage volatility table.

RMSE, MSE and MAE were used as metrics however, the RMSE value was used as the primary metric to evaluate the models. We can observe that the machine learning models outperform classical volatility models. It is observed that the neural networks perform the best followed by the SVR-GARCH models and then the HAR-RV model and then GARCH. Among the classical volatility models, GARCH performs the best. However, we should also note that there is not major performance difference between the classical volatility models.

Below are the prediction results for the models implemented.

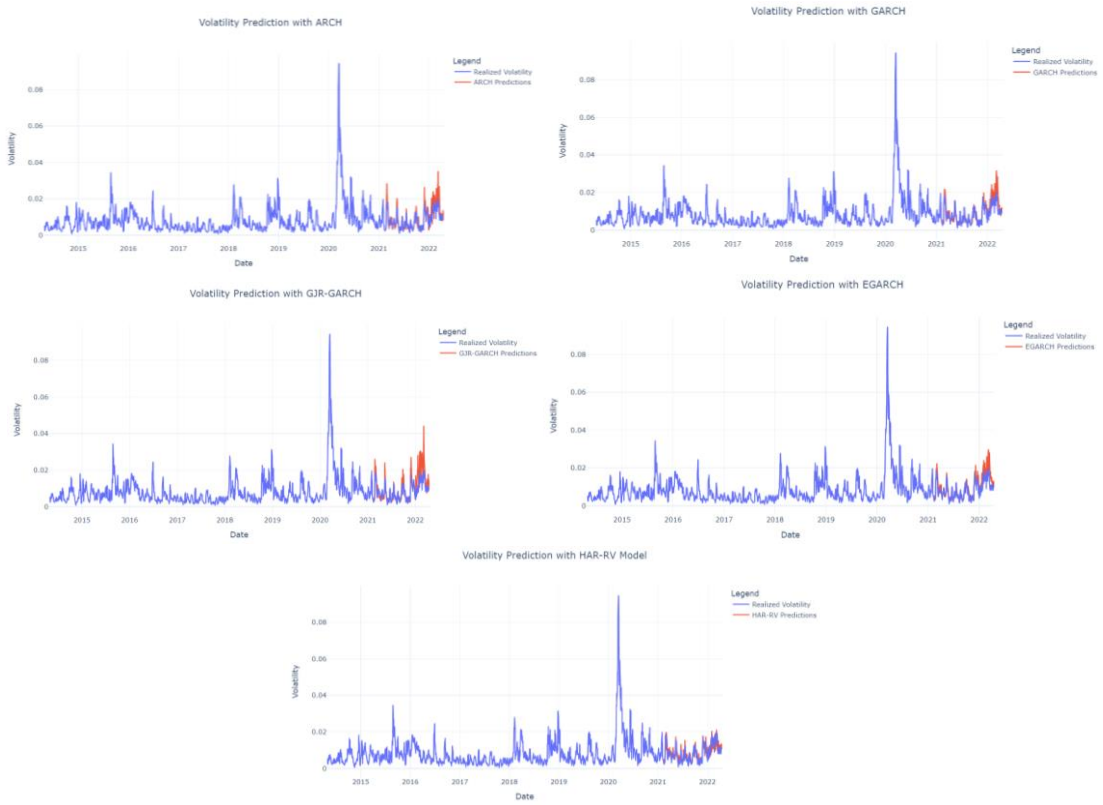

Figure 10. Volatility Prediction results of the classical volatility models.

Figure 11. Volatility Prediction results of the machine learning models.

You can see from the prediction results graphs that the machine learning models clearly outperform the classical volatility models.

Future Scope

As a future score and if this project were a larger research undertaking, following are some areas that would be explored.

- Prediction of volatility by considering an implied volatility index like VIX or daily traded volume of the stock. This can be integrated with the HAR-RV model and may show better performance.
- Prediction of volatility of traditionally neglected categories like municipal bonds and how to leverage them in a low-risk diverse portfolio.
- Implementing clustering on sector indexes to see which overall industry is more volatile than the others. Volatility prediction can also be done on these sector indexes.
- Analyzing different commodities like crude oil, gold, silver and so on and measuring the impact of volatility in these commodities on the stock data.
- Forecasting US stock market volatility using the volatility of international stock market data like that of G7 countries, China and so on. Examples of international stocks include NIKKEI 225 of Japan, CAC 40 of France, FTSE 100 of the United Kingdom and so on.

References

[1] Machine Learning for Financial Risk Management with Python by Abdullah Karasan published by O'Reilly.

[2] https://en.wikipedia.org/wiki/List_of_S%26P_500_companies

[3] A Practical Guide to Harnessing the HAR Volatility Model by Adam Clements and Daniel P. A. Preve.

[4] Sun, H., Yu, B. Forecasting Financial Returns Volatility: A GARCH-SVR Model. Comput Econ 55, 451–471 (2020). https://doi.org/10.1007/s10614-019-09896-w

[5] Haugom, Erik and Haugom, Erik and Langeland, Henrik and Molnár, Peter and Westgaard, Sjur, Forecasting Volatility of the U.S. Oil Market (January 29, 2014). Available at SSRN: https://ssrn.com/abstract=2691391 or http://dx.doi.org/10.2139/ssrn.2691391

[6] Dritsaki, C. (2017) An Empirical Evaluation in GARCH Volatility Modeling: Evidence from the Stockholm Stock Exchange. Journal of Mathematical Finance, 7, 366-390. doi: 10.4236/jmf.2017.72020.

[7] Nelson, D.B. (1991) Conditional Heteroskedasticity in Asset Returns: A New Approach. Econometrica, 59, 347-370. https://doi.org/10.2307/2938260

[8] Karasan, Abdullah and Esma Gaygısız. "Volatility Prediction and Risk Management: An SVR-GARCH Approach." (2020).

[9] Project Source Code: https://github.com/kedarghule/Stock-Portfolio-Diversification-Using-Clustering-and-Volatility-Prediction