# Regression Analysis to Predict Stock Prices and Call Option Profits:

# Tesla (NASDAQ: TSLA) Motors

Capstone Final Project

Maria Camila Asprilla Paredes

August 2, 2024

## Introduction

### Background

**The company – TSLA**

Tesla Inc. (TSLA) has rapidly established itself as a leading global player in the all-electric vehicle market, selling cars and trucks across the U.S., China, and other international markets. As a result, it has become the most valuable automaker in the world by market capitalization. The company's automotive segment, based in Austin, Texas, is the primary driver of its revenue and profitability.

**The investment**

A stock represents a small ownership share in a company. When you buy a stock, you own a piece of that company and may benefit from its success through price increases and dividends. Stocks and options are related because options are financial contracts that derive their value from underlying stocks.

According to Investopedia, an option is a financial instrument that provides the holder with the right, but not the obligation, to buy or sell an underlying asset at a predetermined price before a specified date. These contracts derive their value from underlying securities, such as common stock, and are used in finance to hedge risks and diversify portfolios, potentially leading to significant profits. While options carry high risk, the maximum loss for a buyer of a call option is limited to the initial investment in the contract. There are two main types of option contracts: puts and calls. For the purposes of this project, we will focus specifically on call options. A call option is a financial contract that grants the holder the right, but not the obligation, to purchase a specified quantity of an underlying asset at a set price before or on the expiration date of the option.

### Call options provide leverage to control more of an asset with less investment, limit losses to the premium paid, and offer potential for substantial profits if the asset's price rises. They are also cost-efficient compared to buying the asset outright and can be used to hedge or speculate on price movements.

Predicting Option Prices is essential for maximizing profitability, managing risks, and making strategic decisions. It enhances trading efficiency, optimizes portfolio performance, and provides valuable insights into market behavior. For traders and investors, precise predictions are a key component of successful financial strategies and risk management practices.

In summary, options are financial instruments that give holders the right, but not the obligation, to buy or sell an underlying asset at a set price before a specified date. Call options, in particular, offer leverage, limit losses to the initial investment, and can potentially lead to significant profits, while also aiding in risk management and strategic decision-making.

### 1.2 Objective

**Project Goal:** To utilize regression analysis techniques, incorporating financial metrics and machine learning models, to forecast the price of Tesla’s stock (TSLA). These stock price predictions will subsequently be used to estimate the prices of Tesla’s call options.

**Expected Outcome:** The project will first assess the performance of various regression models in predicting option prices to determine the most effective method. In the second phase, these predictions will be utilized to estimate the premiums (costs) of option contracts. By analyzing these estimates, we aim to identify which contracts were profitable based on the predicted data.

## Data Collection

### 2.1 Data Sources

**Yahoo Finance:** I utilized the *yfinance* library in Python to obtain both historical and real-time stock and option data. The dataset automatically updates with the most recent stock prices each time the model is executed.

**Data Dictionary**

Stocks

* **Date:** The specific date for which the data is recorded.
* **Open:** The price of the security at the beginning of the trading day (9:30 a.m. ET).
* **High:** The highest price reached by the security during the trading day
* **Low:** The lowest price reached by the security during the trading day.
* **Close:** The price of the security at the end of the trading day (4:00 p.m. ET).
* **Adj Close (Adjusted Close):** The closing price of the security after adjustments for any corporate actions such as dividends, stock splits, or other events that might affect the stock's value. This gives a more accurate reflection of the security's value over time.

**Note:** TSLA has not paid any dividends since it was first traded.

* **Volume:** The number of shares or contracts traded during the trading day. It indicates the level of activity or liquidity in the security.

Options

* **Contract Symbol:** A unique identifier for the option contract, often including the underlying asset, expiration date, strike price, and type (call or put). In the case of TSLA:

“contractSymbol” = TSLA240802C00075000

**TSLA:** The ticker symbol for the underlying asset, which is Tesla, Inc. In this case, the option is related to Tesla's stock.

**240802:** This represents the expiration date of the option in the format YYMMDD. For this symbol, "240802" translates to August 2, 2024.

**C:** Indicates the type of option. "C" stands for a call option. If it were a "P", it would indicate a put option, but in this case, we would only have call options.

**00075000:** This represents the strike price of the option. In this format, the strike price is 75.00. The last part of the symbol is typically expressed in cents, so "00075000" means a strike price of $75.00.

* **Last Trade Date:** The date on which the most recent trade of the option contract occurred.
* **Strike:** The strike price, or exercise price, is the price at which the underlying asset can be bought (for call options) or sold (for put options) when the option is exercised.
* **Last Price:** The most recent price at which the option contract was traded.
* **Bid:** The highest price a buyer is willing to pay for the option contract.
* **Ask:** The lowest price a seller is willing to accept for the option contract.
* **Change**: The difference between the last price and the previous day's closing price of the option contract.
* **Percent Change:** The percentage change in the option’s price from the previous day’s closing price.
* **Volume**: The total number of option contracts traded during a specific period, typically within a day.
* **Open Interest**: The total number of outstanding option contracts that are not yet settled or closed, indicating the liquidity and popularity of the option.
* **Implied Volatility:** A measure of the market's expectation of the underlying asset's volatility, derived from the option's price. Higher implied volatility suggests greater expected movement in the underlying asset's price.
* **In-The-Money:** Indicates whether the option is currently profitable if exercised (for a call option, the underlying asset’s price is above the strike price; for a put option, the price is below the strike price).
* **Contract Size:** The number of units of the underlying asset represented by one option contract. For most equity options, this is typically 100 shares.
* **Currency**: The currency in which the option contract is priced (e.g., USD, EUR).

### 2.2. Data Preparation

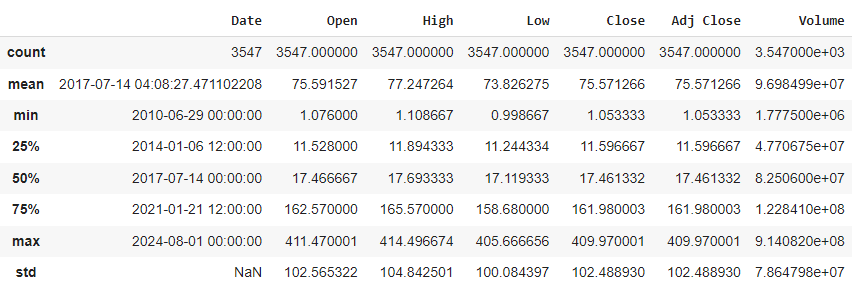
1. **Libraries used in Python:**
   * **Dataset:** yfinance
   * **Data manipulation and cleaning:** Pandas
   * **Calculations:** Math and NumPy
   * **Visualizations:** Matplotlib, Seaborn, Plotly
   * **Preprocessing:** Sklearn - MinMaxScaler, PolynomialFeatures, StandardScaler, Pipeline
   * **Model selection:** Train\_test\_split, GridSearchCV, cross\_val\_score, KFold
   * **Model:** Sklearn – Linear Regression, Ridge, Random Forest Regressor
   * **Statistics and metrics:** 
     + Sklearn – Mean squared error, mean absolute error, confusion matrix and r^2 score
     + Scipy Stats - Norm
2. **Data Cleaning:** The data obtained from Yahoo Finance is typically in CSV format, which often requires minimal initial cleaning. However, to effectively preprocess the data and conduct exploratory data analysis (EDA), several key modifications were necessary to ensure accuracy and usability
3. Changing data types by converting variables such as the date from string to “datetime” for the option contracts data frame
4. Extract the expiration date of the option contracts from the symbol
5. Dropping columns once the data was cleaned

## Exploratory Data Analysis (EDA)

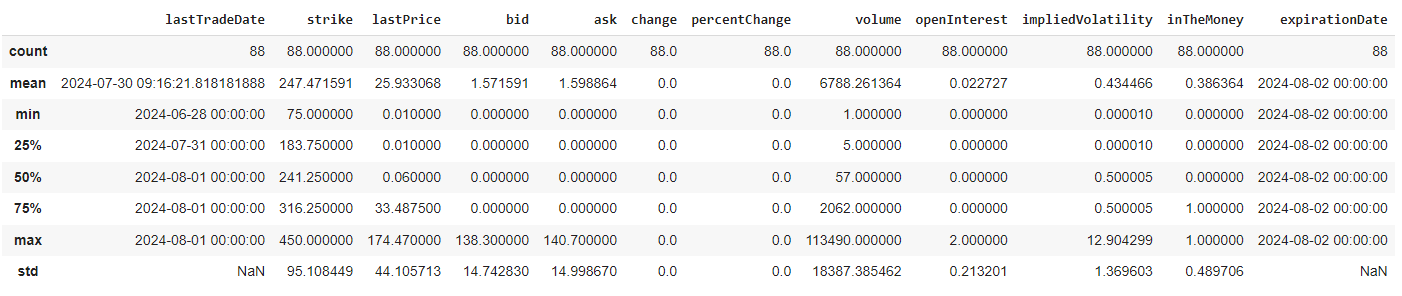
3.1. Correlation Analysis

* **Descriptive Statistics:**

**TSLA:** The average adjusted close price for TSLA, from its first trading day to August 1st, 2024, is $75.57. The highest recorded price for TSLA is $414.496, and it first traded at $0.99. There are also 3,547 records in the entire dataset.



**Call options:** Each contract should be analyzed separately as they may be purchased at different times, and the time value of money reduces their value as the expiration date approaches. The dataset contains 88 individual call option contracts, with an average implied volatility of approximately 0.43. While there isn't a single benchmark for determining if implied volatility is high or low, typically, an implied volatility of no more than 30% is considered stable for most options. Clearly, TSLA’s call contracts exceed this range, and our analysis will help visualize these metrics. The main goal of this project is to predict the price of these contracts. On average, over 6,700 contracts of all strike prices are traded daily, as indicated by the volume. Lastly, the average strike price is $247.47. Although the average stock price mentioned earlier was $75.57, the options are current contracts that only reflect current market prices.

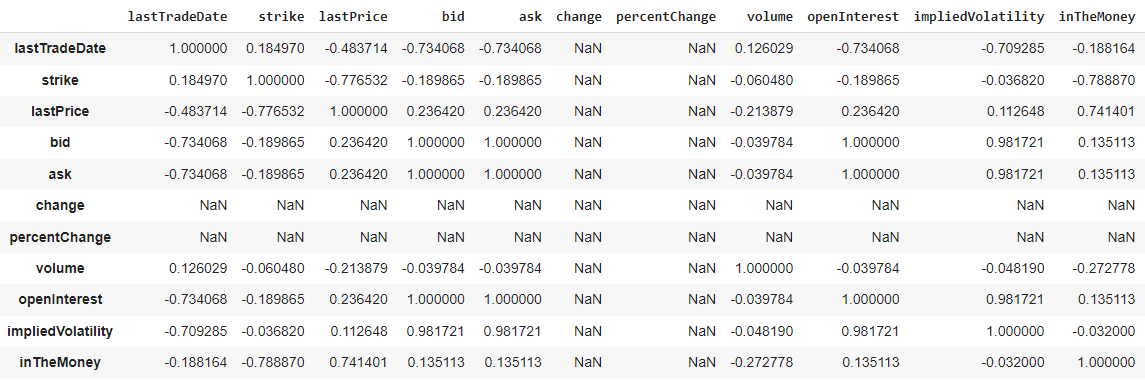


* **Correlations:**

TSLA

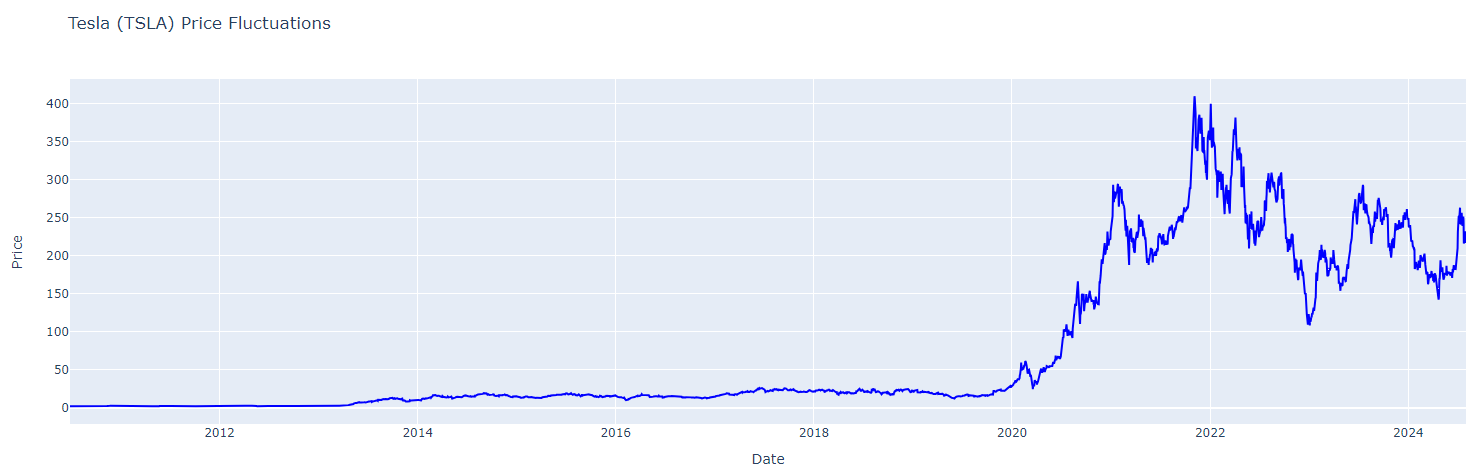


1. The dates have a strong positive correlation with the prices, but it is not as correlated to the volume of stocks traded (0.37).
2. The correlation between the volume and the other features in the dataset were not as strong, for this same reason it was removed before the models were trained to predict the prices for TSLA.
3. It is also clear that the price at which the stock closes by the end of the day is also highly correlated to how it started and the maximum and minimum prices at which it traded.

Options

* + - 1. **Strike Price vs. Option Value:** The strike price has a strong negative correlation with the last price (-0.7765), indicating that as the strike price increases, the option's last price tends to decrease. The strike price also has a strong negative correlation with the "in the money" status (-0.7889), suggesting that higher strike prices are less likely to be in the money.
      2. **Bid and Ask Prices:** The bid and ask prices are perfectly correlated (1.0000), indicating they move in unison.
      3. **Implied Volatility:** Implied volatility has a high positive correlation with bid (0.9817) and ask (0.9817) prices, indicating that as implied volatility increases, bid and ask prices also increase.
      4. **Time Sensitivity:** The negative correlation between last trade date and strike price (-0.1850) indicates that newer contracts tend to have slightly lower strike prices.

3.2. Data Visualization

Tesla’s price movements since 2010

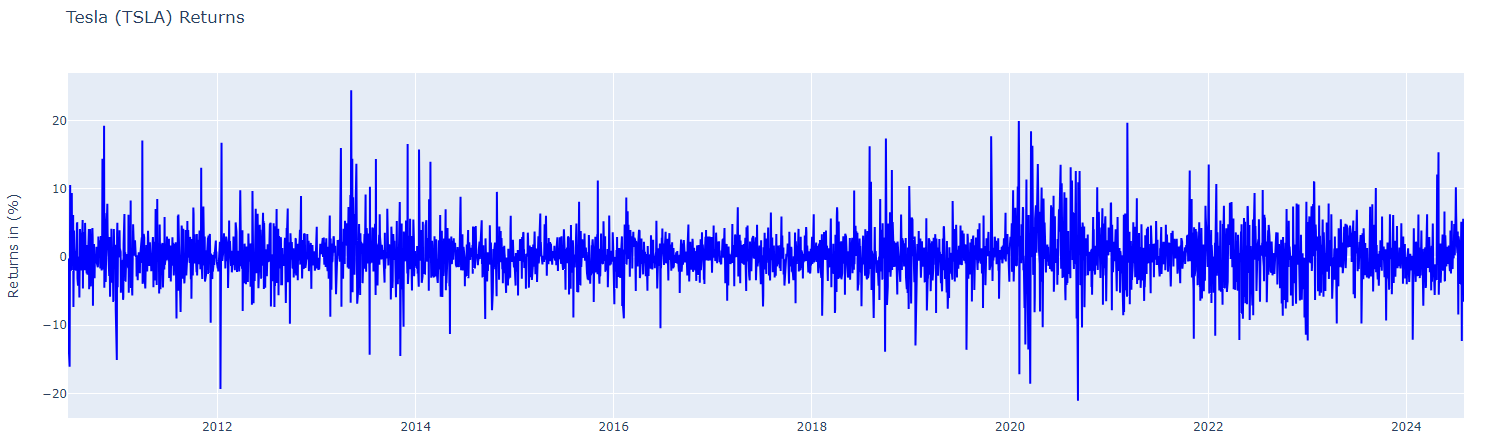
The highest price recorded was in November of 2021 as depicted by the graph above. As explained in an article by Investopedia:

“According to a recent report by investment firm Goldman Sachs, the nominal trading value of Tesla options averaged $241 billion a day in recent weeks. For context, average trading volume for the rest of the S&P 500 excluding Amazon was $112 billion. Goldman has attributed the explosion in options trading within the United States to Tesla, calling it a "critical driver of the market.”

“…Typically, options are a risky bet on an underlying equity's future price movement. The leverage risk, inherent in most options, is reflected in a stock's price volatility. While Tesla's stock is prone to wild swings, its rise has also been accompanied by a corresponding change in the perception of risk in the markets.”

They attribute the exponential growth of Tesla starting in 2020 due to climate change: “…The pandemic changed that calculus. Intensifying debate about climate change coupled with the current administration's green subsidy push and record delivery numbers (although they still lag those of established car manufacturers by a wide margin) have helped push Tesla past the $1 trillion mark. This article was written back in December of 2021. As of today, August 1, 2024, Tesla is worth $679,520,000,000.

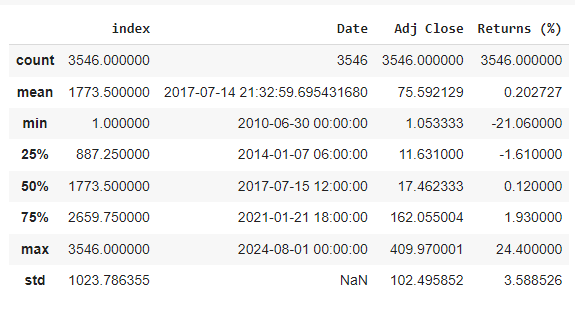
Stock price returns:



Notice how the most volatile periods for TSLA were during the pandemic up until 2022, after which the returns began to normalize and stabilize.

The descriptive statistics for the returns:

* **Return Extremes**: The returns ranged from -21.06% to 24.40%, showing the potential for both substantial gains and losses.



## Feature Engineering: Volatility Indicators

4.1 Simple and Exponential Moving Average

**Simple Moving Average (SMA)**: Widely used technical indicator that calculates the average of a selected range of prices, typically closing prices, over a specified period. To calculate the SMA, you sum up the closing prices of a stock over a certain number of days and then divide that total by the number of days. For this project I decided to use a window of 90 days and 180 days. SMA is used to smooth out price data and identify trends by filtering out the “noise” from random price fluctuations.

* If the closing prices for the last 5 days are:

$10, $12, $14, $16, and $18, the 5-day SMA would be = (10+12+14+16+18)/5 = $14.

**Exponential Moving Average (EMA):** Is a type of moving average that places a greater weight and significance on the most recent data points. EMA responds more quickly to recent price changes compared to SMA, making it more suitable for identifying short-term trends. It is favored by traders who want to capture quicker price movements and make timely decisions based on the latest market data.

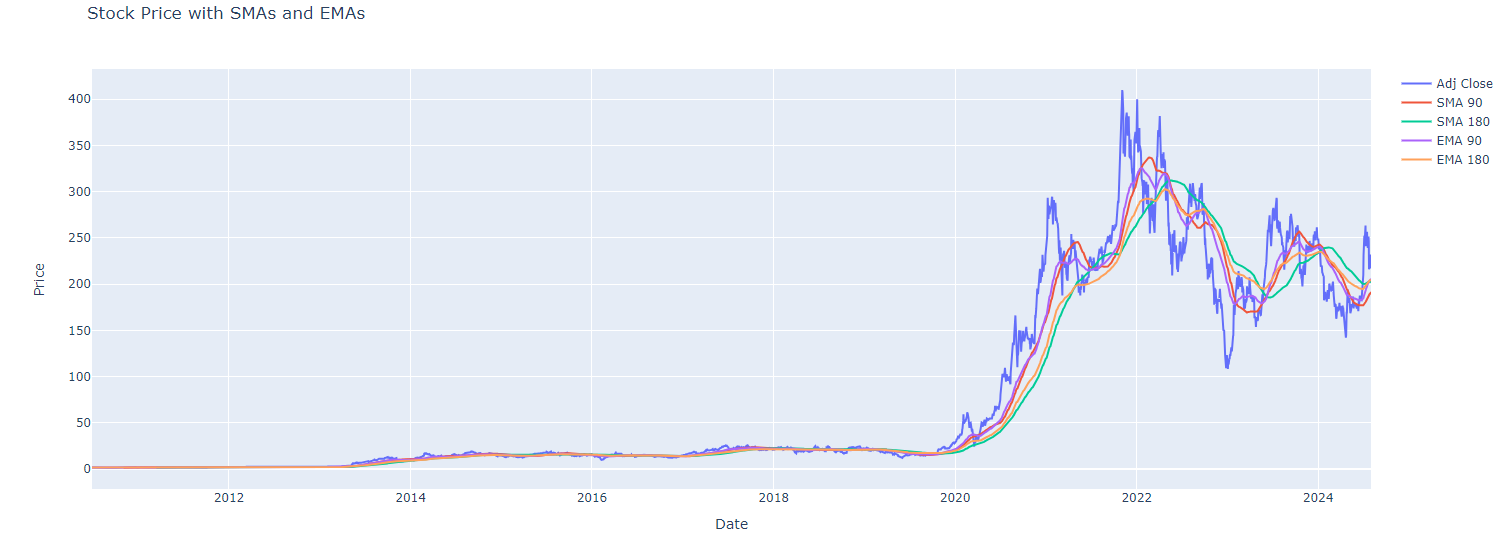
In summary,

|  |  |
| --- | --- |
| **SMA** | **EMA** |
| Is simpler to calculate and is a good indicator of overall trends, but it can be slower to respond to recent price changes. | Gives more weight to recent prices, making it more responsive to new information and better for capturing short-term movements. |

The following chart displays the calculations of the moving averages using windows of 90 and 180 days:



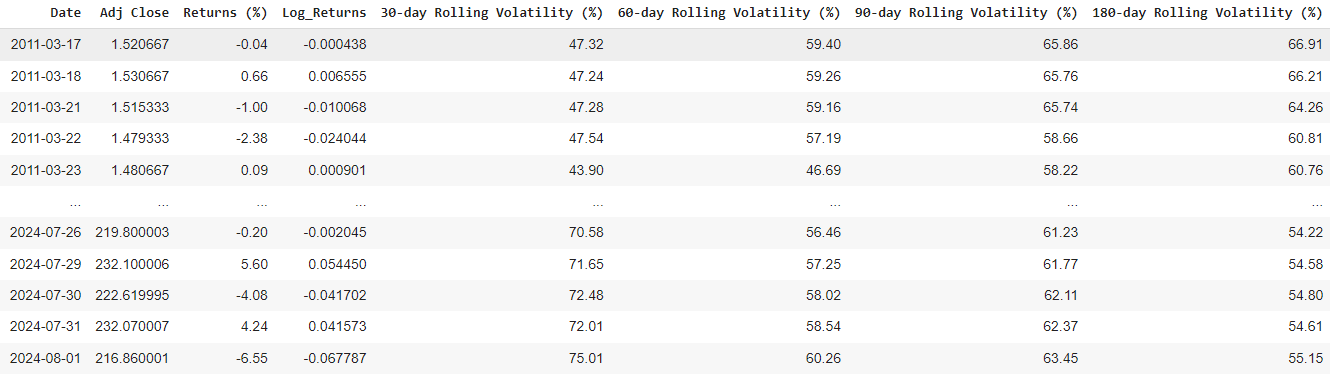
I decided to integrate moving averages into the original TSLA dataset to enhance its usability for future modeling. Notably, as the adjusted closing prices rise, the variance between the SMA and EMA and the actual price increases. This likely results from sharp price surges that the moving averages, calculated over 3 and 6-month periods, struggle to fully capture. Although a shorter window was tested, it did not significantly improve the results, so I chose to proceed with the initial approach.

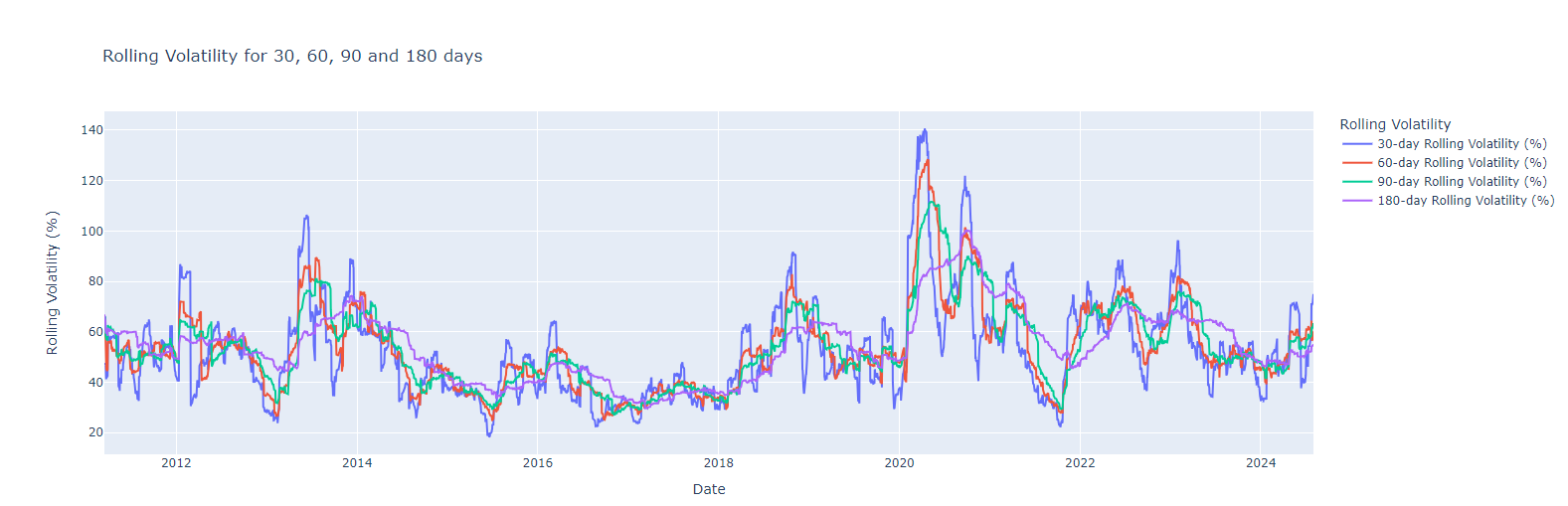


The graph above reveals that noise levels increase following sharp stock peaks starting in 2020. Despite this, the technical indicators continue to track the stock price movements effectively and begin to normalize as the stock price stabilizes.

* 1. Rolling Average in Percentage terms

A rolling average, also known as a moving average, smooths out data by averaging values over a specified period. In my example, I decided to express this metric in percentage terms for the windows of 30, 60, 90 and 180 days.



As shown in the graph below, the 30-day rolling volatility percentage provides the most accurate representation of TSLA's significant increase in 2020, effectively capturing the sharp rise during that period:

## Machine Learning Models to Predict TSLA’s Prices

* 1. Model Selection

Overview of Models: Since this project aims to predict a numerical value I needed to use a regressor. I decided to experiment with the following models:

1. Linear Regression
2. Ridge Regression
3. Random Forest Regression

**Linear Regression**

Advantages

* **Simplicity:** Easy to understand and implement. This is the most common for financial analysis.
* **Interpretability:** Coefficients provide clear insights into relationships between variables.
* **Speed:** Fast to train and predict, especially with smaller datasets.

Disadvantages

* **Linearity:** Assumes a linear relationship between predictors and target, which might not capture complex patterns in stock prices.
* **Assumptions:** Requires assumptions of homoscedasticity (the spread or dispersion of the residuals is uniform, regardless of the value of the predictor variable.) and normally distributed residuals, which may not always hold.
* **Overfitting:** Prone to underfitting if the relationship is not linear.

**Ridge Regression**

Advantages

* **Regularization:** Helps to prevent overfitting by adding a penalty to large coefficients, which can improve model performance on new data.
* **Stability:** More robust to multicollinearity compared to Linear Regression.

Disadvantages:

* **Interpretability:** Regularization makes it harder to interpret the model coefficients directly.
* **Complexity:** More complex than Linear Regression, with the need to tune the regularization parameter. This means that it may be overcomplicated for the type of data.
* **Assumptions:** Still assumes a linear relationship between predictors and the target.

**Random Forest Regressor**

Advantages:

* **Non-Linearity:** Can model complex non-linear relationships between features and target.
* **Feature Importance:** Provides insights into the importance of different features in predictions.
* **Robustness:** Generally robust to outliers and noisy data due to averaging multiple decision trees.

Disadvantages:

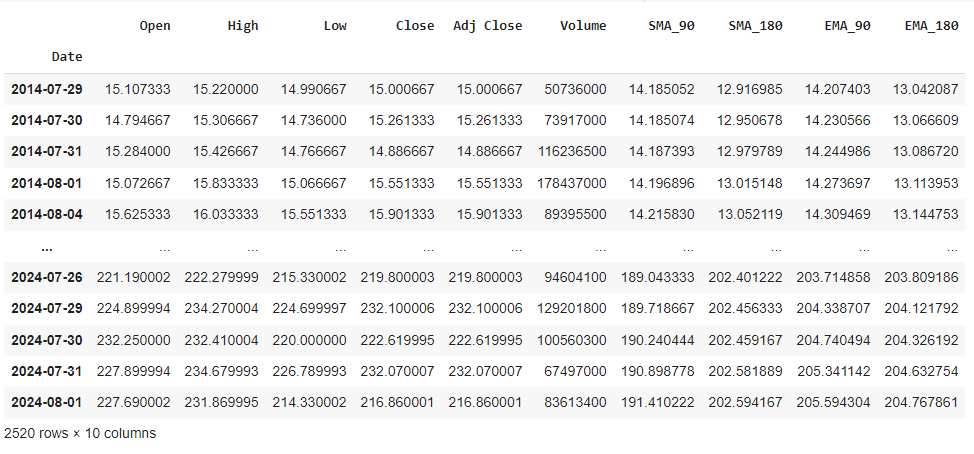
* **Complexity:** More complex and computationally intensive, requiring more resources and time for training and prediction.
* **Interpretability:** Less interpretable compared to Linear and Ridge Regression because it involves an ensemble of many trees.
* **Overfitting:** Can still occur, especially with many trees or excessive depth.
* **Hyperparameter Tuning:** Requires careful tuning of hyperparameters to achieve optimal performance.
  1. Evaluation Metrics

I used the following metrics to evaluate the three models:

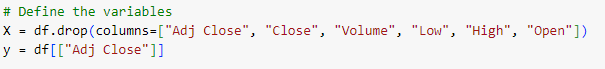
* **Mean Squared Error (MSE):** Measures the average squared difference between actual and predicted values, indicating the model's prediction error.
* **Mean Absolute Percentage Error (MAPE):** Represents the average absolute percentage difference between actual and predicted values, showing prediction accuracy in percentage terms.
* **R-squared (R²):** Indicates the proportion of variance in the dependent variable explained by the independent variables, with 1 being perfect fit and 0 being no fit.
* **Adjusted R-squared:** Adjusts R² for the number of predictors, providing a more accurate fit measure by penalizing unnecessary variables.
  1. Model Training

General information about the three models:

1. The following chart includes 10 years’ worth of data to date which was used to train all the models



1. The variables were defined as follow:

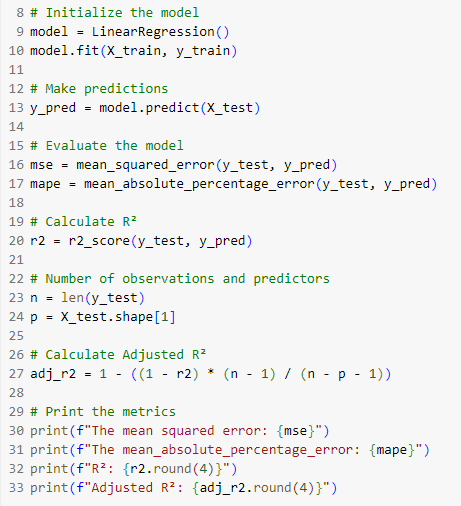


**Note:** Notice how all of the actual prices were removed to define the X variable and avoid bias. The volume was removed since it also has no significant impact in the predicted variable

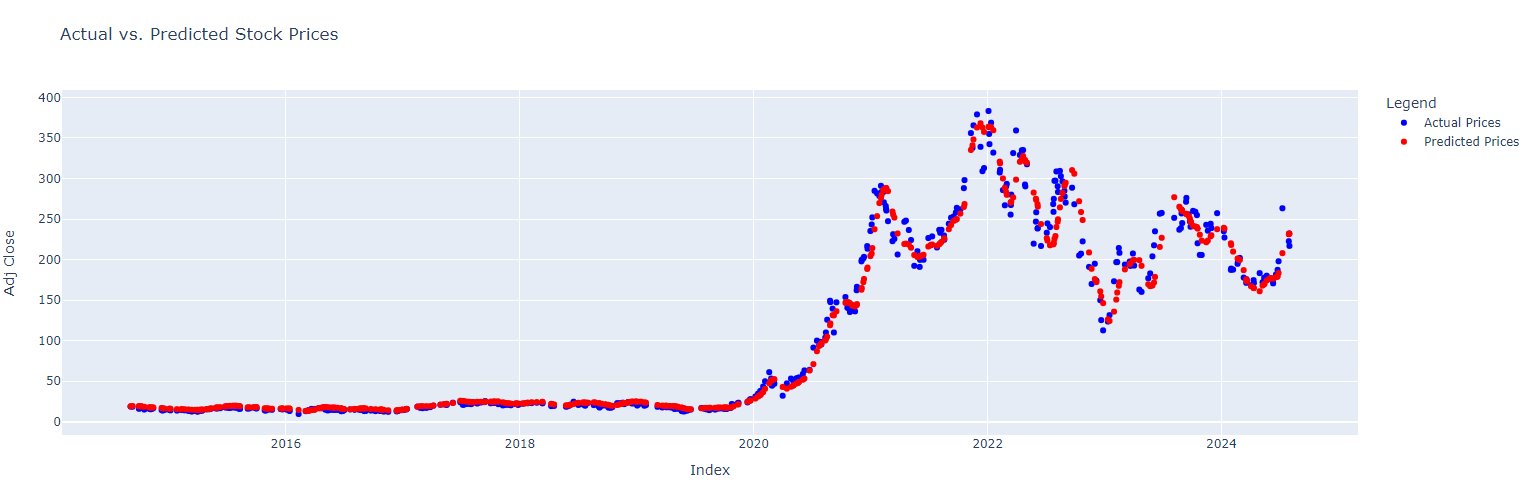
1. The models were trained using a test size of 0.2 which is equivalent to (2520 \* 0.2) 504 records.
2. The random state was 42 for all models

### Linear regression – Steps (Assuming the data was defined and split as mentioned above)

* **Initialize the model:**
  + LinearRegression() creates a linear regression model instance.
* **Fit the model:**
  + model.fit(X\_train, y\_train) trains the model on the training data.
* **Make predictions**
  + y\_pred contains predicted values based on the test set features.
* **Evaluate the model and print the following metrics:**
  + MSE
  + MAPE
  + R² and R² Adjusted

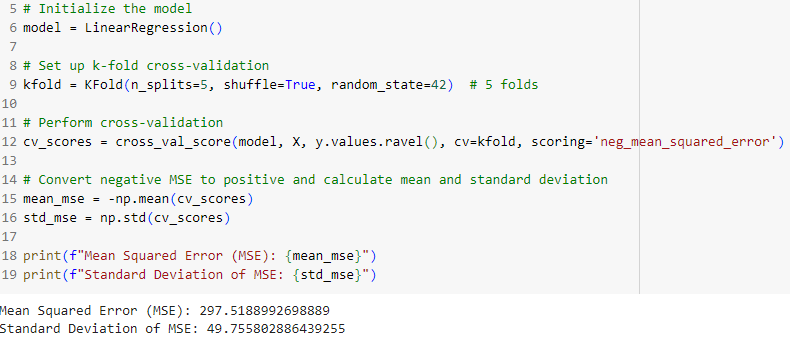


From the graph below, we can see that the linear regression results suggest a tendency to overfit the data during periods of low prices, which aligns with the earlier predictions from the SMA and EMA. However, starting in 2020, there is a noticeable increase in prediction noise, as anticipated. See the green line:



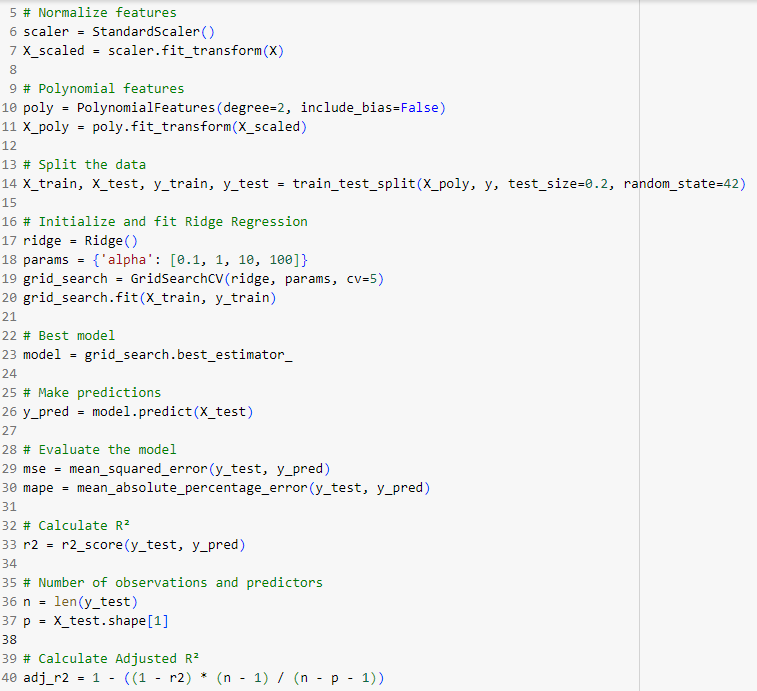
**Hyperparameter Tuning:**

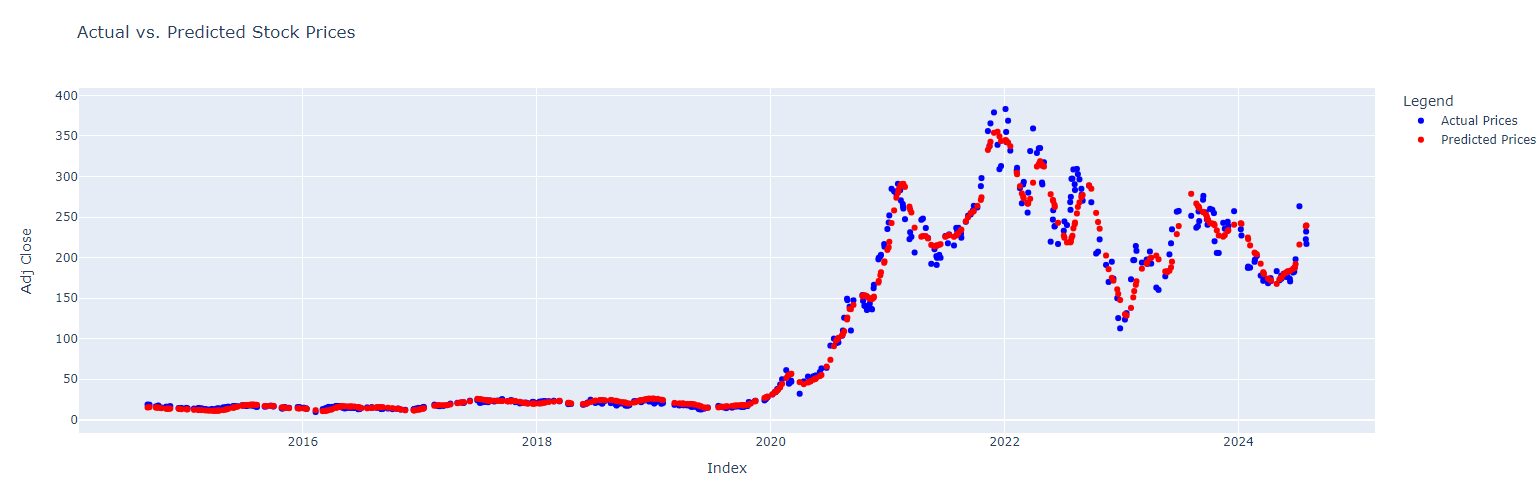
I decided to use k-fold to validate the model using k =5. Unfortunately, the validation increased the MSE from 264.49 to 297.52



### Ridge Regression – Steps (Assuming the data was defined and split as mentioned above)

* **Normalize features:**
  + Scaler: standardizes feature values to have mean 0 and variance 1.
  + X\_scaled: contains normalized feature values.
* **Polynomial features:**
  + poly generates polynomial features of degree 2 for capturing interactions.
  + X\_poly includes these polynomial features.
* **After you split the data you can initialize and fit Ridge Regression:**
  + ridge creates a Ridge Regression model.
  + grid\_search performs hyperparameter tuning to find the best `alpha` using cross-validation. (This was done simultaneously as opposed to the linear regression model)
* **Best model:** model - is the Ridge Regression model with the best hyperparameters.

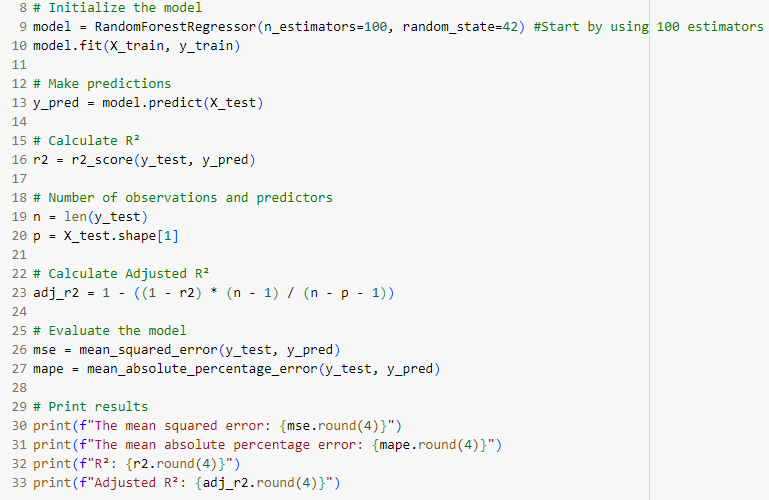




Notice how the difference between the graph from ridge regression and the graph from the linear regression are identical. This indicates their proximity in terms of performance, but the difference remains in the evaluation metrics.

### Random Forest Regression – Steps (Assuming the data was defined and split as mentioned above)

* **Initialize the model:**
  + model creates a Random Forest Regressor with 100 trees and a fixed random seed for reproducibility of 42.



The visualization for the random forest regressor looks completely different from the other 2



* 1. Model Evaluation

After running the models and tuning them, I used the following metrics to evaluate them and create the final stock price predictions for TSLA:

* **Mean Squared Error (MSE):** Measures the average squared difference between actual and predicted values, indicating the model's prediction error.
* **Mean Absolute Percentage Error (MAPE):** Represents the average absolute percentage difference between actual and predicted values, showing prediction accuracy in percentage terms.
* **R-squared (R²):** Indicates the proportion of variance in the dependent variable explained by the independent variables, with 1 being perfect fit and 0 being no fit.
* **Adjusted R-squared:** Adjusts R² for the number of predictors, providing a more accurate fit measure by penalizing unnecessary variables.

Summary of the metrics for the three models

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **Linear** | **Ridge** | **Random Forest** |
| MSE | 245.833 | 226.384 | 23.66 |
| MAPE | 0.093 | 0.088 | 0.024 |
| R² | 0.9796 | 0.9812 | 0.998 |
| Adjusted R² | 0.9794 | 0.9807 | 0.998 |

**Based on the 4 metrics, the Random Forest Regressor outperformed the other 2 models across all 4 metrics so this was the chosen model to create the predictions for the option pricing model.**

## Option Contract Price Predictions

6.1 Models

1. **Monte Carlo Simulation**

* **Principle:** Uses random sampling to simulate many price paths and estimates option prices by averaging the payoffs.
* **Process:** Generates numerous random paths and calculates the average payoff, discounted to the present value.
* **Advantages:** Flexible and can handle complex or exotic options, easy to implement for path-dependent features.
* **Disadvantages:** Requires many simulations for accuracy, results are subject to random sampling error, and computationally intensive.

1. **Binomial Tree**

* **Principle:** Uses a discrete-time model to simulate price movements with a tree structure.
* **Process:** Builds a lattice of possible prices and calculates option values backward from expiration.
* **Advantages:** Provides exact solutions for European options, can be adapted for American options, and does not rely on random sampling.
* **Disadvantages:** Becomes complex and computationally expensive with many steps or complex options.

In summary,

|  |  |
| --- | --- |
| **Monte Carlo** | **Binomial Tree** |
| Flexible and suitable for complex options, but needs many simulations to achieve accurate results. | Exact and structured, suitable for simpler options but can be complex for large trees. |

**Note:** The Black-Scholes model can only be used for European Options which can only be exercised upon expiration.

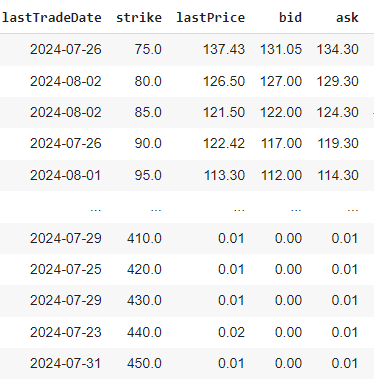
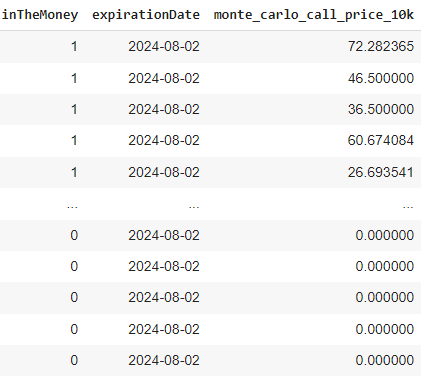
* 1. Monte Carlo Simulation

### Components of the formula:

The following components were extracted from the call options dataset

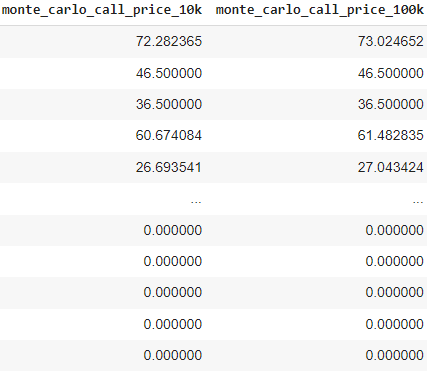
1. **The risk-free rate (r):** is a theoretical interest rate that represents the interest an investor would expect from a risk-free investment over a specific period of time. For the simulations this will be fixed at 5%
2. **Num\_simulations:** Is the number of iterations that will be used for the model (10,000 and 100,000)
3. **S:** Current stock price (last price).
4. **K:** Strike price of the option.
5. **T:** Time to expiration in years.
6. **Sigma:** Implied volatility of the stock.
7. **dt**: Time step for each day in the simulation.
8. **discount\_factor:** Discount factor to account for the time value of money.
9. **S\_paths:** Array to store simulated stock prices for each path and day.
10. **Z:** Random standard normal variables.
11. **call\_payoff:** Payoff of the call option at expiration for each simulation.
12. **call\_price:** Average discounted payoff of the call option, which is the Monte Carlo estimated call option price.

For the Monte Carlo Simulation with 10,000 simulations:

Notice how the “inTheMoney” column indicates that the call option would yield a profit ("1") if exercised at the strike price of $75.00. To determine the exact profit for each contract, we need the actual market price of the underlying stock.

I decided to run a second Monte Carlo simulation using 100,000 iterations. These were the results in comparison with the first model:



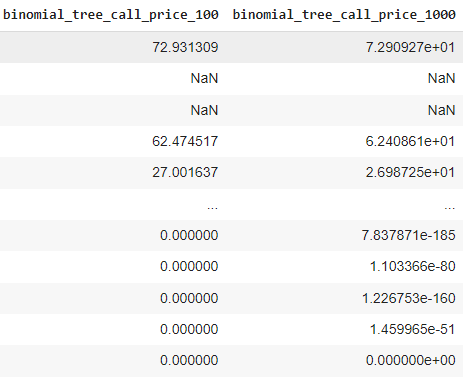
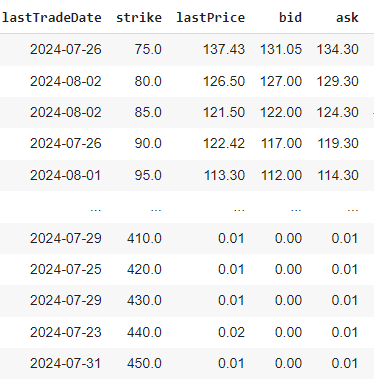
* 1. Binomial Tree

### Components of the formula:

The following components were extracted from the call options dataset

1. **r = 0.05:** Sets the risk-free interest rate.
2. **num\_steps = The** number of steps for the binomial tree (100 and 1000 were used)
3. **S:** Current stock price (last price).
4. **K:** Strike price of the option.
5. **T:** Time to expiration in years.
6. **Sigma:** Implied volatility of the stock.
7. **dt**: Time step for each day in the simulation.
8. **U (Up factor):** This represents how much the stock price can increase in one step in the binomial model. If the stock can go up by a certain percentage, the up factor calculates that percentage increase.
9. **d (Down factor):** It's the opposite of the up factor and calculates the percentage decrease.
10. **risk-neutral probability (p):** The probability that the stock price will move up in one step in a risk-neutral world.

Binomial tree - 100 vs. 1,000 steps:

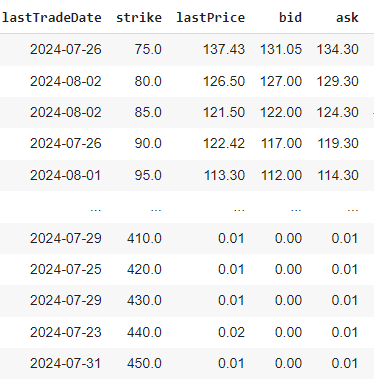


The initial results using a binomial tree with 100 steps showed option prices consistently below the strike price, which is expected for in-the-money options. However, as the strike price increased, there was an unexpected sharp decline in predicted option prices. This decline contradicts the theory that option prices should align closely with the value of the underlying asset (TSLA) and remain stable if the company is financially sound.

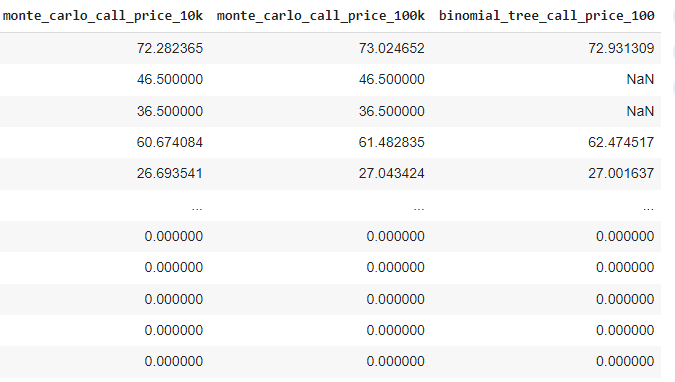
In contrast, the binomial tree model with 1,000 steps failed to predict any prices for in-the-money options, leading to its exclusion from further analysis.

* 1. Comparison of the models

Recall that the strike price is $75.00 for our first listed contract:

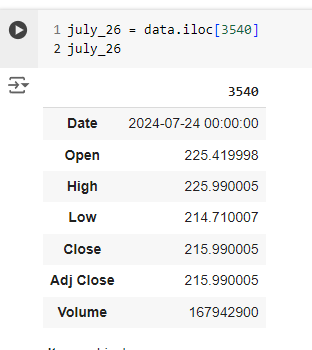


From the figure below, you can see that the price predictions from both Monte Carlo simulations and the binomial tree with 100 steps fall within the same range. However, we can only rely on the Monte Carlo simulations, as the binomial tree occasionally failed to produce a price. Among the models, the Monte Carlo simulation with 100,000 iterations is preferred. This is because using a more conservative estimate for option prices ensures that our profit calculations are more cautious by assuming higher option prices:

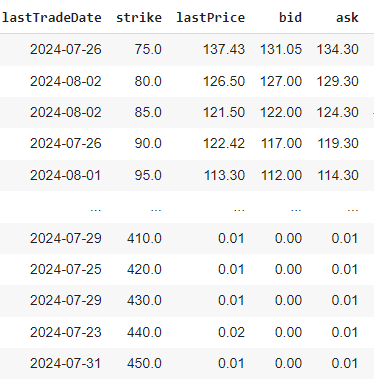


* 1. An example using the result

1. The stock price of TSLA on July 26, 2024 was $215.99.



1. For a call option to be in the money (Profitable) we would have to add the strike price plus the actual cost of the call contract (Premium). Remember that the strike price for an option contract on the same day was $75.00



1. If we use the price predicted by the Monte Carlo simulation with 100,000 iterations we would be saying that the contract that can be exercised at $75.00 cost $73.02 (Premium).

This means that for this option contract to be “in the money” the price of the underlying asset (TSLA) has to be higher than the cost and the premium:

* TSLA (July 26) = $215.99
* Premium (Monte Carlo Simulation) = $73.02
* Strike = $75.00

If an investor had 100 contracts, which is the common size for options trading, exercised on July 26 for a price of $75 when TSLA was $215.99 my total profit would be $6,797 for the trade.

## Random Forest Regression and Monte Carlo Simulation Integration

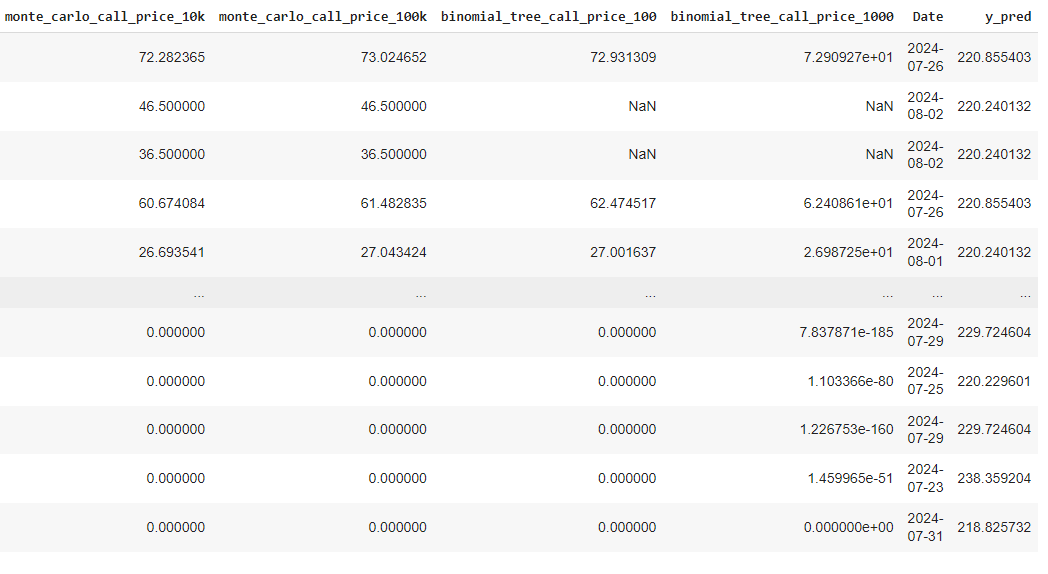
* 1. Introduction

We will use the TSLA price predictions from the Random Forest Regressor model for our analysis. With a mean squared error (MSE) of 23.66, the model’s predictions are expected to deviate by approximately ± $5.00 from the actual stock prices on average. This deviation is calculated using the square root of the MSE. We will compare these predicted prices to the adjusted closing prices from the original dataset, allowing for a ± $5.00 margin of error. This approach will help us assess the model's performance and evaluate the profitability of options based on their strike prices.

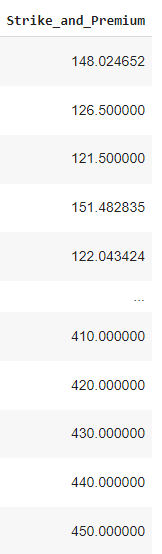
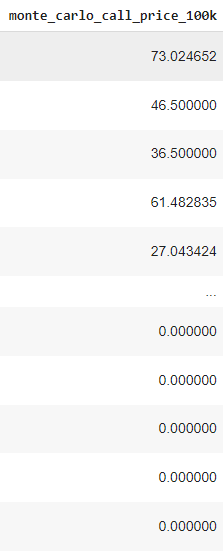
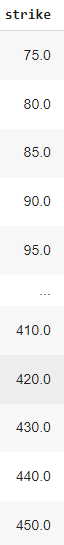
In summary, the goal is to calculate option profits or losses based on the model’s predicted stock prices rather than the actual adjusted closing prices, and to assess whether options are "In-the-Money" or "Out-of-the-Money" with this margin of error.

* 1. Process

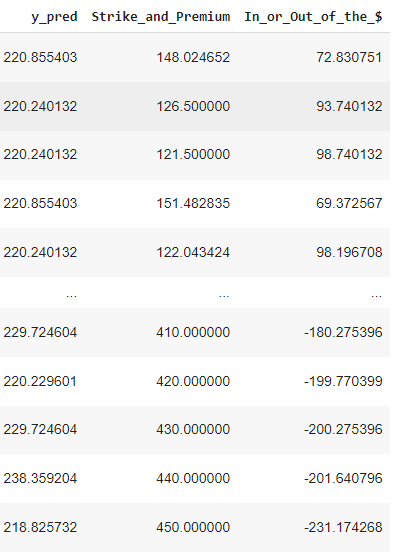
1. Include the predictions from the random forest regression (y\_pred) into the call options dataset.



1. Add a column with the strike and the premium to reflect the total cost if the option were to be exercised:



1. Now that we have the total cost of the investment, we can calculate the profit and see if the options are in or out of the money based on our predictions:



Interpretation

As long as the strike price and the premium are below the actual market price, a call option will be in the money. This principle applies to any underlying security. When the strike price and the premium exceed the market price, the investment starts to incur losses. However, it's important to note that having an option that is out of the money does not mean you will lose $180.27, as shown in the example above. If you buy a call option, your loss is limited to the amount you paid for it, as you are not legally obligated to exercise the option.

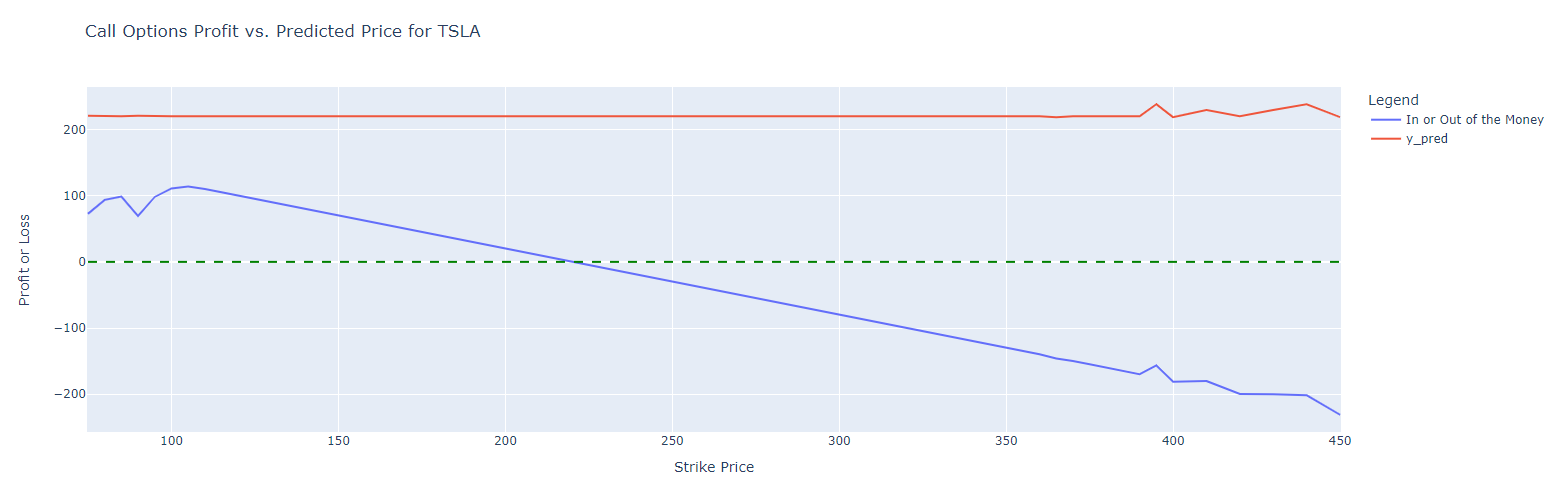
For example, if we use the previous example but change the premium from $75.00 to $200.00, the calculation would be:

* TSLA (July 26) = $215.99
* Premium (Monte Carlo Simulation) = $200.00
* Strike = $75.00

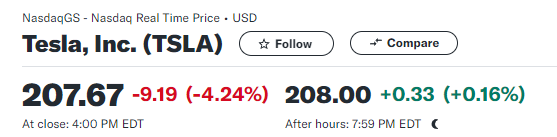
You are not obligated to exercise the option at $215.99; instead, you can let the option expire, accepting a loss equal to the full premium paid. For 100 contracts, this would result in a total loss of $20,000.

* 1. Model Performance

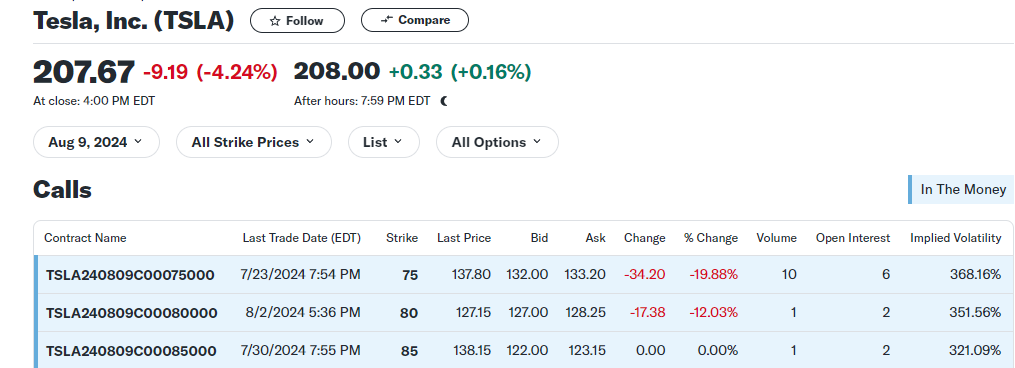
For all predicted prices above $200 by the Random Forest model, which reflect TSLA’s current performance, there was a decrease in profits, as indicated by the blue line. Once the strike prices exceeded $225.00, the call options became "out of the money."



Although the expiration date of the contracts was not highlighted in this project, it's important to note that the options in our dataset expire today, August 2, 2024, when TSLA’s adjusted closing price is $207. This means that our profit and loss calculations are as accurate as the model's predictions allow. Options with a premium and strike price above $207 by the end of the day are considered out of the money. Consequently, the model predicted a $0.00 price for options with a strike price above $400. Investors typically buy call options when they expect the stock price to rise above the strike price. Since TSLA did not reach this threshold, investors of those call options contracts above $200.00 faced losses today:



Even if we examine new call option contracts and apply the same models, we should still be able to generate reliable predictions, given TSLA’s consistently high volatility, as depicted in the initial graphs of this project. Therefore, the model is expected to continue performing at this level. See below:



* 1. Analysis and Practical Implications

1. **Error Analysis:** The error was assessed by comparing the actual adjusted closing prices to the predictions made by the Random Forest regression model. This comparison was intended to evaluate the accuracy of the call option contract profitability when using prices generated by the Monte Carlo simulation versus actual market prices.
2. **Practical Implications**

* **Profitability Analysis:** The profitability of options must be evaluated on a contract-by-contract basis because their prices can vary depending on the seller. For example, today you might be able to buy a call option for TSLA for $50 from one broker and for $60 from another, based on their “ask” prices. Despite both options being for the same underlying asset, the profit margins will differ. Therefore, I assessed the profitability of the options using the strike price and premium to calculate the total cost for each contract.
* **Limitations:** TSLA is relatively new compared to established auto manufacturers like GM or Toyota, resulting in a more limited historical data set. Additionally, the absence of a dividend yield affected the returns initially calculated at the beginning of the project.
* **Considerations:** Assuming that TSLA is a profitable investment solely based on stock price movements would be an oversight. A comprehensive financial analysis involves many more factors, and a machine learning model for investment decisions should incorporate these aspects to be effective in real-world applications.

## Conclusion

* 1. Summary of Findings

Best Model Based on Metrics

1. **Mean Squared Error (MSE):**
   1. Lower is Better: This metric indicates the average squared difference between predicted and actual values. A lower MSE suggests better model performance.
   2. Random Forest has the lowest MSE (23.66), indicating it has the smallest average squared prediction error among the three models.
   3. Ridge has a lower MSE than Linear Regression, suggesting that Ridge Regression is better at minimizing prediction errors compared to Linear Regression.
2. **Mean Absolute Percentage Error (MAPE):**
   1. Lower is Better: MAPE measures the average magnitude of the errors in percentage terms. A lower MAPE means the model's predictions are closer to the actual values.
   2. Random Forest has the lowest MAPE (0.024), meaning it provides the most accurate percentage error in its predictions.
   3. Ridge Regression has a lower MAPE than Linear Regression, showing it performs better in terms of percentage errors.
3. **R-squared (R²):**
   1. Higher is Better: This metric represents the proportion of variance in the dependent variable that is predictable from the independent variables. A higher R² value indicates a better fit.
   2. Random Forest has the highest R² (0.998), meaning it explains almost all of the variance in the data.
   3. Ridge Regression has a slightly higher R² than Linear Regression, but not as high as Random Forest.
4. **Adjusted R-squared (Adjusted R²):**
   1. Higher is Better: This metric adjusts R² for the number of predictors in the model, providing a more accurate measure of model performance, especially with multiple predictors.
   2. Random Forest also has the highest Adjusted R² (0.998), reflecting that it not only fits the data well but also accounts for the complexity of the model.

### Summary

* Random Forest outperforms both Linear and Ridge Regression across all metrics. It has the lowest MSE and MAPE, and the highest R² and Adjusted R², indicating superior prediction accuracy and model fit.
* Ridge Regression performs better than Linear Regression in terms of both MSE and MAPE, as well as slightly better R² and Adjusted R² values.

**Conclusion:** Based on the provided metrics, Random Forest is the best model among the three, offering the lowest errors and the best fit for the data.

* 1. Future Work

### Improvements:

* **Expand the dataset**: Acquire additional option contracts to enhance the dataset, as the available data from Yahoo Finance may be limited. Consider exploring brokers that provide more comprehensive datasets.
* **Enhance visualizations:** Explore and utilize additional libraries for more sophisticated and interactive charts to improve data representation.
* **Consolidate data:** Integrate multiple subsets into a single Data Frame to streamline data management and simplify graphing and analysis processes.
* **Develop a Stronger Foundation in Options Trading and Industry Knowledge:** At the start of this project, I had a basic understanding of how options work but lacked detailed knowledge on applying machine learning to these securities. A deeper familiarity with coding for derivatives and options trading strategies would have accelerated the process and improved the efficiency of translating theoretical concepts into practical machine learning applications.
  1. Applicability of the model

Model Performance on Less Volatile Securities

The model is anticipated to perform more effectively with less volatile securities. This is because the simple moving average (SMA) method used in the model is particularly adept at tracking stock price movements when there are no significant fluctuations. In less volatile securities, stock prices change in a more predictable and gradual manner, which allows the SMA to provide smoother and more accurate predictions.

When stock prices do not experience sharp swings, the SMA can effectively capture the underlying trend, leading to more reliable and stable forecasts. As a result, the prediction error is expected to be lower for less volatile securities. In contrast, for highly volatile stocks, where prices exhibit frequent and unpredictable changes, the SMA may struggle to keep up with rapid movements, leading to increased prediction errors.

Therefore, across the dataset, we anticipate that securities with lower volatility will yield better model performance, with smaller prediction errors and more accurate results.

Adaptability for Other Options

In addition to assessing call options, this model can be adapted for evaluating put options and predicting bearish stock investments. The principles underlying the model remain applicable, but adjustments will be needed to account for the unique characteristics of put options and bearish strategies.

Put Options: The model can be adapted to calculate the profitability of put options by modifying the profit and loss calculations to reflect the dynamics of put options, which profit from declines in stock prices. By integrating the relevant strike prices, premiums, and market prices, the model can estimate the potential returns for put options.

Bearish Stock Investments: For bearish stock investments, where the expectation is that stock prices will decline, the model can be adjusted to predict price movements and profitability from a downward trend perspective. This involves revising the predictive features and calculations to focus on scenarios where the value of the underlying asset decreases.

By tailoring the model to accommodate these variations, it can provide valuable insights and accurate predictions for a broader range of financial strategies beyond just call options.

* 1. References

<https://www.nasdaq.com/market-activity/quotes/option-chain>

<https://www.investopedia.com/articles/optioninvestor/09/buying-options.asp>

<https://www.merrilledge.com/investment-products/options/options-pricing-valuation>

<https://pages.stern.nyu.edu/~sfiglews/documents/What%20does%20an%20option%20pricing%20model%20tell%20us.pdf>

<https://www.investopedia.com/articles/optioninvestor/07/options_beat_market.asp>

<https://www.investopedia.com/terms/c/calloption.asp>

<https://www.investopedia.com/articles/trading/09/linear-regression-time-price.asp>

<https://prjpublication.com/backend/file/Stock%20Index.pdf>

<https://www.youtube.com/watch?v=vuKfuDJQUJk>

<https://www.youtube.com/watch?v=RHeUqqrxP-w>

<https://finance.yahoo.com/quote/TSLA/options/>

<https://finance.yahoo.com/quote/TSLA/>

<https://www.investopedia.com/tesla-s-largest-revenue-source-is-automotive-sales-4799069#:~:text=6%20The%20segment%20posted%20a,million%20reported%20in%20FY%202020>

* 1. Appendices

**Code and Data:**

<https://colab.research.google.com/drive/19_Kuhp52PW7PpP62h3f3JNFhaeTMCnjW?usp=sharing>