**Stock Price Prediction Using ARIMA + GARCH: Tesla (TSLA) Example**

**1. Objective**

The main goal of this project is to forecast next day return and volatility for Tesla (TSLA) stock based on historical price data.  
Additionally, we estimate the forecasted price and the uncertainty range (confidence interval) for the next trading day.

**Why Tesla?**

* Tesla is a highly traded and volatile stock, making it a good candidate to demonstrate ARIMA + GARCH modeling.
* Sufficient historical data is available on Yahoo Finance.

**2. Introduction**

Stock price prediction is an important task in financial modeling. Its goal is to give investors and traders an idea of the probable future market behavior.

* **ARIMA (AutoRegressive Integrated Moving Average):**
  + Captures the predictable part of stock returns (trend/mean).
  + Provides the next day expected return.
* **GARCH (Generalized AutoRegressive Conditional Heteroskedasticity):**
  + Models the conditional variance of residuals (actual return minus ARIMA predicted return).
  + Captures volatility clustering, which is common in stock markets.

**Combination:** ARIMA forecasts the mean, and GARCH models the volatility. Together, they provide both **expected return** and **risk estimate**.

**3. Tools and Libraries**

* **Python** programming language.
* **Libraries used:**
  + yfinance → To download historical stock data
  + pandas / numpy → For data handling and calculations
  + matplotlib → For plots and visualizations
  + statsmodels → For ARIMA modeling and diagnostics
  + arch → For GARCH model implementation

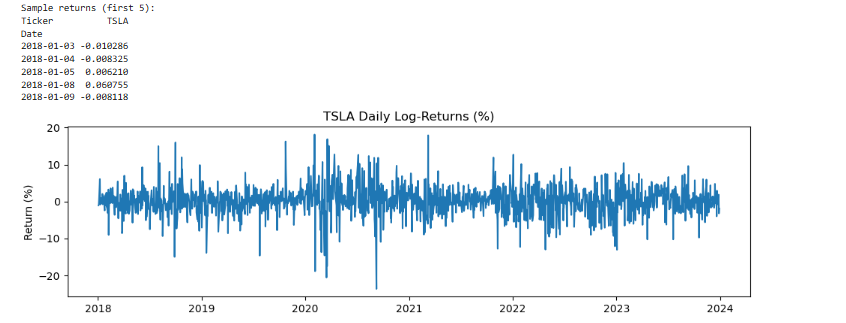
**4. Methodology / Steps**

**Step 1: Download Historical Price Data**

* Tesla (TSLA) adjusted close prices are downloaded from Yahoo Finance.
* Date range: January 1, 2018 to December 31, 2023.
* Data is checked for missing values or inconsistencies.

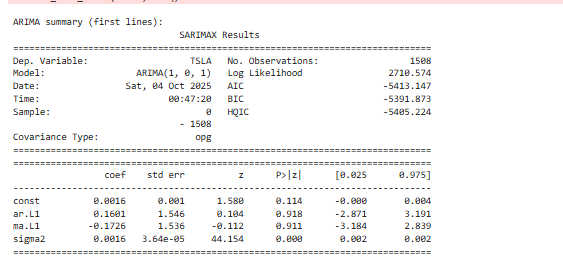
**Step 2: Calculate Log-Returns**

* Returns are calculated as: log(today’s price / yesterday’s price).
* Log-returns are stationary, making them suitable for modeling.



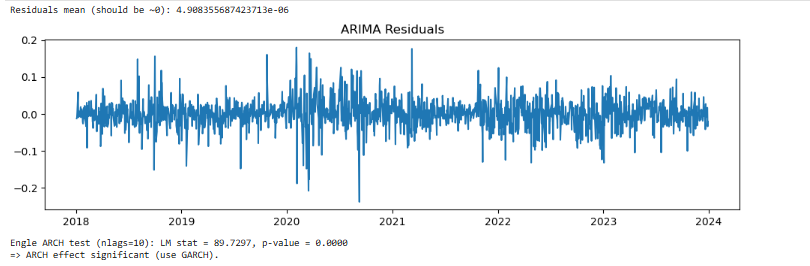
**Step 3: Fit ARIMA Model**

* ARIMA(1,0,1) is applied on returns.
* Purpose: model the mean or predictable part of returns.
* Output: next-day expected log-return.



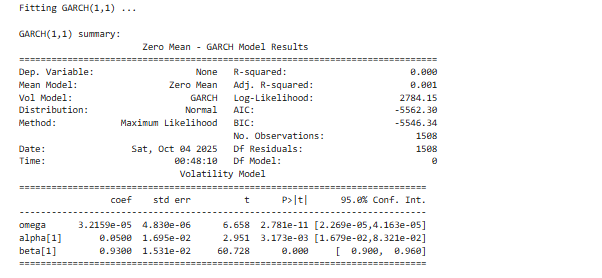
**Step 4: Check for ARCH Effects**

* Residuals from ARIMA are tested using the ARCH test.
* Purpose: check if volatility clustering exists in the residuals.
* If p-value < 0.05, GARCH modeling is justified.



**Step 5: Fit GARCH Model**

* GARCH(1,1) is fitted on residuals.
* Purpose: model the conditional variance (volatility).
* Output: forecasted volatility (standard deviation) for the next few days.



**Step 6: Forecast Returns and Volatility**

* ARIMA predicts the next-day log-return.



The ARIMA (1,0,1) model predicts a next-day log-return of 0.00194, which corresponds to an expected price increase of approximately 0.19%. Based on the last observed price of $248.48, the forecasted price for the next trading day is $248.96.

* GARCH predicts volatility for the next 5 days.

A number of numbers and symbols

AI-generated content may be incorrect.

Standard deviation from GARCH indicates the uncertainty in price. The GARCH (1,1) model forecasts the stock’s volatility for the next 5 days.  
The predicted daily standard deviation ranges from 2.94% on Day 1 to 3.04% on Day 5, showing a slight increase in volatility.  
This indicates the level of uncertainty around the ARIMA forecasted return, helping investors estimate the potential price range for each day.

**Step 7: Convert Log-Return Forecast to Price**

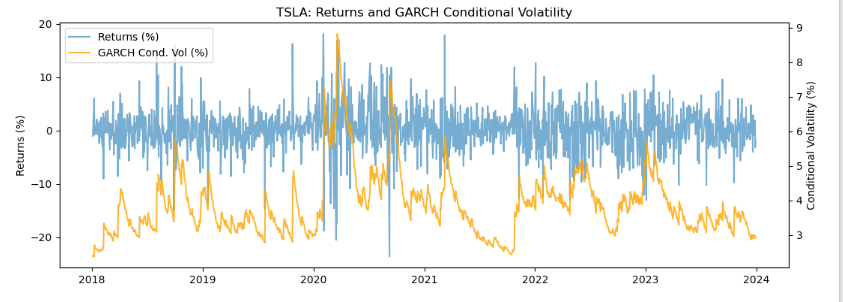
* Last observed price is taken, and forecasted log-return is converted to price using the exponential formula.
* Point forecast price is calculated.

**Step 8: Compute Confidence Interval**

* ±1 standard deviation (from GARCH) is used to calculate the forecasted price range.
* This gives an approximate range for the next-day price.

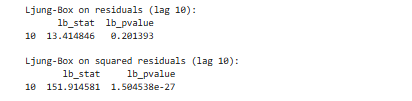
**Step 9: Visualization**

* Returns and conditional volatility are plotted together.
* Forecasted price point and volatility range can be compared visually.



**Step 10: Diagnostics**

* Residuals are tested for autocorrelation using the Ljung-Box test.
* Purpose: check if the model is properly fit and residuals are random.



Ljung-Box test results show that residuals from the ARIMA (1,0,1) model do not have significant autocorrelation (p-value = 0.201), indicating that the mean model fits the data well.  
However, squared residuals show strong autocorrelation (p-value ≈ 1.5e-27), confirming the presence of volatility clustering. This justifies using the GARCH (1,1) model to forecast conditional variance.

**5. Results**

| **Metric** | **Value** |
| --- | --- |
| Last observed price (2023-12-29) | $248.48 |
| Forecasted next-day price | $248.96 |
| Approx. 1-day volatility range | $244.20 → $253.85 |

**Interpretation:**

* Forecasted price is slightly higher than the last observed price, indicating a small upward movement.
* Volatility range shows the realistic uncertainty of the next-day price.
* Investors can assess risk using this range.

**6. Conclusion**

* ARIMA + GARCH is a simple yet effective method for stock price prediction.
* The model provides both a **point forecast** and an **uncertainty range**, offering practical insights.
* Possible improvements:
  + EGARCH / GJR for asymmetric volatility
  + Heavy-tailed distributions (like t-distribution)
  + Multi-step forecasts
* This project is useful for educational purposes and can help investors evaluate risk.

**7.Key Learnings:**

1. Stock returns can be modeled using ARIMA for mean prediction.
2. Volatility clustering exists in stock markets, confirmed by Ljung-Box on squared residuals.
3. GARCH model effectively captures conditional variance (risk) over forecast horizon.
4. ARIMA + GARCH combination provides both point forecast and price range, useful for investors.
5. Real financial data can be analyzed practically using statistical models.
6. Model diagnostics are important to ensure reliability of forecasts.
7. Python libraries make data download, analysis, modeling, and visualization efficient and reproducible.