Least Squares Approximation in Financial Mathematics

A PROJECT REPORT SUBMITTED
FOR THE REQUIREMENTS OF

Math 104A (Numerical Analysis)

By

Anthony Scattini
Henry Hartwell
Peyton McGuinness
Raghav Thondikulam
Zak Spero

Department of Mathematics
University of California, Santa Barbara 2024

Table of Contents

Introduction

In the vast and dynamic landscape of financial markets, the ability to predict stock volatility is essential for creating investment strategies and risk management. Predicting volatility is one of the most difficult tasks in the financial industry given the ever-changing economy and populus trends. By leveraging historical stock and volatility trends and a selection of macroeconomic indicators, we aim to construct a durable model for volatility of the S&P 500 using the method of least squares approximation.

We will introduce some information vital to reading and understanding the remainder of the project:

Stock Volatility

Stock volatility measures the degree of variation in a stock's trading price over time. It is quantified by statistical measures like standard deviation or variance, indicating the extent of price fluctuations from the mean. Volatility can be historical, derived from past market data, or implied, reflecting market expectations for future price movements. Investors use volatility for risk management, and it plays a crucial role in determining option prices and understanding market dynamics.

We will use the VIX, or Volatility Index, as a baseline to compare our volatility metrics. The VIX is a financial metric that gauges market expectations for future volatility. This is commonly known as the "fear index," and is calculated based on the prices of options. It reflects investors' consensus on the anticipated level of market volatility over the next 30 days. A higher VIX value suggests increased expected volatility, often indicating higher perceived risk in the market, while a lower value implies lower expected volatility.

Macroeconomic Trends

Macroeconomic trends refer to the long-term, broad patterns and developments in an economy as a whole. These trends encompass significant indicators that shape the overall economic landscape. They provide a comprehensive view of an economy's performance and help individuals make informed decisions about future economic conditions.

The Macroeconomic trends we used to evaluate in our project are:

- Federal Interest Rate- benchmark rate set by the Federal Reserve which influences borrowing costs for banks
- Inflation Rates
- CPI Data - The Consumer Price Index (CPI) is a measure of the average change over time in the prices paid by urban consumers for a market basket of consumer goods and services.
- Interest Rates
- Technological Advancement

Least Squares Approximation

Least squares approximation is a mathematical method used to find the line of best fit for a linear relationship between variables. The method minimizes the sum of squared errors between observed and predicted values. This technique is commonly used in regression analysis to estimate coefficients for a linear model, allowing a straightforward and optimal way to fit lines to data points.

We will attempt to use this method for the historical volatility and macroeconomic trends to weigh the extent of the impact of each factor. This will allow us to best understand the fluctuations in volatility as well as which factors are necessary to our model.

S&P 500

The S&P 500 (Standard & Poor 500) is a stock market index that measures the performance of 500 large publicly traded companies. The index is a market-capitalization-weighted index, meaning that larger companies have a greater impact on its value.

Motivation:

The motivation behind this project is to examine if the numerical methods, in this case the discrete least squares approximation, we have learned in Math 104A can assist us in improving upon existing volatility forecasting models. Volatility forecasting is an essential factor in the financial industry, especially for portfolio optimization and risk management.

Problem Statement:

Investigating whether Macroeconomic indicators improve one day out volatility forecasting. Our objective is to determine volatility forecasting using macroeconomic indicators and determine what the next steps are to improving volatility forecasting.

References and History

The development of linear regression can be traced back to the early 19th century, with Carl Friedrich Gauss and Adrien-Marie Legendre introducing the method of least squares. This method aims to minimize the sum of the squared differences between observed and predicted values, providing the best-fitting line to the data. Francis Galton contributed significantly to linear regression in the late 19th century by introducing correlation and regression concepts and developing techniques for estimating regression coefficients. In the 20th century, advancements in computing and statistics led to the widespread adoption of linear regression across various fields.

Least squares estimation forms the basis of linear regression modeling, where the coefficients of the regression equation are estimated by minimizing the sum of squared residuals between observed and predicted values. This approach provides an algorithmic way to find the line that best fits the data.

The basics of our linear regression is well described by Hector D. Ceniceros in his unpublished work, "Introduction to Numerical Analysis" that we have been provided. In this work, in chapter 4 covering least squares approximation, he covers the discrete case in which "given a data set $(x_0, f_0),(x_1, f_1), \cdots ,(x_N , f_N)$ and we would like to find the best 2-norm approximation" (131). In our case, the best 2-norm approximation comes from the equation:

$$// f - p_n //^{2} = {}^{N}\sum_{j=0} |f_j - p_n(x_j)|\ 2 = 0$$

Ceniceros continues to write, "The solution of the discrete least square problem is again characterized by the orthogonality of the error and we can write the least squares approximation $f * \in W$ explicitly when the set of functions $\{\phi_0, \phi_1, . . . , \phi_n\}$ is orthogonal with respect to the inner product. $W = P_n$ is often used for data fitting, particularly for small n. It is worth noting that when $N = n$ the solution to the discrete least squares problem in P_n is the interpolating polynomial p_n of the data, for [the above equation]. The case $W = P_1$ is also known as linear regression." As we are working with discrete data points from our VIX pricing points and macroeconomic data sets, we apply this method of least squares approximation, in particular the case in which $W = P_1$, creating a linear regression model for our data.

In the calculation of our linear regression model, we utilized volatility data gathered from the S&P 500. To calculate this volatility we applied the Garman-Klass volatility formula to obtain an estimator for future daily, weekly and monthly volatility. In Peter Molnár's paper, "Properties of Range-Based Volatility Estimators", he covers the difference in accuracy and applicability between multiple volatility estimators and states in his abstract that, "we analyze properties of these estimators and find that the best estimator is the Garman–Klass (1980) estimator" (1). Traditionally, the simplest form of realized volatility is calculated using the squared difference of closing prices between days. However, this method omits important intraday data such as the difference between high and low trading prices as well as opening prices. As Molnár mentions, “It is intuitively clear that the difference between high and low prices tells us much more about volatility than close price. High and low prices provide additional information about volatility” (2). This difference between high and low prices is known as the range of the stocks price throughout the day and the most effective estimators utilize this information to make more educated predictions about future values as “range-based volatility estimators provide significant increase in accuracy compared to simple squared returns” (9). With this in mind, the obvious choice is to utilize a range based estimator for our calculations. Furthermore, when choosing between possible estimators, Molnar concludes that , “... for most purposes, the best volatility estimator is the Garman–Klass volatility estimator” (9).

Methodology: Least Squares Approximation

Our regression model follows a simple flow to make its interpretation easy to understand and run. It can be broken down into data initialization, data separation, the linear regression model itself, the predictions, and the visualization.

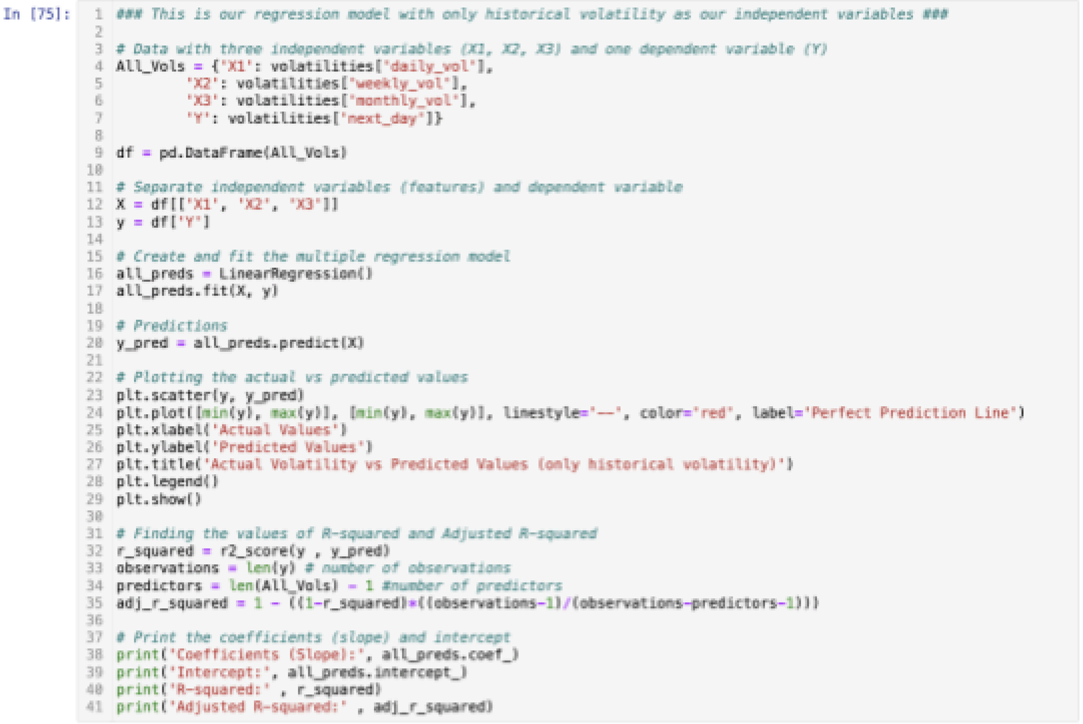
To initialize the data, we construct a dictionary containing the independent variables, denoted “X1”, “X2”, “X3”, …, and the dependent variable denoted “Y”. We use this data to create a pandas data frame in which each column corresponds to a different X variable and the Y variable.

Then, we separate the data into our independent and dependent variables. The independent variables construct a data frame of its own, and the dependent variable constructs its own data frame.

The linear regression model itself comes from the “scikit-learn” package. This function finds the coefficients and intercept that best fits the data between the independent and dependent variables.

We can then use the model to create “y_pred,” predictions based on the input features. This allows us to visualize the effects of multiple independent variables on a 2-dimensional graph.

Further, we visualize the outputs of the regression model in a scatter plot which includes the prediction line, predicted values and actual values, and the slope coefficients along with the intercept term.



```
In [75]:
### This is our regression model with only historical volatility as our independent variables ###

# Data with three independent variables (X1, X2, X3) and one dependent variable (Y)
All_Vols = {'X1': volatilities['daily_vol'],
        'X2': volatilities['weekly_vol'],
        'X3': volatilities['monthly_vol'],
        'Y': volatilities['next_day']}

df = pd.DataFrame(All_Vols)

# Separate independent variables (features) and dependent variable
X = df[['X1', 'X2', 'X3']]
y = df['Y']

# Create and fit the multiple regression model
all_preds = LinearRegression()
all_preds.fit(X, y)

# Predictions
y_pred = all_preds.predict(X)

# Plotting the actual vs predicted values
plt.scatter(y, y_pred)
plt.plot([min(y), max(y)], [min(y), max(y)], linestyle='--', color='red', label='Perfect Prediction Line')
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')
plt.title('Actual Volatility vs Predicted Values (only historical volatility)')
plt.legend()
plt.show()

# Finding the values of R-squared and Adjusted R-squared
r_squared = r2_score(y , y_pred)
observations = len(y) # number of observations
predictors = len(All_Vols) - 1 #number of predictors
adj_r_squared = 1 - ((1-r_squared)*((observations-1)/(observations-predictors-1)))

# Print the coefficients (slope) and intercept
print('Coefficients (Slope):', all_preds.coef_)
print('Intercept:', all_preds.intercept_)
print('R-squared:' , r_squared)
print('Adjusted R-squared:' , adj_r_squared)
```

Figure: Linear Regression Model Visualization

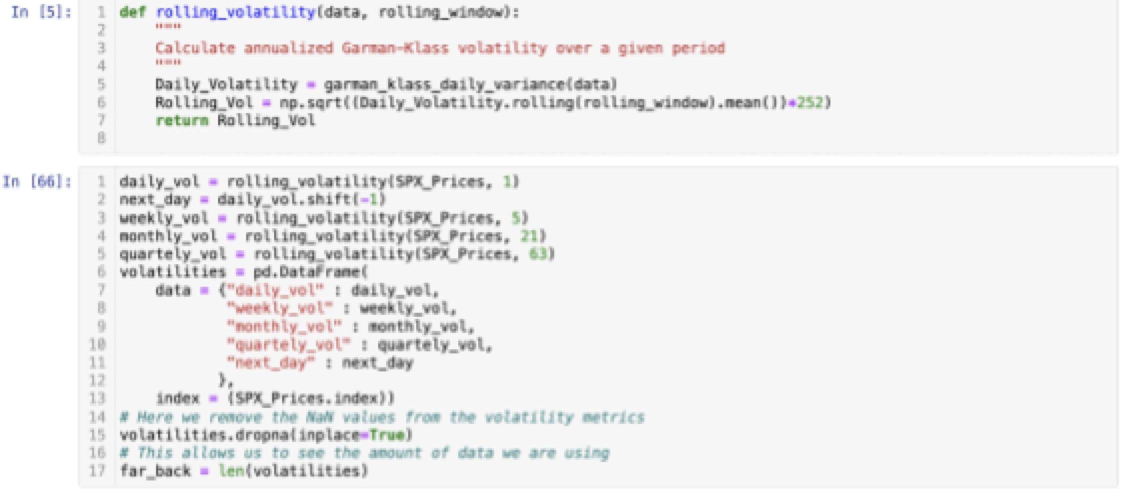
Data Collection and Preparation

Our data was collected from numerous sources including Yahoo Finance, the FRED (Federal Reserve Economic Data), and Cboe Global Markets. The macroeconomic variables we attempted to integrate into our model are the Federal Interest Rate, Inflation Rate, CPI, and Real GDP. We found these variables to be easiest to understand and collect.

SPX Historical Data

The SPX historical data was pulled from the Yahoo Finance library (yfinance) built into python. Our data ranges from 2002 through 2023. From yfinance, we constructed code to return the rolling volatility, which we then used to calculate the rolling volatility on a daily, weekly, monthly, and quarterly basis.

The different rolling volatility times took varying data entries to compute their first volatility value, so we had to cut off the first few entries from the longer data sets. This ensured all of our data was uniform length, allowing us to construct any functions necessary.



```
In [5]:
def rolling_volatility(data, rolling_window):
    """
    Calculate annualized Garman-Klass volatility over a given period
    """
    Daily_Volatility = garman_klass_daily_variance(data)
    Rolling_Vol = np.sqrt((Daily_Volatility.rolling(rolling_window).mean())*252)
    return Rolling_Vol

In [66]:
daily_vol = rolling_volatility(SPX_Prices, 1)
next_day = daily_vol.shift(-1)
weekly_vol = rolling_volatility(SPX_Prices, 5)
monthly_vol = rolling_volatility(SPX_Prices, 21)
quartely_vol = rolling_volatility(SPX_Prices, 63)
volatilities = pd.DataFrame(
    data = {"daily_vol" : daily_vol,
            "weekly_vol" : weekly_vol,
            "monthly_vol" : monthly_vol,
            "quartely_vol" : quartely_vol,
            "next_day" : next_day
           },
    index = (SPX_Prices.index))
# Here we remove the NaN values from the volatility metrics
volatilities.dropna(inplace=True)
# This allows us to see the amount of data we are using
far_back = len(volatilities)
```

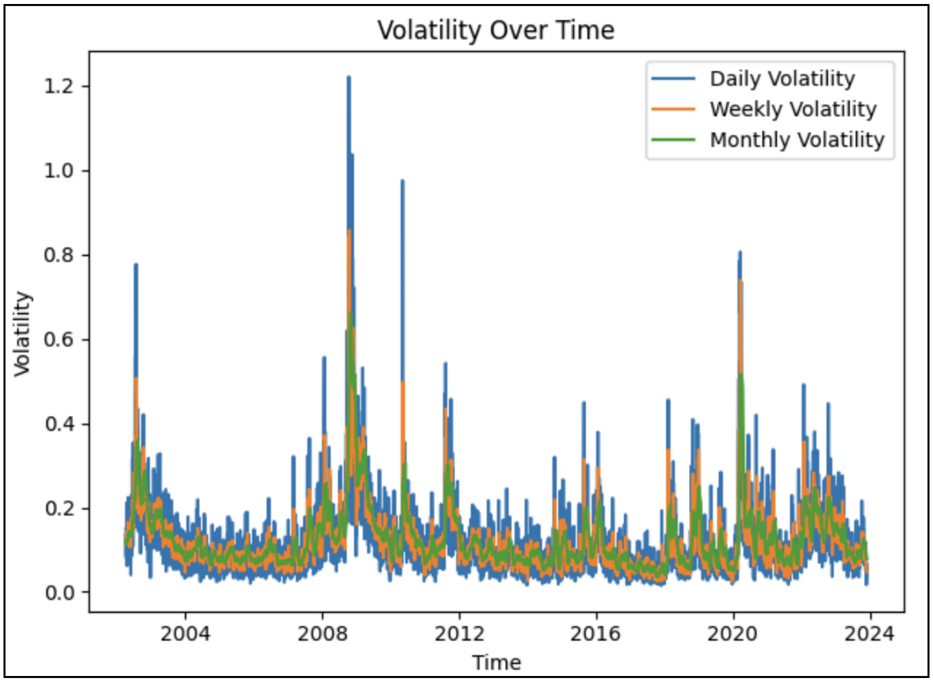
Figure: SPX Data Collection and Correction

SPX Volatility

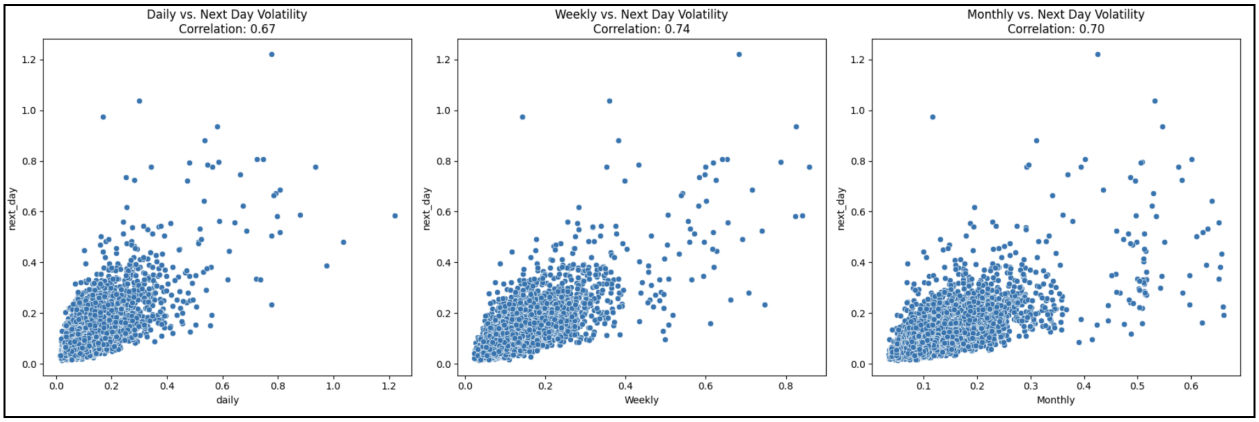
We calculated the daily, weekly and monthly annualized volatility on a daily basis using the garman-klass volatility estimator. Garman-Klass (GK) volatility estimator consists of using the returns of the open, high, low, and closing prices in its calculation. It is calculated as follow,

$$GKHV = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\left(ln\frac{h_i}{l_i}\right)^2 - \frac{1}{N}\sum_{i=1}^{N}(2ln2-1)\left(ln\frac{c_i}{o_i}\right)^2}$$

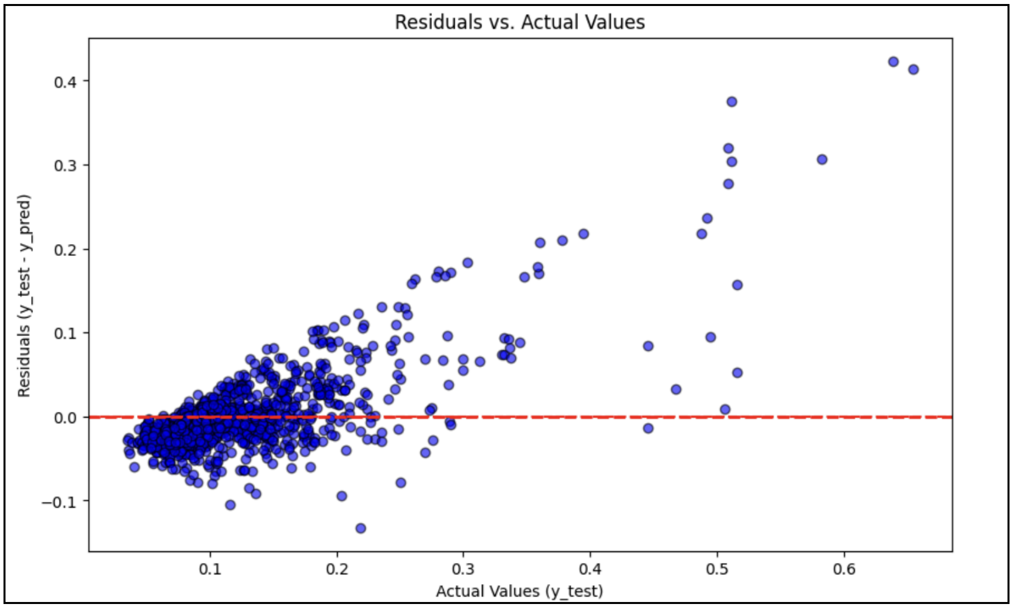
where hi denotes the daily high price, li is the daily low price, ci is the daily closing price and oi is the daily opening price. We implemented the above equation in Python. We downloaded SPY data from Yahoo finance and calculated GK historical volatility using the Python program. The picture below shows the GK historical volatility of SPY from January 2002 to January 2023.



Next we display each of the independent variables (daily, weekly, and monthly volatility) correlation with the next day's volatility. It is evident that as volatility increases the relationships between the independent variables and the next day's volatility contains significantly more variance. This problem is known as heterogeneity, and is a reason for concern when utilizing a linear regression model.



This non-constant variance may be seen further examining the relationship between the actual next day's volatility and the residuals:



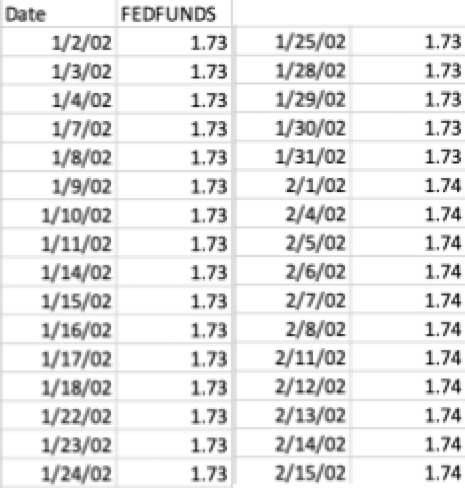
Future projects may examine how a weighted least squares model may improve the forecasting ability of next day's volatility by accounting for realized quarticity and correlated error terms.

Federal Interest Rates

Information of the federal interest rates was extracted directly from FRED. It was downloaded as a csv file. Then, we edited the data in excel to ease its use in our python code. The interest rate data comes in monthly terms (giving the interest rate for the first of every month), so we decided to duplicate the monthly interest rate for each day in that month. Of course, we only considered the 252 days that consist of a stock market cycle.

Once our excel work was completed, we uploaded the csv file to our python code. Then, we had to go through a few edits to make sure our data could be read and added to the regression model. As seen in the figure below, the interest rates are constant by month (1.73 for all of January 2002), and skips inactive days in the market (skips from 1/4/02 to 1/7/02).

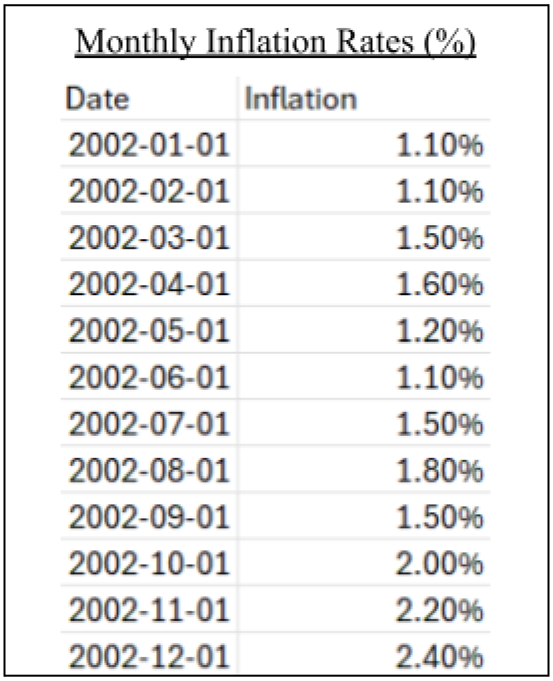
Figure: Interest Rates Data in Excel



Date	FEDFUNDS		
1/2/02	1.73	1/25/02	1.73
1/3/02	1.73	1/28/02	1.73
1/4/02	1.73	1/29/02	1.73
1/7/02	1.73	1/30/02	1.73
1/8/02	1.73	1/31/02	1.73
1/9/02	1.73	2/1/02	1.74
1/10/02	1.73	2/4/02	1.74
1/11/02	1.73	2/5/02	1.74
1/14/02	1.73	2/6/02	1.74
1/15/02	1.73	2/7/02	1.74
1/16/02	1.73	2/8/02	1.74
1/17/02	1.73	2/11/02	1.74
1/18/02	1.73	2/12/02	1.74
1/22/02	1.73	2/13/02	1.74
1/23/02	1.73	2/14/02	1.74
1/24/02	1.73	2/15/02	1.74

Inflation Rates

Information on inflation rates was gathered from the Bureau of Labor Statistics monthly reports of inflation rates. It was extracted into an excel spreadsheet and reformatted to a CSV file in order to be accessed and modified in our Python code. As the data was originally organized monthly, it was transformed into daily numbers to match the daily volatility values being used. This was achieved by averaging the monthly values across days in order to achieve a consistent format.

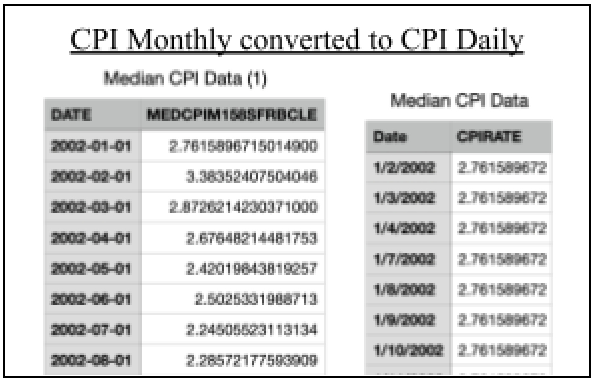


Monthly Inflation Rates (%)

Date	Inflation
2002-01-01	1.10%
2002-02-01	1.10%
2002-03-01	1.50%
2002-04-01	1.60%
2002-05-01	1.20%
2002-06-01	1.10%
2002-07-01	1.50%
2002-08-01	1.80%
2002-09-01	1.50%
2002-10-01	2.00%
2002-11-01	2.20%
2002-12-01	2.40%

CPI Data

Information of the CPI (Consumer Price Index) Data was extracted directly from FRED. It was downloaded as a CSV file. Then, we edited the data in Excel to ease its use in our Python code. Like interest rates, the CPI data comes in monthly figures so we had to break it down into daily numbers. We did so by using the same monthly figures for every day in that month through a formula in Excel and applying that to our entire data set.



CPI Monthly converted to CPI Daily

Median CPI Data (1)

DATE	MEDCPIM158SFRBCLE
2002-01-01	2.7615896715014900
2002-02-01	3.38352407504046
2002-03-01	2.8726214230371000
2002-04-01	2.67648214481753
2002-05-01	2.42019843819257
2002-06-01	2.5025331988713
2002-07-01	2.24505523113134
2002-08-01	2.28572177593909

Median CPI Data

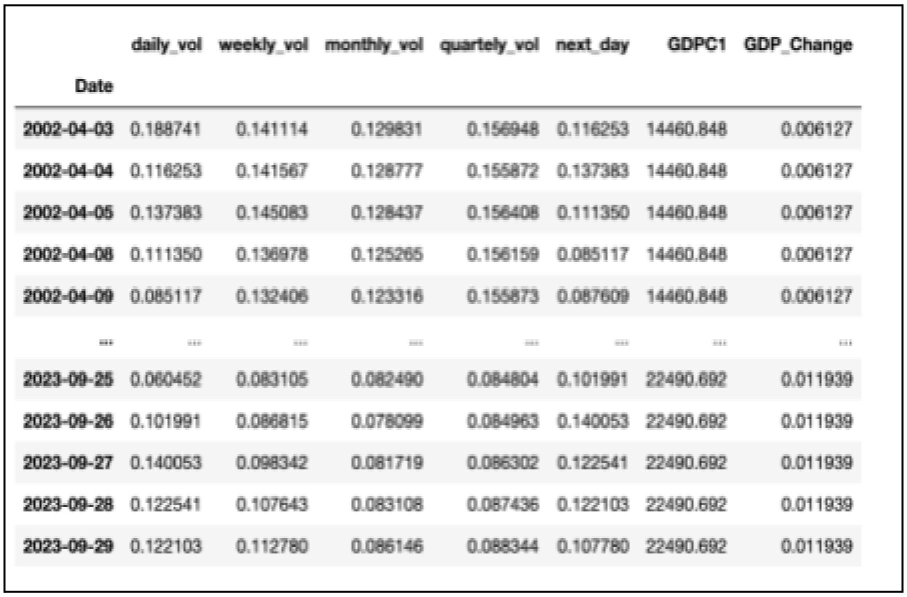
Date	CPIRATE
1/2/2002	2.761589672
1/3/2002	2.761589672
1/4/2002	2.761589672
1/7/2002	2.761589672
1/8/2002	2.761589672
1/9/2002	2.761589672
1/10/2002	2.761589672

Real GDP

Information of the real quarterly GDP was collected from the FRED. The data was downloaded as a CSV file, and then in the code, we used a pathfile name to extract the file and implement it into our functions.

Some trouble we encountered was how the real GDP was only given in quarterly values, so when trying to immediately implement this into our code, we experienced a length error which did allow the functions to run. Therefore, we needed to duplicate the data into a daily model where the data remained constant until the next quarter and then the code would run. The data is included below for reference (GDPC1).

We then coded the data so the GDP percent change was implemented into our regression model to show the scale of change between each quarter. This is also included below in the data table (GDP_change).

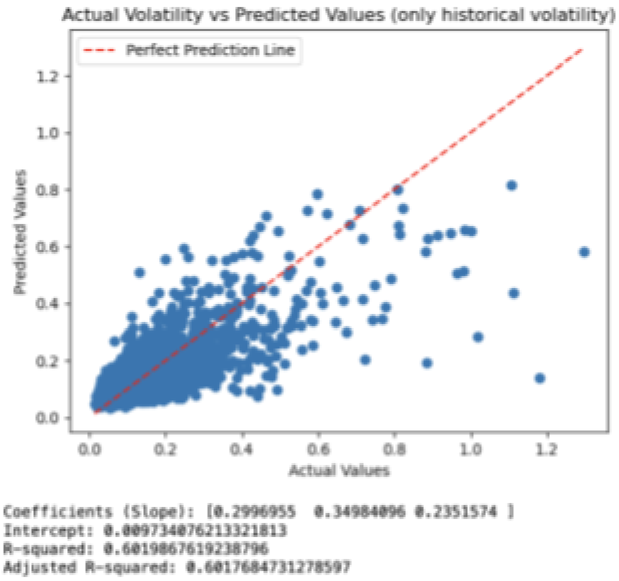


	daily_vol	weekly_vol	monthly_vol	quartely_vol	next_day	GDPC1	GDP_Change
Date							
2002-04-03	0.188741	0.141114	0.129831	0.156948	0.116253	14460.848	0.006127
2002-04-04	0.116253	0.141567	0.128777	0.155872	0.137383	14460.848	0.006127
2002-04-05	0.137383	0.145083	0.128437	0.156408	0.111350	14460.848	0.006127
2002-04-08	0.111350	0.136978	0.125265	0.156159	0.085117	14460.848	0.006127
2002-04-09	0.085117	0.132406	0.123316	0.155873	0.087609	14460.848	0.006127
...	...	...	...	...	...	...	...
2023-09-25	0.060452	0.083105	0.082490	0.084804	0.101991	22490.692	0.011939
2023-09-26	0.101991	0.086815	0.078099	0.084963	0.140053	22490.692	0.011939
2023-09-27	0.140053	0.098342	0.081719	0.086302	0.122541	22490.692	0.011939
2023-09-28	0.122541	0.107643	0.083108	0.087436	0.122103	22490.692	0.011939
2023-09-29	0.122103	0.112780	0.086146	0.088344	0.107780	22490.692	0.011939

Independent Variable Selection and Results

In our project we use the historical volatility trends on a daily, weekly, and monthly rolling basis to estimate future volatility. We also wanted to pair this data with one to two macroeconomic variables to see if we could better the approximation without leading to problems in our model like "overfitting" and "multicollinearity." (Overfitting occurs when a model is too complex and multicollinearity occurs when one or more independent variables are correlated to another.) Our goal is to find macroeconomic variables that better our approximation and are not correlated with another or the rolling historical volatility.

Our base case, of sorts, is the model which did not include any macroeconomic variables. We can see the graph of this below. We wanted a model which visually betters the model as well as increases adjusted R-squared. Adjusted R-squared is a metric which indicates how much of the variance of a regression model can be attributed to the regression model itself, while also taking into account the number of predictors. A higher R-squared indicates a better model.

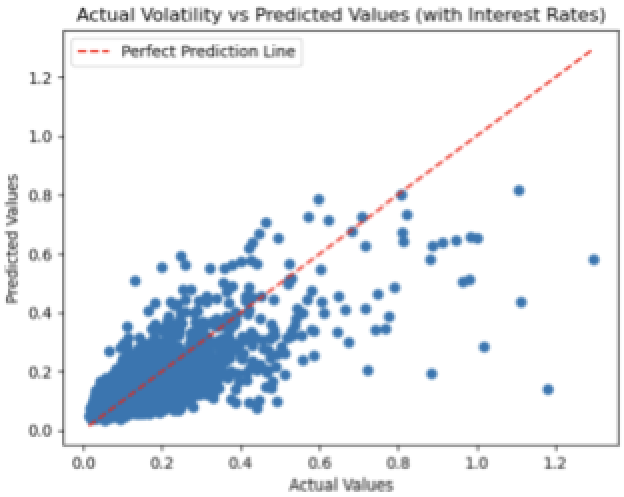


```
Coefficients (Slope): [0.2996955  0.34984096 0.2351574 ]
Intercept: 0.009734076213321813
R-squared: 0.6019867619238796
Adjusted R-squared: 0.6017684731278597
```

Figure: Linear Regression Model with No Macroeconomic Variables Visualization

Federal Interest Rates

When using the real interest rate as our macroeconomic variable, we saw very little change in our model. The historical volatility parameters changed by minute amounts and the macroeconomic variable only had a regression coefficient of .000011. Further, adjusted R-squared (though not by much) decreased from .60176 to .60177.

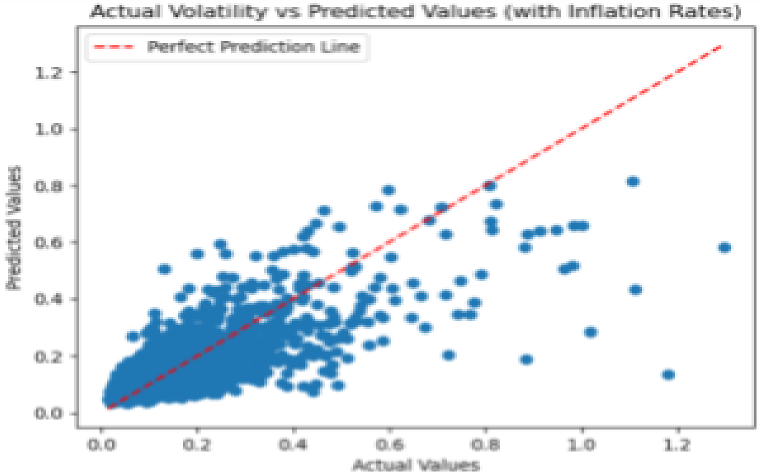


```
Coefficients (Slope): [2.99678461e-01 3.49730130e-01 2.35555673e-01 1.12166774e-04]
Intercept: 0.009530628310783l2
R-squared: 0.601990357649682
Adjusted R-squared: 0.6016992553330973
```

Figure: Linear Regression Model with Real Interest Rates Visualization

Inflation Rates

When applying inflation rates to our linear regression model, we saw a minimal but noticeable change in our model. The historical volatility information was altered slightly but still comprised the majority of the fitting. The inflation rate coefficient was 0.0869 which is small when compared to the volatility measures but quite substantial compared to the other macroeconomic variables. However, the adjusted R-squared value decreased from 0.6022 to .06019.

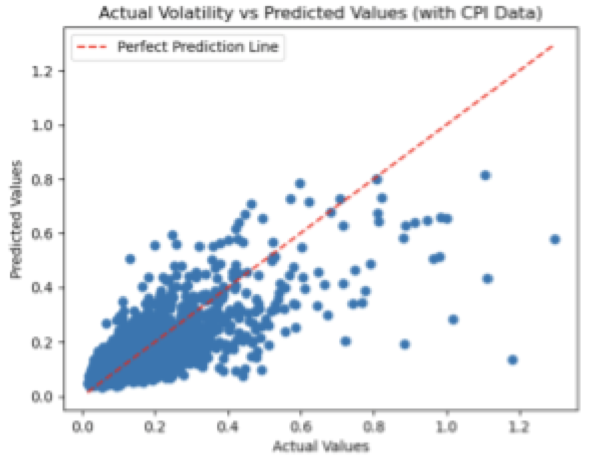


```
Coefficients (Slope): [0.29918094 0.34748693 0.23840368 0.08690649]
Intercept: 0.007468866737214169
R-squared: 0.6022514878992942
Adjusted R-squared: 0.601960576572104
```

Figure: Linear Regression Model with Inflation Rates Visualization

CPI Data

When using the Daily CPI Data as our macroeconomic variable, there is still a small change, however in the opposite direction compared to interest rates. Further, adjusted R-squared increased from .60179 to .60177.

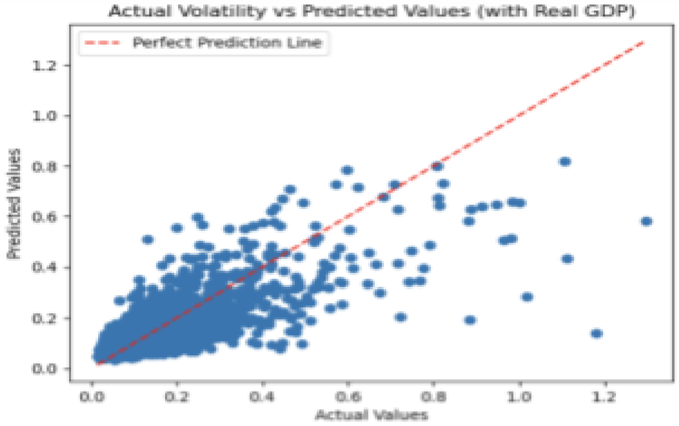


```
Coefficients (Slope): [0.29941964 0.34936615 0.2363022  0.0007142 ]
Intercept: 0.007729433442559552
R-squared: 0.6020899545233773
Adjusted R-squared: 0.6017989250514617
```

Figure: Linear Regression Model with CPI Data Visualization

Real GDP Percent Change

When using the real GDP percent change on a monthly basis, it was unable to improve our forecasting ability for next day's volatility, however, if we considered a smaller interval for calculating the percent change of real GDP it might do a better job. We can see this in the graph of the data below. As observed, the R-squared barely increased as it went from 0.6016 to 0.602 and the adjusted R-squared increased from 0.6017 to 0.60018. Therefore, the real GDP was not helpful in our case to predict the next day's volatility.



```
Coefficients (Slope): [ 0.29992403  0.35044077  0.22840138 -0.10081716]
Intercept: 0.011120417371729807
R-squared: 0.6020835254211909
Adjusted R-squared: 0.6018627877318905
```

Figure: Linear Regression Model with Real GDP Data

Detailed Analysis, Discussion, and Conclusion

Despite our efforts, none of the macroeconomic variables tested improved the predictive power of the model significantly. Here are some of the potential reasons:

Complexity of Market Dynamics

Financial markets are influenced by a multitude of factors. This includes geopolitical events, investor sentiment, corporate earnings, and global economic conditions. The regression model may oversimplify these factors by attempting to be linear, leading to inaccuracy of data interpretation.

Non-Linear Relationship

As mentioned before, the relationship between macroeconomic variables and stock market volatility may not be linear. This would go against the assumption in linear regression (of linearity), leading us to misinterpret the effects of the variables.

Lag Effects

Economic variables may exhibit lagged effects on stock market volatility. For example, changes in interest rates or inflation may take time to impact investor behavior and market dynamics. Incorporating lagged variables or dynamic modeling techniques could address this issue.

Multicollinearity

The macroeconomic variables considered in the analysis may suffer from multicollinearity, where independent variables are highly correlated with each other. Multicollinearity can distort the estimation of regression coefficients and inflate standard errors, leading to unreliable model predictions. In the presence of multicollinearity, the interpretation of individual coefficient estimates becomes challenging, as they may not accurately reflect the true relationship between independent and dependent variables.

Heterogeneity

Financial data often exhibit a behavior that is known as volatility clustering: the volatility changes over time and its degree shows a tendency to persist, i.e., there are periods of low volatility and periods where volatility is high. This leads to a non-constant variance of the error term in the regression model which is a violation of the normality assumption of the error term for linear regression models.

Imperfect Macroeconomic Variable Collection

As with much of the data from the FRED, our macroeconomic variables were all reported in monthly data. This presents problems when trying to estimate the volatility trends on a rolling basis. Our data experiences jumps which can cause the macroeconomic variables to seem as improper and unsuitable.

Implications and Future Directions

While our initial findings suggest that the selected macroeconomic variables may not be effective predictors of S&P 500 volatility, several avenues for further research and improvement exist. We could explore alternative macroeconomic variables or composite indicators that may have stronger predictive power for stock market volatility. This could involve incorporating financial market data, sentiment analysis from news articles or social media, or alternative economic indicators. We could also consider using higher-frequency data to improve the realized volatility estimates. Finally, future projects may also examine how a weighted least squares model may improve the forecasting ability of next day's volatility by accounting for realized quarticity and correlated error terms.

In closing, while our initial attempt to predict S&P 500 volatility using macroeconomic variables did not yield significant improvements in model performance compared to existing methods, our work is not to go unappreciated. We can now better envision a method to create a similar project in the future, and learn from the challenges that this project instilled. By addressing the limitations outlined above and leveraging advanced analytical techniques, we can enhance our understanding of market dynamics and develop more robust forecasting models for investors and financial predictors.

Bibliography

Ceniceros, Hector D. *Introduction to Numerical Analysis*. Manuscript, 14 Sept. 2023.

Corsi, Fulvio. "A Simple Approximate Long-Memory Model of Realized Volatility." *Journal of Financial Econometrics*, vol. 7, no. 2, 2009, pp. 174–196. https://statmath.wu.ac.at/~hauser/LVs/FinEtricsQF/References/Corsi2009JFinEtrics_LMmodelRealizedVola.pdf.

“Federal Reserve Economic Data.” *FRED*, Federal Reserve Bank of St. Louis, Mar. 2024, fred.stlouisfed.org/. Accessed 17 Mar. 2024.

Molnár, Peter. “Properties of Range-Based Volatility Estimators.” *International Review of Financial Analysis*, vol. 23, June 2012, pp. 1–10, doi:10.1016/j.irfa.2011.06.012.