Predicting S&P 500 Recession Trends with Machine Learning

Harvard CS109A - Introduction to Data Science
Tony Hua, Faisal Karim, Maggie Mano, Oscar Mercado

Abstract

In 2022, the United States is encountering yet another recession sparked by the global Covid-19 pandemic. However, this is not a new phenomenon - historical U.S. economic data has shown that economic markets are cyclical, and trends can be observed on a macroscale. With today's access to data, applying machine learning models gives us the opportunity to potentially predict market trends. The implications and the models can be utilized by business leaders, retail investors, and more. In our study, we will apply machine learning techniques to predict S&P 500 prices surrounding recession events using a wide variety of publicly available economic indicator data.

Introduction

Background

The S&P 500, short for Standard & Poor's 500, is a stock market index to track the value of 500 of the largest U.S. companies that are representative of the U.S. Economy. The S&P market index is widely thought of as a bellwether representation of the stock market, and an indicator of the direction of the economy. Several notable recessions and bull runs have been preserved in S&P 500's price history: the tech bubble collapse of the 2000s, the 2008 financial crisis and Great Recession, and now currently, the recession and market uncertainty sparked by the 2020 coronavirus pandemic.[1]

Motivation

Given the current economic uncertainty as our context, the application of machine learning to predict the future price action of the U.S. economy is a timely problem to solve, with large implications for policy making, market participation, business strategy, social well-being, retirement funds, and more.

Research proposal

This project aims to build a simple and robust model to predict recession trends proxied by performance of the S&P 500 price based on various economic data. The project makes use of the following data: consumer price index, industrial production index, unemployment rate, building permits, mortgage rates, median home value, credit discount rate, commercial and industrial loan rates, federal monetary fund rate, household savings rate, and disposable personal income, to predict future S&P 500 prices.

Literature review

Global Investment firm, Guggenheim Partners, forecasted a recession in the next 24 months along with a predictive recession dashboard of six economic indicators that exhibit consistent cyclical behavior of which they used to predict the recession we are now experiencing.[2] Guggenheim's predictions proved to be accurate and shows that two assumptions are likely: 1) that U.S. recessions exhibit markers or early warning signs, and 2) future recessions will be similar to historical recessions. Individually, economic indicators may provide limited informational value but combined and studied together, machine learning may reveal trends and patterns that provide more useful informational value.[3] In another effort, Ligita Faspareniene and her team used a dataset of 27 economic indicators and followed a 4 stage process to model S&P 500's price action: data sourcing, exploratory data analysis of the dependent variable, model

[1] (Investopedia, 2022)
[2] (Forecasting the Next Recession: How Severe Will the Next Recession Be?, 2019)
[3] (Wang, 2019)

fitting and tuning, and finally model performance evaluation[4]. Other prior efforts and techniques we studied included: time series analysis with the ARIMA model[5], time series models ARCH and GARCH[6], and critiques on the reliability of economic prediction factors for world markets by Nidal Rashid Sabri.[7]

From these papers, we can take away methods such as regression and time series modeling and a multitude of indicators to build our predictor model. Combining the techniques and lessons learned from prior recession prediction efforts, we proceeded forward to model our prediction.

Methodology

Feature selection

We rely on the price of the S&P 500 market index as a proxy for studying stock market price action, so the price of S&P 500 will be our model's response variable. We selected 12 economic indicators for our analysis, and after research we selected indicators that accounted for a broad overview of the U.S. market. By broadening the areas where we gather data from, we can account for more factors and discover correlations that we may not have known prior to the study.

Sourcing our data

We sourced our data from publicly available repositories such as Yahoo Finance, FRED.org, the U.S. Bureau of Labor Statistics, etc. Some considerations we took were the time frame, the frequency, the unit, the scales, and the credibility of the data. For our specific project, we limited the scope of predictor data to the years 2000 until present, giving us coverage of large recession events: the tech bubble collapse - 2000s, the financial crisis - 2008, the first correction due to the coronavirus pandemic - 2020.

Data Preprocessing

The economic data is released and sampled at different frequencies. Therefore, we time-matched our data points and reorganized the varying timestamps into a single monthly cadence, starting from the year 2000. To ensure uniformity for our data we set all the variables, which in this case are quantitative, to floats with two decimal points, regardless of their unit. The raw data we collected contained variables that were on different scales, so we set all the value and currency variables to be in thousands.

Exploratory Data Analysis

Before building our learning models, we apply an exploratory data analysis step to understand the data we are working with and preview any obvious and immediate trends present. From this we observe which indicators have an eye-catching trend over time, so that we can see which predictor variables prove most effective in predicting the S&P 500 stock price.

We cleaned our data by mean imputation from the prior twelve months for missing values, standardized the variables which were on many different scales, aligned timestamps - set a timeframe from January 2000 to October 2022. We then compared the trend in our predictor variables over time.

We split up the indicators into groups determined by their units, to observe trends more easily, as follows:

1. Currency Indicators: Median Home Value, Real Disposable Personal Income: Per Capita, Commercial and Industrial Loans
2. Integer Value Indicators: Private Housing Permits, Total Vehicle Sales
3. Percentage Rates: 30-year Fixed Mortgage Rate, Discount Rate, Personal Savings Rate, Seasonally Adjusted Unemployment Rate, Federal Funds Effective Rate
4. Percentage Indexes: Industrial Production Index, CPI for All Urban Consumers

[4] (Gaspareniene¹ et al., 2021)
[5] (Stock Market Forecasting Using Time Series Analysis With ARIMA Model, 2021)
[6] (Kumar, 2020)
[7] (Sabri, 2021)

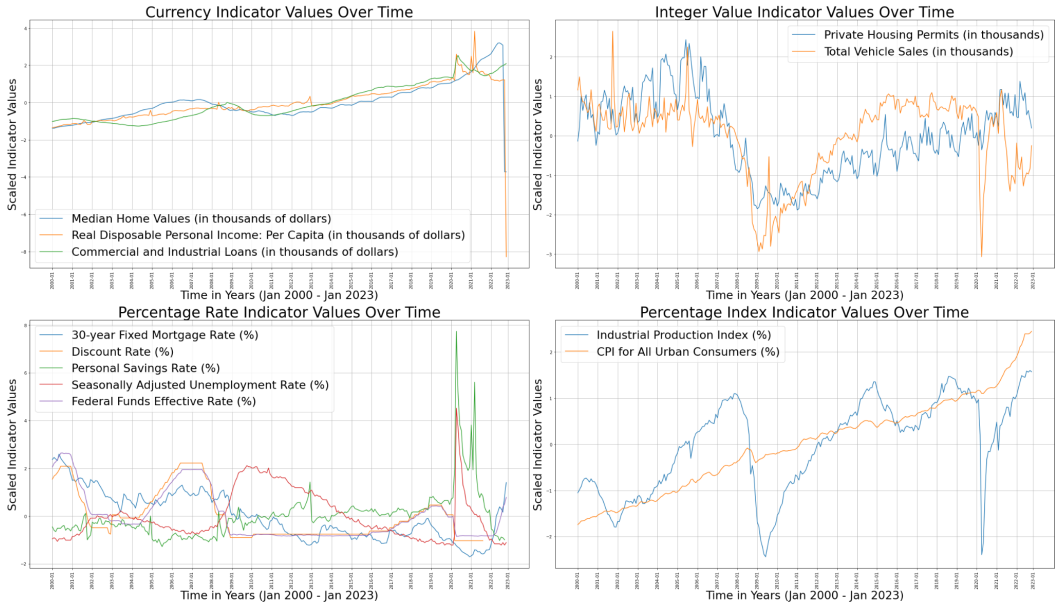

Figure 1. Categorized indicators over time

From Figure 1, we observe that the currency indicator variables follow similar trends with a dip in the years 2007-2010. This could be explained by the Great Recession of 2008, and we see a similar observation with the private housing permits, vehicle sales and unemployment rate. The personal savings rate dropped back down recently which could be because the economy is currently facing another recession. CPI does not seem to be as severely affected by recession periods compared to other variables.

Relationship between S&P 500 Price and predictors

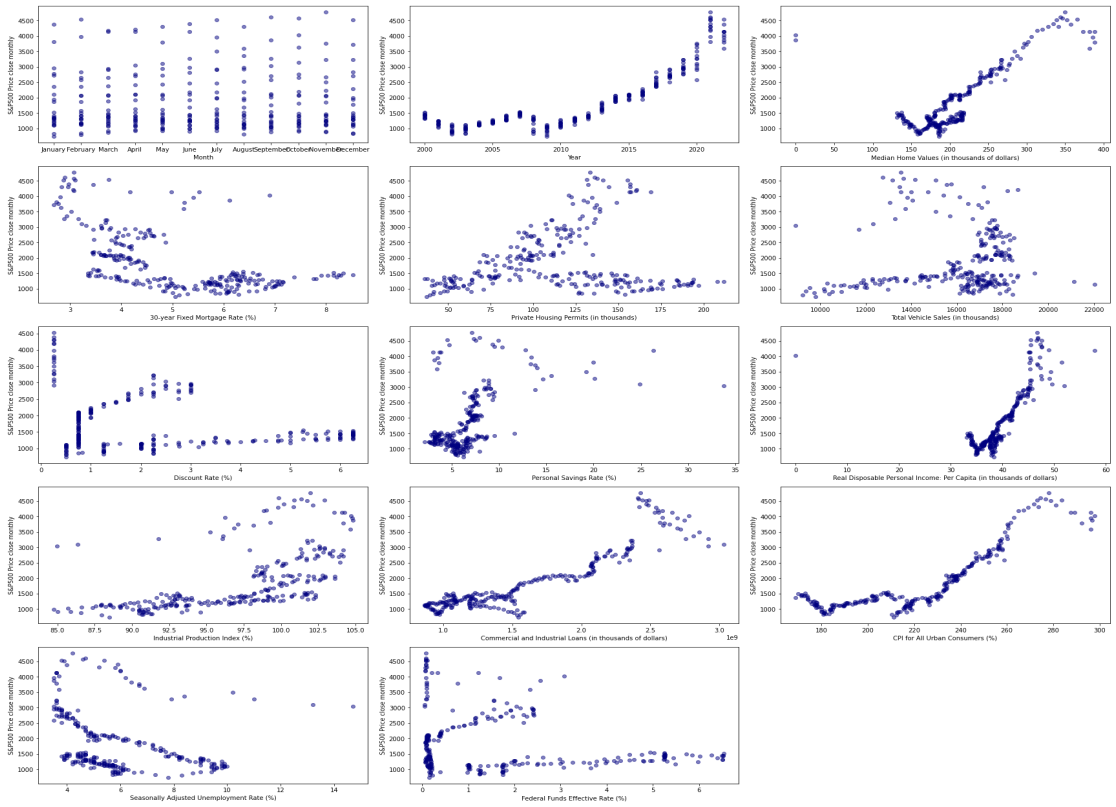

Figure 2. The Relationship between S&P 500 Prices

Figure 2. is a representation of our predictors against our response variable. The relationship between inflation (as measured using CPI) and stock performance appears to be best approximated using polynomial regression because the rate of change varies. It appears that we reach a maximum around 280 CPI, after which increasing inflation further hinders economic performance. Rising inflation yields record profits for certain sectors of the economy, boosting the overall stock market, but it remains questionable why there is this maximum around 280 CPI. Additionally, most predictors appear to have a linear relationship with stock performance; the strength of this linear relationship varies by predictor. Our graphs show that most of our predictors are correlated with stock performance.

We then observed how the response variable has changed over time, which can be seen in Figure 3. This exhibits the possible effects of the recessions as there is a dip in the S&P Price from 2008 to 2009 as well as from 2021 to present day. These are important observations because they help us understand the various factors that come into play that may affect the S&P Price which is what we are trying to predict

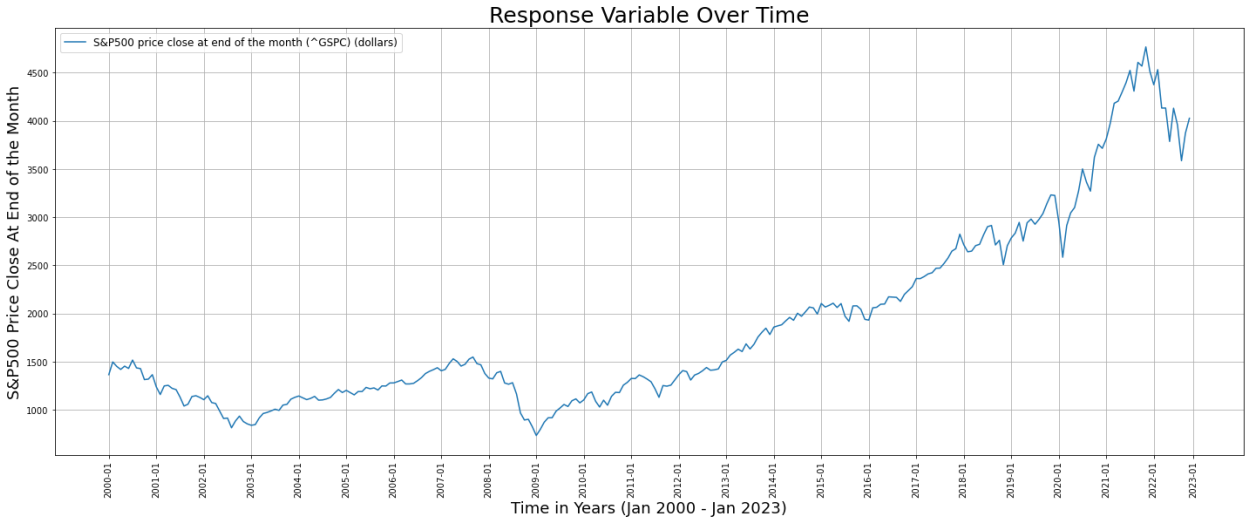

Figure 3. S&P 500 Price over time

Baseline time series prediction model

To come up with a time series model that would accurately predict the S&P 500 closing price, a baseline model was needed to compare with. Specifically, to determine the performance of the final time series model, a simple AR(1) model was fitted on the S&P 500 closing price time series process. We decided to use this model to see if we could predict from the target variable itself before looking at other predictor variables. Since the target variable is a time series, it made sense to use a time series model, and since AR(1) is the simplest time series model to predict future data points, we decided to go with that and use it to compare to the other models used in this project.

By looking at Figure 3, we make an initial analysis that the time series is *not* stationary. This is due to the upward trending nature of the time series. Since the expected value of the time series is changing, in this case increasing over time, we cannot claim that this time series is stationary. Furthermore, an Augmented Dickey-Fuller (ADF) test conducted on the time series showed the p-value to be 0.998160; in other words, we must reject the null hypothesis, that there are no unit roots in the time series, of the ADF test and accept the alternate hypothesis. Since we reject the null hypothesis in favor of the alternative, we have enough statistical evidence to show that the time series is not stationary.

In the interest of seeing if it is possible to obtain a stationary time series from the dataset, we can construct a time series from the differences between each data point in the original time series. In fact, in Figure 4, when taking the first difference, we construct a time series that seems somewhat stationary. Here we can see that the time series seems to have a constant expected value somewhere between 0 and 100.

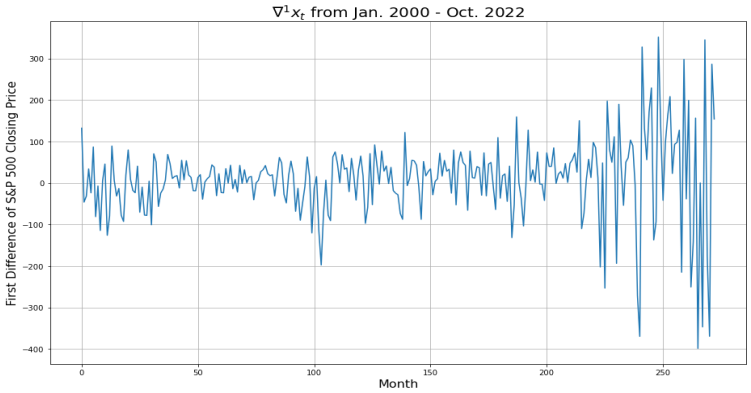

Figure 4. Time series after taking first difference

However, we can see that the variance is not constant over time, thus, the first difference time series is *not* stationary. While the ADF test gives a p-value of 0.000047, the test fails to consider the variance of the time series, and thus can no longer be used.

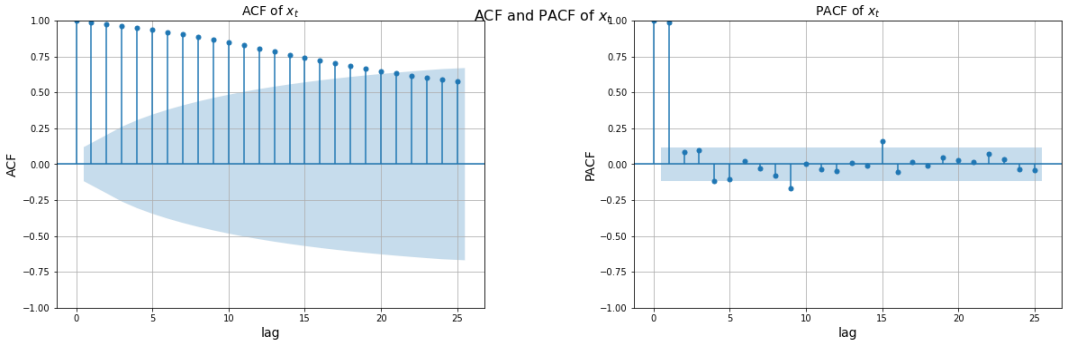

Figure 5. ACF and PACF Plots

Continuing with the construction of the baseline model, the reason for choosing AR(1) instead of another model was due to the ACF and PACF plots of the original S&P 500 price time series in Figure 5. The ACF plot exhibits the characteristic of an AR process, in which there is a geometric decay rather than a cut off after a specific lag. Since the PACF has a cut off after lag 1, we can assume that the time series follows an AR(1) process. Therefore, an ARIMA model of order (1, 0, 0) - an AR(1) model - was constructed and fit to a subset of the time series. For the purposes of determining performance, the last ten data points were removed from the time series to construct the subset.

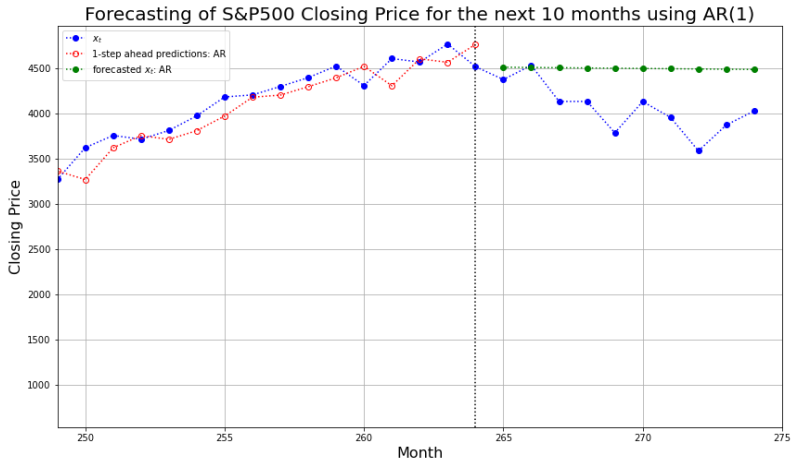

Figure 6. Forecasting the S&P 500 closing price for the next 10 months using an AR(1) time series model.

After the fitting of the AR(1) model, the model was tested against the ten data points that were removed using mean squared error. The graph in Figure 6 displays the last few points of the time series, along with 1 step ahead predictions of the model on the subset of the time series and the ten predicted values. The **MSE** of the predicted and actual ten data points was calculated to be **263,452.63** - to lower this value, higher orders of ARIMA would be needed along with a GARCH component to take into consideration the changing expected value and variance.

Final Time Series Model

Figure 7 displays the forecasts of the S&P 500 Closing Price of the final time series model of ARIMA(2, 1, 2)-GARCH(2, 2).

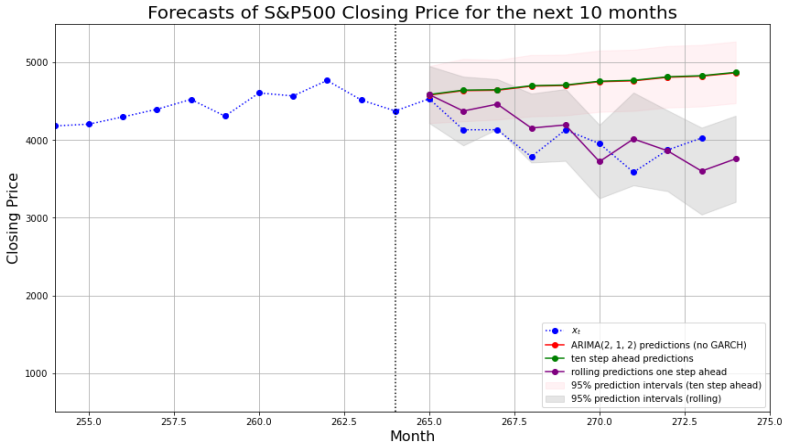

Figure 7. Forecasting S&P 500 Closing Price for the next 10 months using ARIMA(2, 1, 2) -GARCH(2, 2)

To construct the model, the residuals of an ARIMA(2, 1, 2) model fitted on the original time series were fitted into a standalone GARCH(2, 2) model. After construction of the ARIMA model, the ten test data points were immediately predicted to compare with the predictions made by the standalone GARCH model. To evaluate the ARIMA-GARCH model performance, two different methods were used - a ten-step ahead predictions calculation (where the model predicted 10 steps ahead directly), and a rolling predictions calculation (where the model was able to see each of the 10 data points one by one and predicted the next point as it saw each data point). Finally, the prediction intervals for the two different methods were also calculated.

For the ARIMA(2, 1, 2) model predictions, an **MSE of 559454.39** was calculated. While this was extremely high compared to the baseline model of AR(1), this model does not include the GARCH component. Therefore, with the inclusion of the GARCH component, for ten-step ahead predictions an **MSE of 567989.47** was obtained, while for the rolling predictions an **MSE of 73665.08** was obtained. These results indicate that the ARIMA-GARCH model only works when predicting one step ahead - that is, predicting two months or more into the future causes the model to perform worse than predicting only one month in advance. This makes sense - the farther into the future a model tries to predict, the more uncertainty there is. As we can see in the graph, the predictions for ten-step ahead mostly overlap with the predictions from the standalone ARIMA model due to this uncertainty, whereas the predictions for rolling one step ahead are like the actual values.

Regression Models

Alongside our time series model, we wanted to include a regression model that made use of all twelve of our economic indicators to predict future S&P 500 Prices. We decided to implement expansions of Linear Regression models as well as a Random Forest Regression model.

MAPE: *The Mean Absolute Percentage Error (MAPE) is a loss function that defines the error of a given model. The MAPE is calculated by finding the absolute difference between the actual and predicted values, divided by the actual value. These ratios are added for all values and the mean is taken. Overall, it is a robust way of forecasting accuracy scores in regression models. Since MAPE is a measure of error, we translated this to accuracy percentages by subtracting the MAPE from 100 for our models.*

Random Forest Regression Model

Approach: We implemented a random forest regression model to observe how an ensemble model would perform on our data. The random forest model in general is a seemingly great method for regression because it produces more accurate results, works well on large datasets, and can work with missing data by creating estimates for them.

Process: Before implementing the model, we tuned the hyperparameters for the model by evaluating various parameters: number of estimators, max_depth and random_state. We carried out this evaluation using RandomizedSearchCV which is a method that helps find optimal parameters for the implementation of a model. The parameters that optimized the random forest model were *random_state* of 2, *n_estimators* of 100 and max_depth of 8, and we fit the model on an 80% split of the training data.

Results: The Random Forest model yielded an **MSE score of 8066.1259** and a **MAPE Accuracy of 96.33%**. These metrics when compared to other models are seemingly optimal. However, because Random Forest models cannot extrapolate outside unseen data, the predicted values are never outside the training values. Because extrapolation is important in our project due to the data being in time series form, we found that using a Random Forest regressor in this case is not helpful in identifying meaningful trends in the S&P 500 Price.

Linear Regression Model

Approach: With the fact that we required a model that can extrapolate past the range of training data, we implemented a simple linear regression model to help a more accurate, predictive model. Linear regression was used to determine the strength of the association between the S&P 500 Price and the twelve independent variables. From this, we could analyze whether there was a linear relationship between our predictors and the response variable and evaluate whether we needed to enhance our regression model further.

Process: Using our correlation table, we reasoned that our best model should include all predictors. Hence, we fit the model on an 80% train split of the data. We further analyzed the appropriateness of the linear regression model using residual plots which confirmed that the model was appropriate since residuals were randomly distributed.

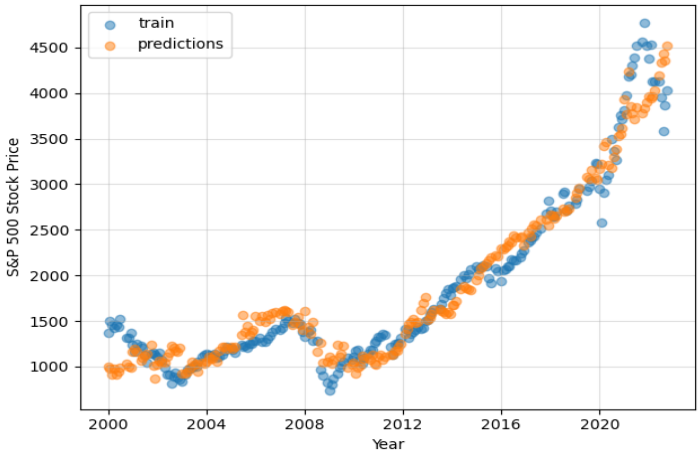

Figure 8: Linear Regression Predictions Trend

Results: From Figure 8, we can see that the relationship between stock price and our predictions is not strictly linear, so we can improve our model by introducing complexity, adding polynomial terms to our regression model. Also, the accuracy metrics left something to be desired from linear regression: the Linear Regression model yielded an **MSE score of 111554.77** and an **MAPE Accuracy of 88.33%**. The key limitation of linear regression is its assumption that the relationship between the response variable and explanatory variables is strictly linear. Thus, the model cannot adapt to data that may have general trends, but are not strictly linear, which is an incredibly stringent assumption.

Polynomial Regression Model

Approach: We expanded on linear regression through polynomial regression due to the underwhelming performance of the linear regression model.

Process: Before implementing the model, we tuned the hyperparameters for polynomial regression, being the degree. First, we used single validation to find the best polynomial degree. This parameter varied greatly depending on the random state of our train-test split. Therefore, we also used cross-validation to find the best degree regardless of our train-test split and considering multiple splits. This yielded more consistent results with polynomial degree 2 being the best performance. However, we were limited to only 5 folds in cross validation due to its high computing power and slow runtime. Similarly, we were limited to a max degree of 10. Finally, we fit the model on the same 80% train split of the data.

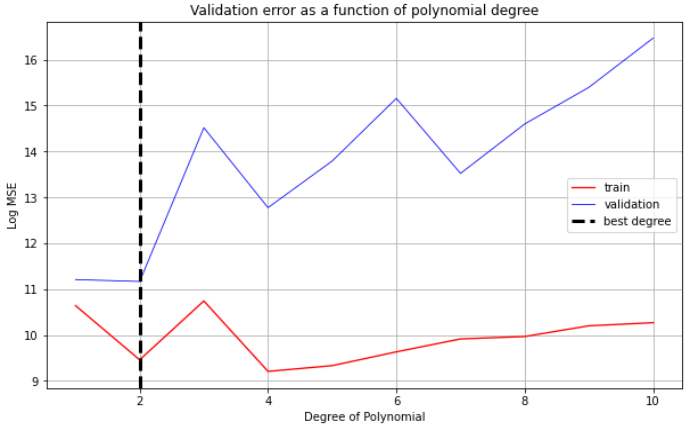

Figure 9: MSE Loss as a function of polynomial degree

Results: From Figure 9, we can see that the relationship between MSE and degree of our polynomial regression. The Polynomial Regression model yielded an **MSE score of 21628.59** and an **MAPE Accuracy of 94.48%**. Limitations of polynomial regression models are its sensitivity to outliers such as recessionary periods in this analysis, and its inclusion of correlated predictors in our model that add very little performance boost.

KNN Regression Model

Approach: We decided to then test out a KNN regression model with the reasoning that prior time stamps were indicative of future stock prices as well as timestamps with similar economic conditions. Hence, it seemed like the ripe opportunity for a KNN model which leverages neighbors for prediction.

Process: To tune the hyperparameters for KNN regression, we built several KNN models with K ranging from 1 to 100, storing the MSE after every iteration. Then, we chose the KNN model with the minimized MSE values.

Results: Figure 10, shows the range of MSE values against K values. From these results, we chose the best k-value of 4 for our KNN regression and implemented our model. The results of this model were an **MSE score of 47457.16** and an **MAPE Accuracy of 89.41%**.

Limitations of Knn regression are its tendency to overfit by including non-important features, its sensitivity to outliers like recessions, and its poor performance as the dimensionality of the data increases.

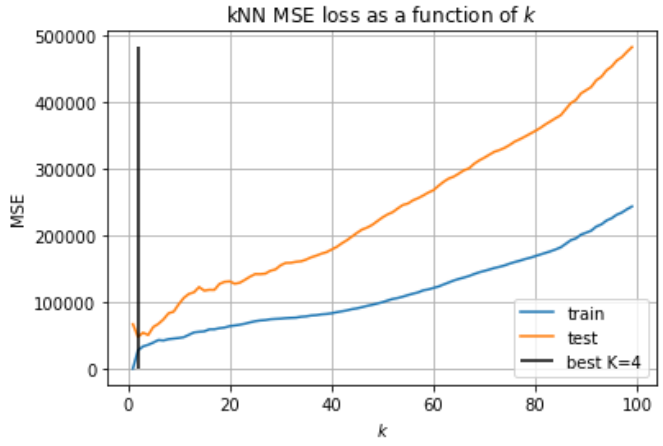

Figure 10: KNN MSE loss as function of k

Final Regression Model

Finally, we implemented a Lasso regression model with the reasoning that some of the predictors were not particularly strong enough when isolated to predict stock prices. As is the case with economic predictors, they tend to be highly correlated with each other. Hence, Lasso regression works to reduce the multicollinearity that exists in our data by penalizing our predictors.

To tune the hyperparameters for Lasso regression, we used Lasso cross validation with 5 folds to find our best alpha which ended up as 0.1. Then, we used our best polynomial degree we previously found with cross validation and fit our Lasso regression model with these hyperparameters.

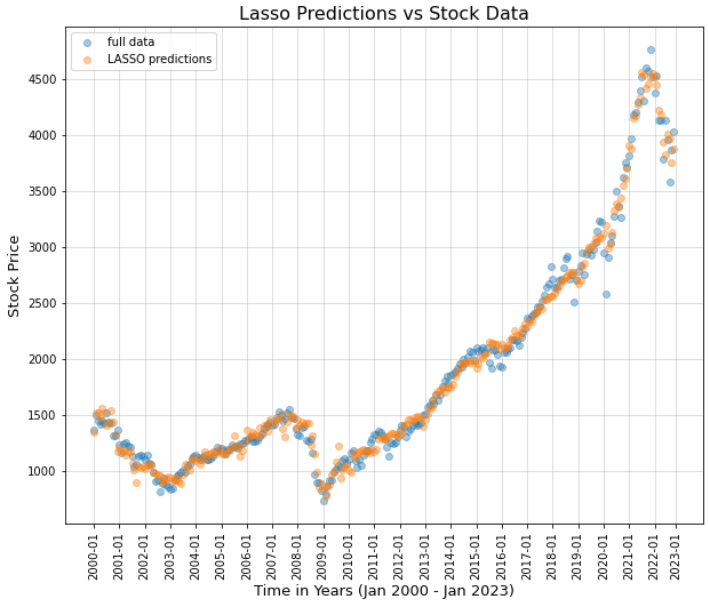

Figure 11. Lasso Regression model predictions

The results of this model were a **95.12% MAPE Accuracy** and an **MSE of 19009.06** on the test set. Looking at Figure 11, we can clearly see that our model performed better when predicting on the entire stock data. However, it is important to note that the key limitation of the Lasso model is the coefficients generated by the model are inherently biased since we penalize correlated predictors. Hence, we reduce the complexity of our model, thus increasing the bias.

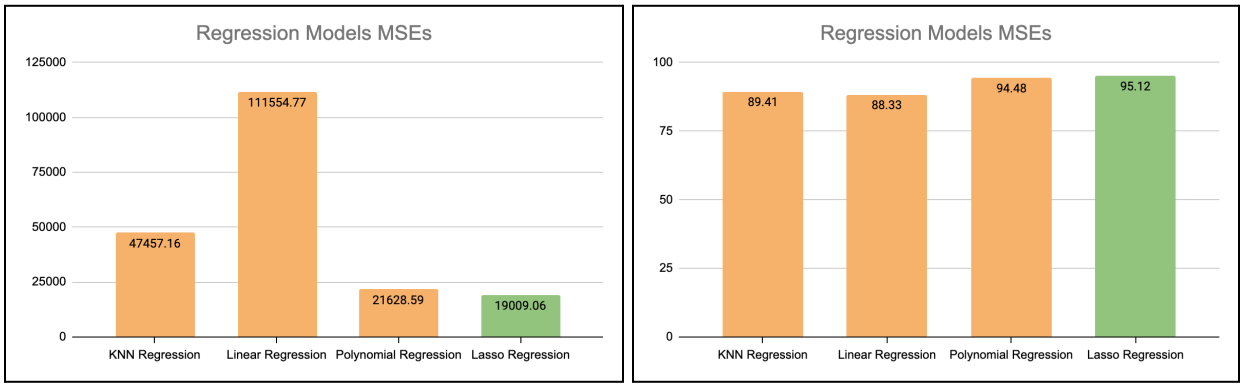

Figure 12. Comparing Regression Models' MSEs and MAPEs

Figure 12 shows a comparison of all our regression models, without the Random Forest model for reasons previously stated. The Lasso Regression yielded the best MAPE Accuracy of 95.12% and the lowest MSE of 19009.06 of all the regression models we utilized, hence why we decided on Lasso Regression as our final regression model.

Results

Figure 13 below, shows the predicted values for twelve months that our lasso regression model has forecasted. From this figure we can see that our model has predicted steep declines in the S&P 500 stock price in February and September of next year. This could correlate to the continuation of the recession that the US economy experienced due to the COVID-19 pandemic. It is important to note that this forecast is based on twelve economic indicators that we selected and together these indicators may not serve as the optimal indicators for predicting S&P 500 prices.

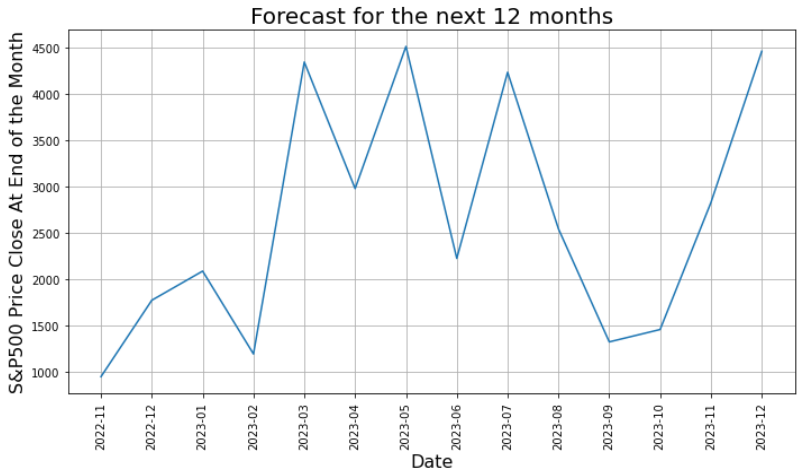

Figure 13. Lasso Regression S&P 500 price forecasts

Verifying prediction results with Stock Trading Indicators

As a sanity check, we compared the S&P 500 price prediction with stock trading technical analysis indicators such as the 200-day Simple Moving Average (SMA), the 14-day Relative Strength Index (RSI), and the Moving Average Convergence and Divergence (MACD) indicator commonly used by financial analysts. The SMA takes the average price over the past 200 periods, therefore telling investors data about whether an investment is under or overpriced.[8] The RSI measures the speed and momentum of a stock's price change, informing investors if a stock has been gaining significantly over the past period or lost

[8] (Hayes, 2022)

significantly.[9] MACD is calculated by comparing a 26 day Exponential Moving Average (EMA) line with a 12 day EMA line.[10] The difference in the long term and short term averages helps investors visualize changes in short term price action compared to longer term historical trends. Stock traders typically use the combination of the three indicators to predict stock price action.

In Figure 14, we observe the S&P 500 price forms underneath the SMA-200 curve and is being rejected by the SMA line, a signal that future price action will trend downwards with the SMA line as a resistance or psychological price ceiling for investors. The RSI chart is also near overbought territory, and MACD indicator shows a red and negative trend. Overall, the indicators imply the price action will continue to decline, therefore supporting the prediction our machine learning models forecasted.

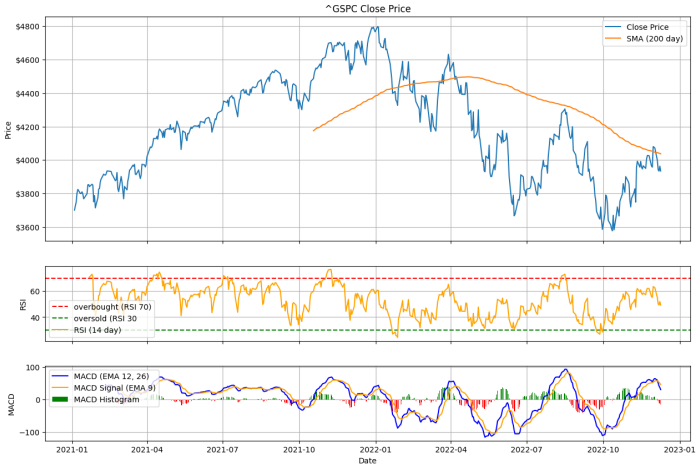

Figure 14: SMA, RSI and MACD Analysis of S&P 500 Price

Conclusions

There are a couple takeaways from our project that we can use from our results analysis. Our models implied a decline in S&P 500 price over the next couple months, thus predicted an overall decline in the US Economy. The trends we observed with the response variable correspond with recession trends and so we can make an overarching assumption that the two are highly correlated. From our earlier EDA, we observed that recessions highly impact industrial productivity, so we can expect radical strategy shifts from industry in the next year or so. From our S&P 500 price forecast, we see that in the next 12 months we will have periods of gains and periods of losses. This could have effects on economic activities in the US such as consumer spending behavior slowing down.

Our project was mainly limited to the scope of concepts we learnt in class and so there are ways in which it can still be improved upon. For example, the use of neural networks as a regression model could have been an interesting technique, especially since the economic indicators and S&P 500 Closing Price are complex datasets requiring complex models. Another example is with the time series forecasting - there are a multitude of models that are outside the scope of the class that could perform better than the models used for this project.

The implications and use cases for this project are boundless as typically anyone can benefit in preparing for a forthcoming recession by evaluating predicted S&P 500 Prices. Specifically, business leaders can use the model to effectively further strategize around downturns or growth periods; retail investors can plan optimal retirement dates based on this model; various employees can plan and time out possible career changes.

[9] (Fernando, 2022)
[10] (Dolan, 2022)

References

Dolan, B. (2022). *MACD Indicator Explained, with Formula, Examples, and Limitations*. Investopedia. https://www.investopedia.com/terms/m/macd.asp

Fernando, J. (2022, July 15). *Relative Strength Index (RSI) Indicator Explained With Formula*. Investopedia. Retrieved December 11, 2022, from https://www.investopedia.com/terms/r/rsi.asp

Forecasting the Next Recession: How Severe Will the Next Recession Be? (2019, April 9). Guggenheim. Retrieved December 1, 2022, from https://www.guggenheiminvestments.com/perspectives/macroeconomic-research/recession-update-how-severe-will-recession-be

Gaspareniene[1], L., Remeikiene, R., Sosidko3, A., & Vebraite, V. (2021). Modeling of S&P 500 Index Price Based on U.S. Economic Indicators: Machine Learning Approach. *Engineering Economics*, *32*(4), 1. https://inzeko.ktu.lt/index.php/EE/article/view/27985/15124

Hayes, A. (2022, Feb 1). *Simple Moving Average (SMA): What It Is and the Formula*. Investopedia. Retrieved December 11, 2022, from https://www.investopedia.com/terms/s/sma.asp

Kumar, R. (2020, January 14). *Time Series Model(s) — ARCH and GARCH*. Medium. https://medium.com/@ranjithkumar.rocking/time-series-model-s-arch-and-garch-2781a982b448

Mobius. (2020, December 2). *Time Series Forecasting with Yahoo Stock Price*. Kaggle. https://www.kaggle.com/datasets/arashnic/time-series-forecasting-with-yahoo-stock-price

Moser, R. (2019, Jan 14). *Predicting stock market crashes*. Towards Data Science. https://towardsdatascience.com/predicting-stock-market-crashes-with-statistical-machine-learning-techniques-and-neural-networks-bb66bc3e3ccd

Predicting S&P 500 with Time-Series Statistical Learning. (2020, May 25). Data Driven Investor. https://medium.datadriveninvestor.com/predicting-s-p-500-with-time-series-statistical-learning-8b9277e30b2a

Sabri, N. R. (2021). The Reliability of Prediction Factors, for the World Stock Markets. *Theoretical Economics Letters*, *11*(3), 462-476. https://www.scirp.org/journal/paperinformation.aspx?paperid=109659

Stock market forecasting using Time Series analysis With ARIMA model. (2021, July 18). Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/07/stock-market-forecasting-using-time-series-analysis-with-arima-model/

Wang, T. (2019, September 1). *Recession Prediction using Machine Learning*. Towards Data Science. Retrieved December 2, 2022, from https://towardsdatascience.com/recession-prediction-using-machine-learning-de6eee16ca94

Arnaud de Myttenaere et al., *Mean Absolute Percentage Error for Regression Models*, Neurocomputing, Advances in artificial neural networks, machine learning and computational intelligence, 192 (June 5, 2016): 38–48, https://doi.org/10.1016/j.neucom.2015.12.114.