Developing Models to Predict Stock Price Movement

Noah Smith, Chinmay Deshpande, Harrison Lampert

Abstract

Forecasting stock market trends is important for investors looking to increase gains. This study examines if a predictive model can predict SPY's daily price changes and provide better returns than SPY itself. Using the yfinance API [1], we collected historical stock data, including open and close prices, volume, and key technical indicators such as SMA, MACD, and RSI, derived using the ta Python library. Linear Regression was applied to model relationships between past price changes and technical indicators over a 32-day window to predict the next day's price movement. Our results demonstrated that the model effectively generated buy and sell signals based on predicted price changes. Back testing on historical data showed a cumulative return of 234.53%, outperforming SPY's 202.52% gain over the exact same period. The analysis revealed that our model successfully showed both upward and downward trends, aligning returns with intended signals. This study proves that these methods can create an accurate predictive model capable of achieving better results compared to SPY.

Key Words

Stock Market Prediction, S&P 500, SPY, Linear Regression, Technical Analysis, Technical Indicators, yfinance API, ETF Tracking.

Introduction

The stock market is constantly changing, and predicting its movements is a challenge that interests both investors and analysts alike. The ability to predict price changes accurately holds the potential for higher returns which makes it a key area of financial research. SPY, which is an ETF tracking the S&P 500 index, is a widely used investment tool due to its tracking data. This project asks: Can we build a model to predict SPY’s daily price movements and achieve better returns than SPY itself?

To achieve this, we focus on analyzing stock price movements using historical price data and technical indicators. Linear regression is particularly useful, as it helps identify market trends and the strength of those trends, allowing for more accurate predictions. Linear regression has advantages in trading because it is simple to understand and implement, assists in recognizing market trends, and can act as a dynamic support and resistance level, which offers valuable trade management insights [3]. By combining these methods our goal is to develop a model capable of predicting SPY’s daily price movements and outperforming its returns.

Methods

Data Collection and Processing

We utilized the Yahoo Finance API to download SPY's closing prices and volume data for each day from January 2010 to December 2024. Then we converted the closing price to percent change to observe the day-over-day growth instead of absolute growth. We generated technical analysis indicators with our data, from a python library. We kept the most popular indicators used by traders for consideration to add to the model that we would filter after correlation and multicollinearity analysis: volume, volume moving average, volume cash money flow, macd trend, macd signal, simple moving average fast and slow, commodity channel index, and relative strength index.

In order to give the model of influence of the previous day's price we extend the closing percent change data to contain lag_n, where n represents the closing percent change of the n days prior to the current day. Also, to prevent any look-ahead bias, we shifted all indicator values back one day so that they reflect the previous day's value. Without this our model would overfit to the training data and not be accurate in real world testing.

Selecting Indicators to use in Model

To limit model overcomplexity we filtered the indicators to those that have the best correlation with the days returns and the lag days.

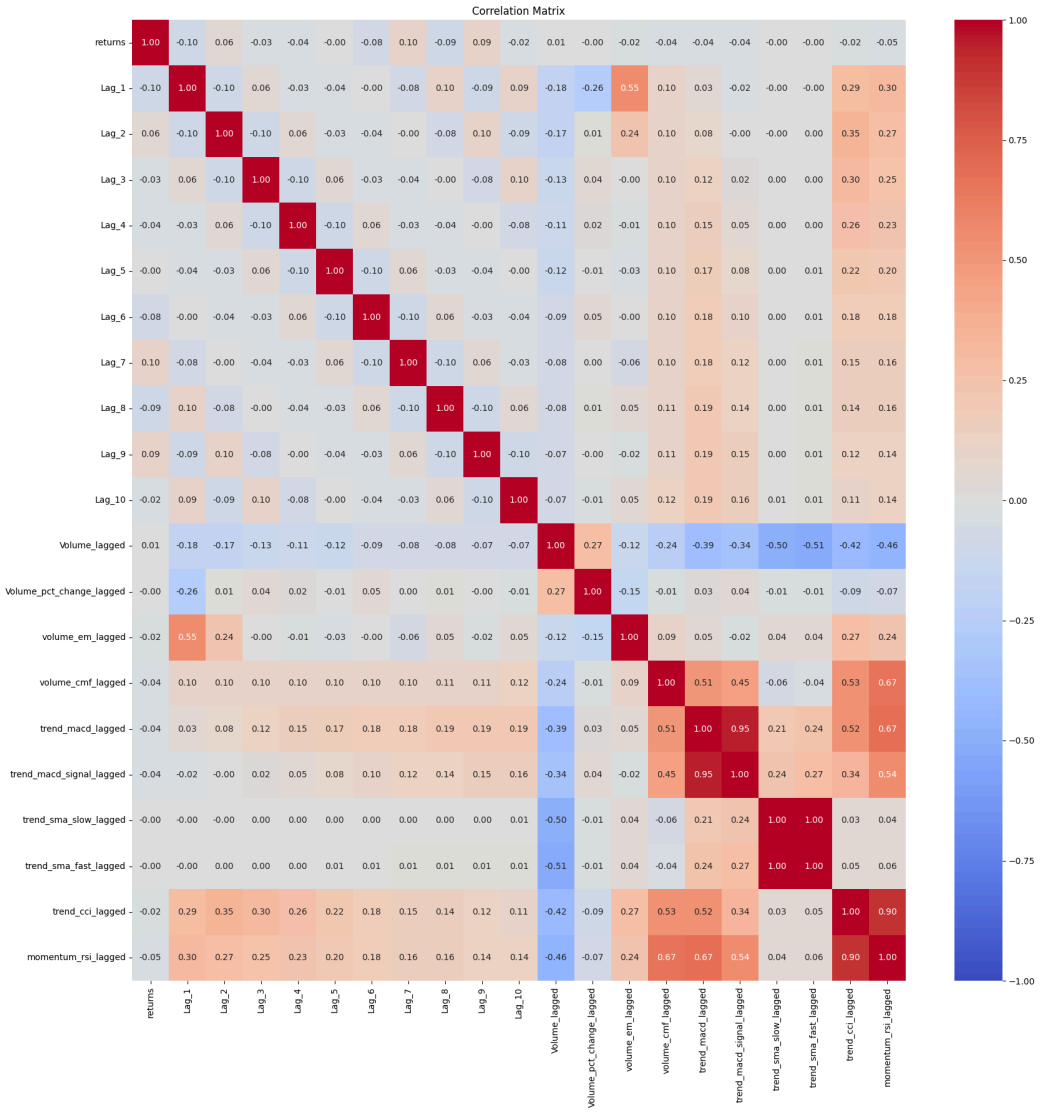

Figure 1. Correlation matrix between all considered attributes, returns, and lag days.

Figure 1 shows some correlation between the lag days and the days return, however there is no direct correlation between any indicator and days return. There is however, correlation between the indicators and lag days. These indicators are still useful as they influence the lag days coefficients with trends they represent, improving the final prediction.

Due to the fact that some of these indicators are highly correlated with each other, such as the macd trend and macd signal indicators, we needed to limit the amount of multicollinearity in the attributes. To do this we analyzed the attributes variance inflation factor (VIF) [6]. We then filtered the attributes such that a VIF < 5 is kept, after consideration of the indicators VIF effects of removing a similar indicator, like the effect removing macd trend and the resulting VIF of macd signal.

Before VIF & Correlation Filtering		**After VIF & Correlation Filtering**	
Lag_1	2.446837	Lag_1	1.457353
Lag_2	2.599059	Lag_2	1.626347
Lag_3	2.558783	Lag_3	1.439254
Lag_4	2.601739	Lag_4	1.342631
Lag_5	2.401965	Lag_5	1.270529
Lag_6	2.149451	Lag_6	1.208886
Lag_7	1.897497	Lag_7	1.181768
Lag_8	1.625566	Lag_8	1.142510
Lag_9	1.463404	Lag_9	1.142173
Lag_10	1.275769	Lag_10	1.106888
Volume_lagged	1.923848	Volume_lagged	1.325030
volume_em_lagged	1.715034		
volume_cmf_lagged	2.169277		
trend_macd_lagged	83.644099		
trend_macd_signal_lagged	43.629892	trend_macd_signal_lagged	1.363019
trend_sma_slow_lagged	7196.766309		
trend_sma_fast_lagged	7317.127672		
trend_cci_lagged	7.439491	trend_cci_lagged	3.151408
momentum_rsi_lagged	10.476959		

Figure 2. Shows the VIF values of the original considered indicator and after hand filtering.

We removed the SMA indicators due to them having no correlation with the lag days. We also decided to remove the macd trend because of its high VIF, which directly lowered the VIF of macd signal. RSI was also removed because of its high VIF which also directly lowered the CCI indicator. These two sets of two indicators are similarly calculated which explains their correlation with each other. We finally decided to use the best volume indicator to reduce complexity, filtering out volume CMF for its higher VIF and volume EM for its little correlation with all lag days. This left us with the three indicators we would pass into our model: volume, macd signal, and CCI.

Linear Regression Model Generation

The indicator values only have context as absolute values; macd signal, CCI, and volume need context to their historical values to determine the current trend, making them hard to be reduced down to a percent like the returns value. This makes the possibility of using Ridge or Lasso regression difficult, therefore we landed on using OLS linear regression. We split the data 70%-30%, training to test data, not shuffling the data such that the beginning of our testing data starts in July 2020 and ends in December 2024.

After our model has generated a prediction for the day's return, we interpret it by generating its return for the day by multiplying the direction of the prediction (1 for prediction>0, -1 prediction<=0) by the day's true return percent change.

Then to determine the best lag n value, we graphed the models return to the true returns of the market.

Results

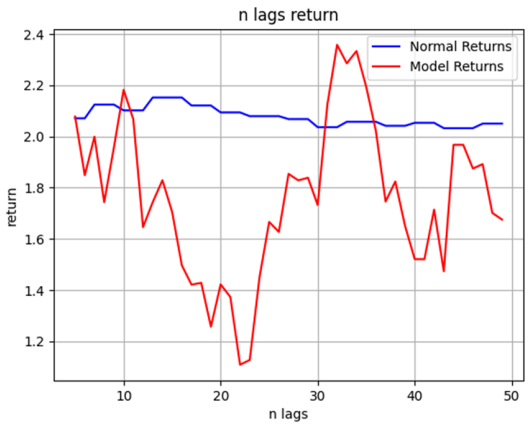

Figure 3. Comparing models return to lag n and normal returns on market.

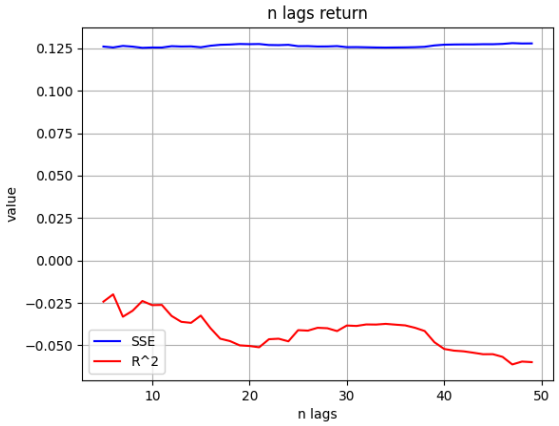

Figure 4. Displays SSE and R^2 compared to lag n value.

Figures 3 and 4 depict the model's performance to its n lag value. The model outperforms the market with 10 lags and 32 lags. The 10 lags model had a higher, but not greater than 0, R^2 with about the same SSE, but because of the market overperformance in training, we determined that 32 lags was the most optimal model with a model return of 234.53%, SSE of 12.57%, and R^2 of -0.0376.

Figure 5. shows the performance of the 32 lag day model on the testing data using the volume, macd signal, and CCI indicators.

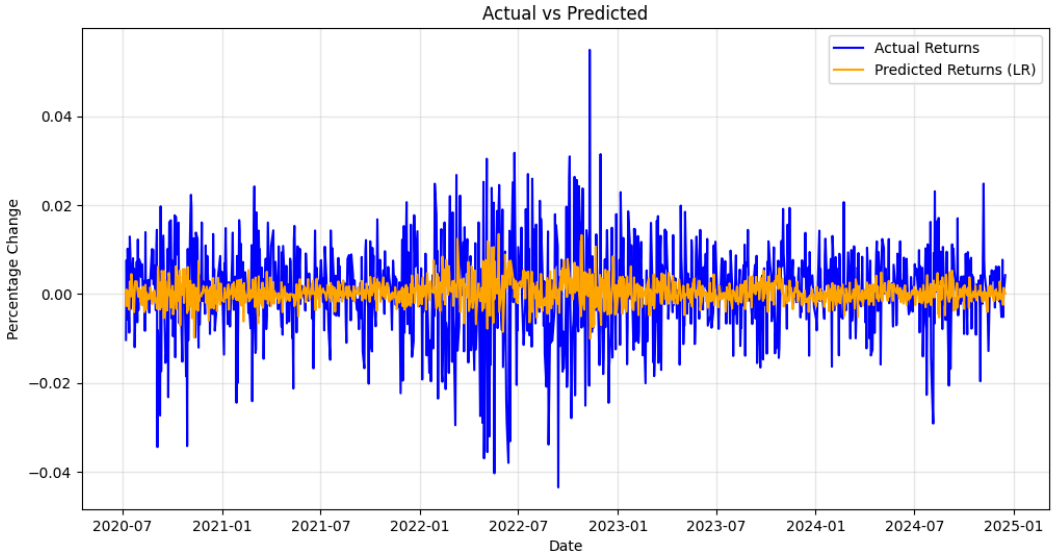

Figure 6. Displays the predicted returns compared to the days actual returns, in percent change

Figure 6 shows that the model does not accurately predict the next day's return well, which is shown in its high SSE value. Despite this, the model still performs well because the model only needs to correctly determine the day's return direction (positive or negative).

Discussion

This project has yielded valuable insights into predicting SPY's daily price movements. By using historical price data and technical indicators, the model demonstrated the ability to identify trends and generate actual buy and sell signals. Linear regression helped uncover key patterns and relationships. The model's ability to predict price movements based on past data shows potential for guiding investment strategies.

Despite its results and promise, the linear regression model has some limitations. As a lagging indicator, it relies on historical data and may not always predict future movements accurately, especially in volatile markets. The assumption of a linear relationship between price and time is restrictive because financial markets often show nonlinear behaviors. The reliance on past data and its simplistic nature can limit the model's adaptability to sudden market shifts or changes in external conditions [5]. Simply put, using this technique also does not take in account external factors that inevitably affect the stock market.

Future research could implement the use of decision trees to improve the model. Decision trees are great for capturing non linear relationships in financial data, which linear regression may not fully address. By breaking the data into subsets based on key features, decision trees can identify complex patterns and interactions that impact stock price movements [3]. Incorporating decision trees into the model could help improve predictions by improving the model's ability to adapt to varying market conditions and better handle the complexities of the stock market data.

Author Contribution Statement

Noah Smith contributed to conceptualization, analysis, model testing, administration, writing, and editing. Chinmay Deshpande contributed to conceptualization, writing and editing, model testing, and methodology. Harrison Lampert contributed to conceptualization, methodology, software, visualization, model training and testing, and writing and editing.

References

[1] Aroussi, Ran. "Yfinance: Yahoo! Finance Market Data Downloader." PyPI, 19 Nov. 2024, pypi.org/project/yfinance/. Accessed 29 Nov. 2004.
[2] James, Gareth, et al. "Linear Regression." An Introduction to Statistical Learning, 1 Jan. 2023, link.springer.com/content/pdf/10.1007/978-3-031-38747-0_3.pdf. Accessed 1 Dec. 2024.

[3] Karim, Rezaul, et al. “Stock Market Analysis Using Linear Regression and Decision Tree Regression.” ResearchGate, 1 Aug. 2021, ieeexplore.ieee.org/document/9515762. Accessed 2 Dec. 2024.
[4] Stojiljkovic, Mirko. “Linear Regression in Python.” Beoptimized, 15 Apr. 2019, www.beoptimized.be/pdf/LinearRegressioninPython.pdf. Accessed 3 Dec. 2024.
[5] Ushman, Dan. “A Comprehensive Guide to Linear Regression for Traders and Investors |.” TrendSpider, 13 Apr. 2023, trendspider.com/learning-center/a-comprehensive-guide-to-linear-regression-for-traders-and-investors/. Accessed 27 Nov. 2024.
[6] “10.7 - Detecting Multicollinearity Using Variance Inflation Factors.” *Penn State Eberly College of Science*, https://online.stat.psu.edu/stat501/lesson/10/10.7. Accessed 5 Dec. 2024.