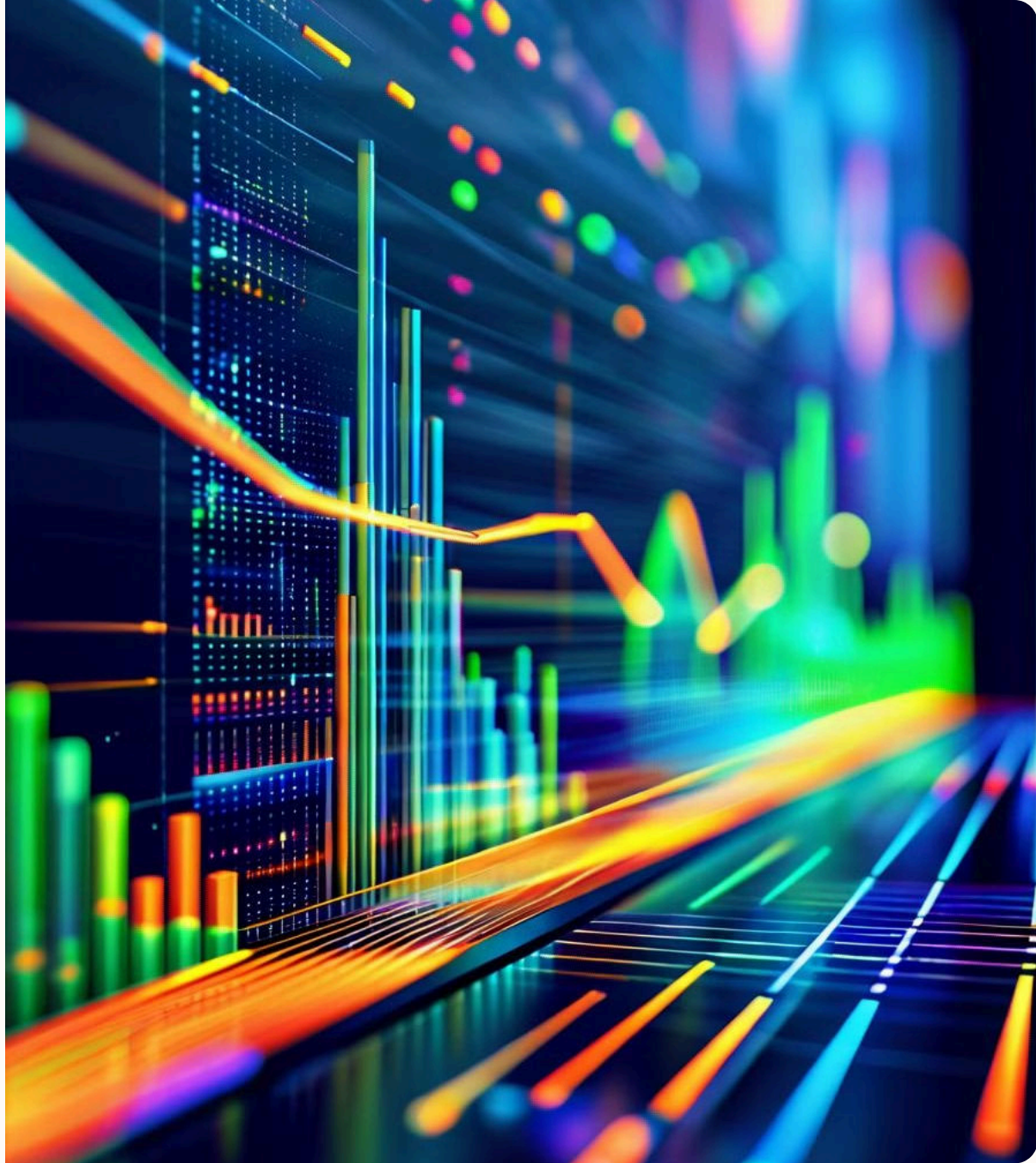
Predicting Stock Prices

AI Boot Camp Project 2

Team Members:

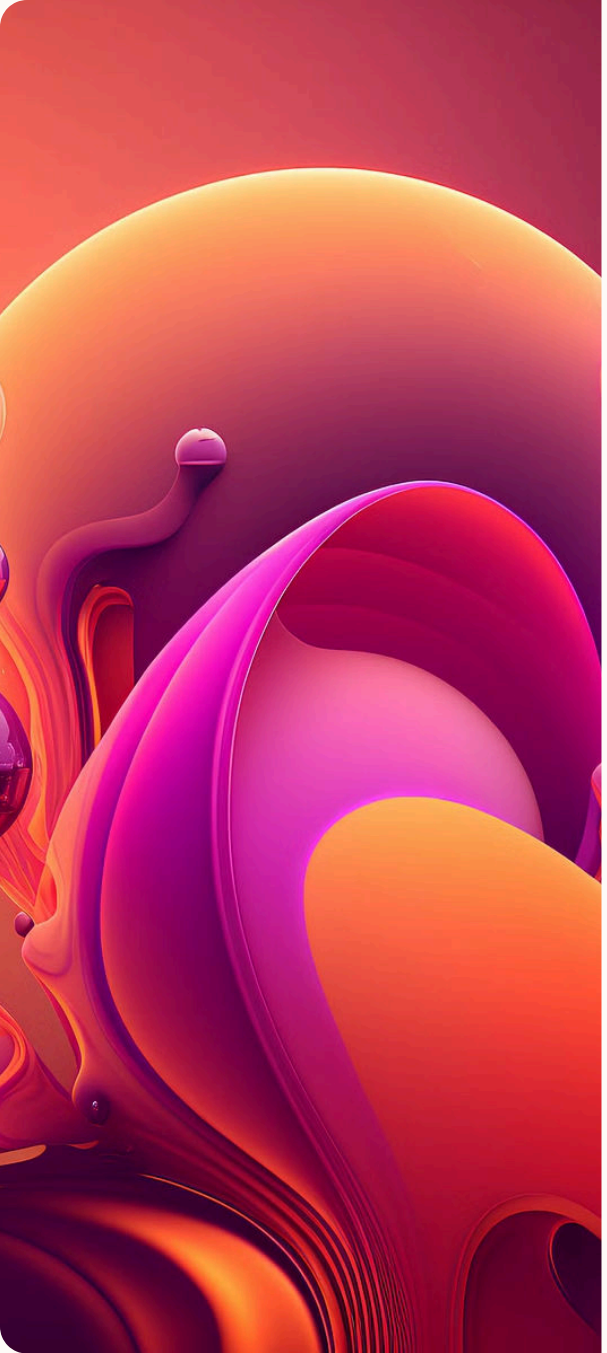
- Priscilla Morales
- Leigh Nash
- J'mari Hawkins
- Dzemal Crnkic
- Antoine Baize
- Andy Bhanderi



Project Overview

This project involved extracting, cleaning, and transforming historical stock data, training machine learning models, and developing an interactive application for users to predict stock prices for up to one year in advance.

- **Methodology:** By incorporating different machine learning techniques, including Support Vector Regression (SVR), XGBoost, Linear Regression, and the Prophet model, we explored the strengths and weaknesses of each approach in handling time series data and capturing complex patterns in the stock market.
- **Evaluation**: Primarily we used Mean Squared Error (MSE) and R-squared (R2)
- **Goal:** Provide a tool in making informed stocks decisions for personal and professional use.



Stock Data Collection

Sources: Data was extracted using the Yahoo Finance API

Ten diverse stocks: Apple (AAPL), Microsoft (MSFT), Johnson & Johnson (JNJ), JPMorgan Chase & Co. (JPM), Procter & Gamble (PG), Exxon Mobil (XOM), NVIDIA (NVDA), Pfizer (PFE), Coca-Cola (KO), and Tesla (TSLA).

Period: Ten years of data were collected to capture a broad spectrum of market dynamics.

Target Variable: The closing price for the day.

Our Approach

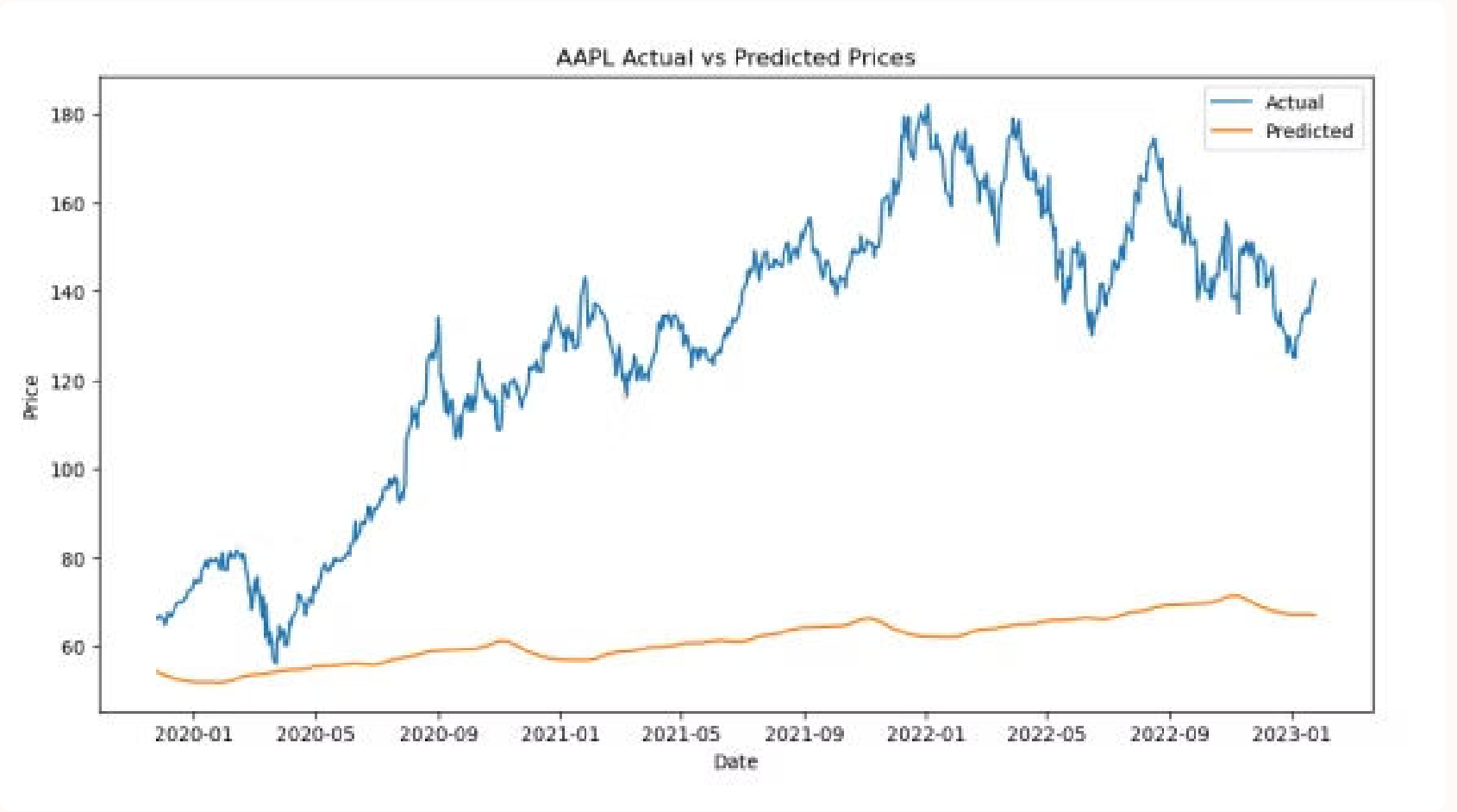
- Initially, we thought predicting stock prices would be too challenging due to the complexity and variability inherent in financial markets.
- Despite the daunting task, our team member, J'Mari, was able to develop the neural network model we just showed.
- We were curious what those data would look like using more traditional and straightforward models that we are more familiar with.
- At this point, we set out to answer the question: Which model is best at predicting stock prices?

Results

Prophet Model

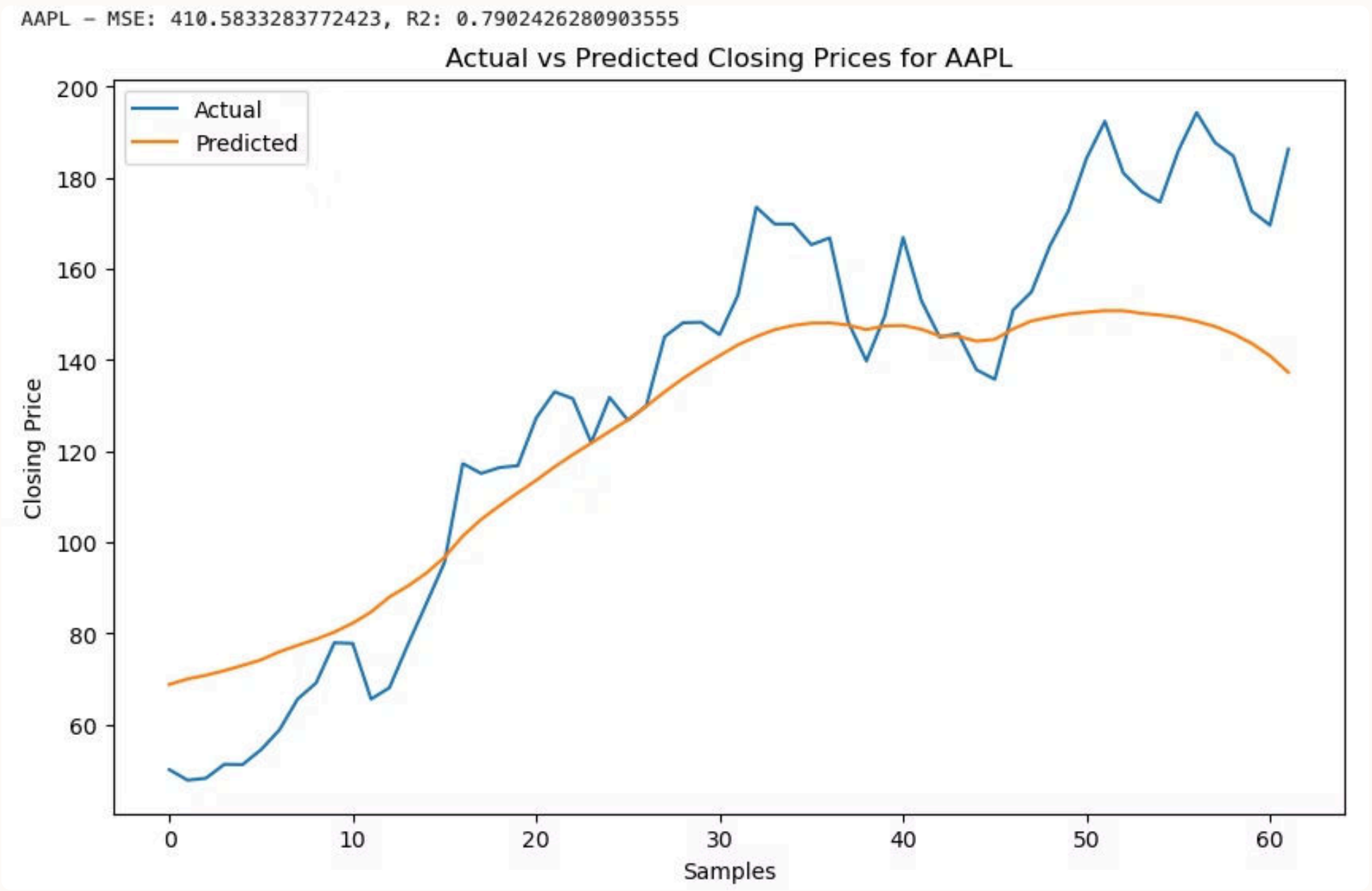
Mean Squared Error: 5247.210288124654 Mean Absolute Error: 66.92327050979875

Mean Absolute Percentage Error: 49.20907363583441%

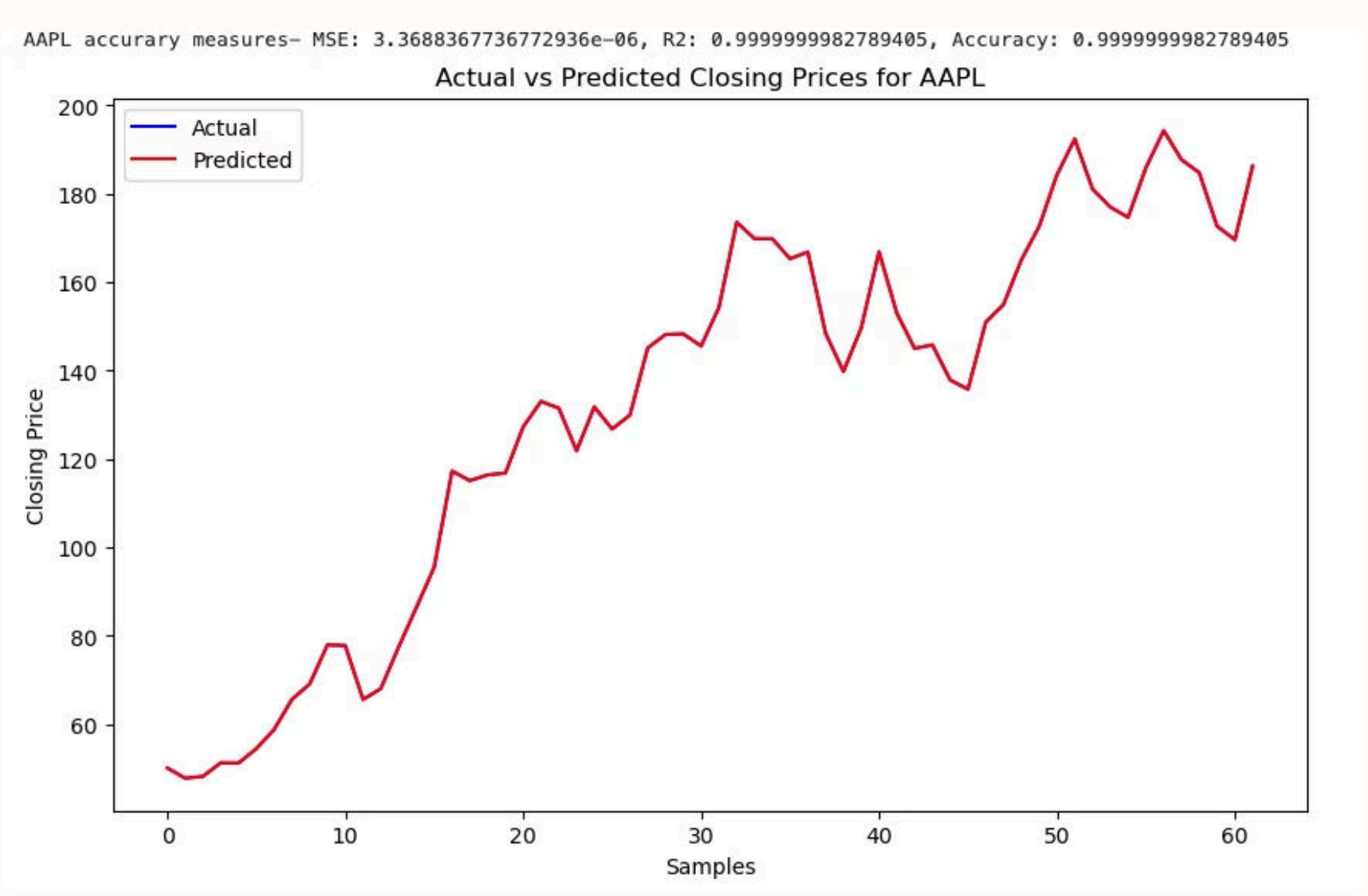


Stocks with Consumer Data

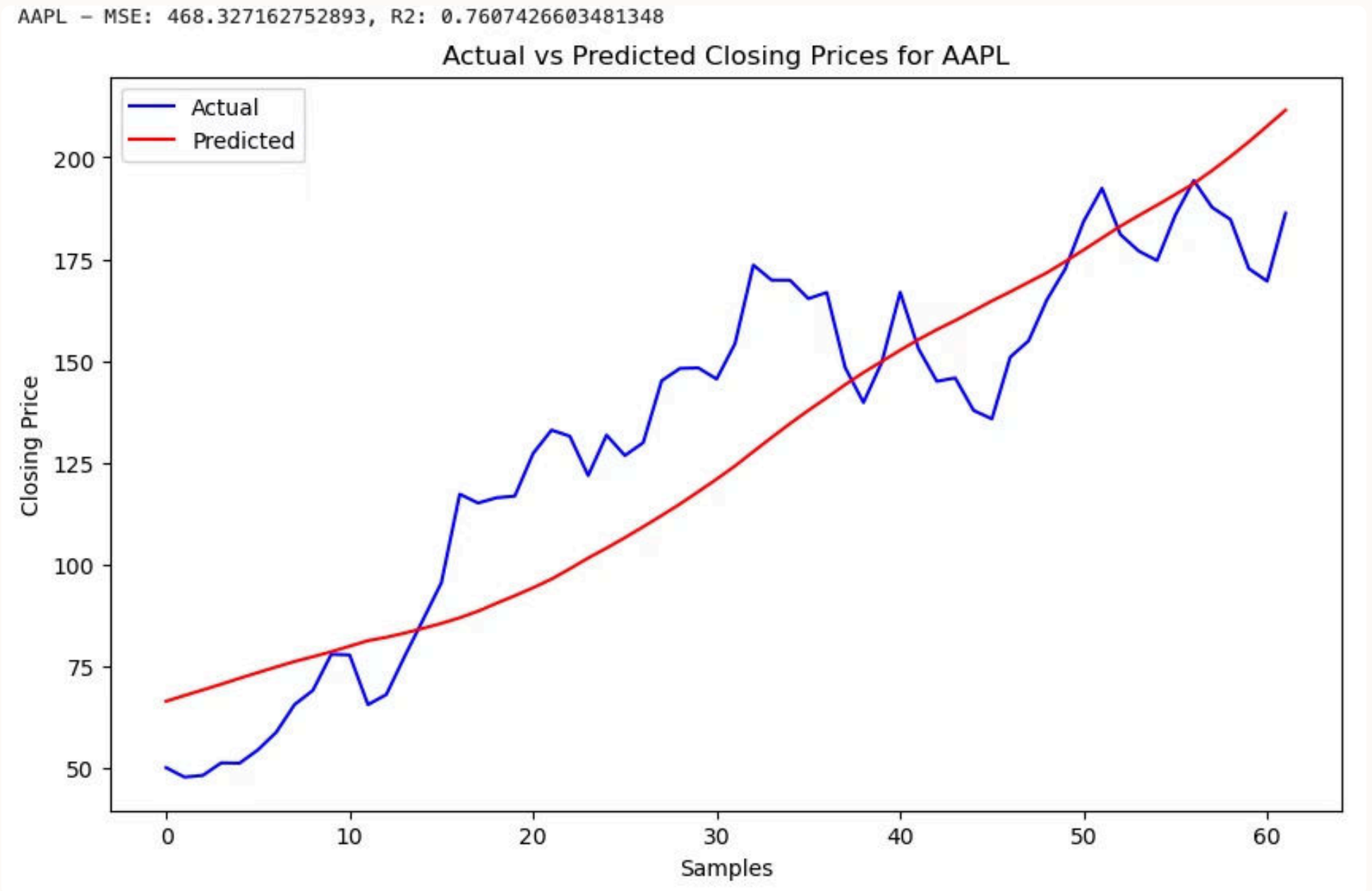
Support Vector Regression Model



XGBoost Model



Linear Regression Model



The battle of the ML Models

Comparing the MSE for the first 4 models

Example MSE comparison for Apple stock:

Model	MSE
LSTM	.000426
LR	0.028
SVR	131.14
XGB	0.67
Prophet	5247.21
LR (with Consumer Data)	468.3
SVR (with Consumer Data)	410.6
XGB (with Consumer Data)	3.36

In conclusion...

- We found that the LSTM neural network outperformed all other models in predicting stock prices.
- Traditional machine learning models like Linear Regression and XGBoost performed ok but they were unable to match the LSTM's ability to capture temporary trends and long-term patterns in the data.
- The Prophet model, while useful for capturing seasonality and trends, fell significantly short in accuracy, highlighting its limitations in stock price prediction.

LSTM Results By the Numbers

- The LSTM models were evaluated using the Mean Squared Error (MSE) on both training and testing datasets.
  - The results demonstrated high predictive accuracy, with percentage accuracy ranging between 92% to 97%.

AAPL

Train MSE: 0.0001156,

Test MSE: 0.0004256,
Accuracy: ~95.8%

MSFT

Train MSE: 0.0000755,

Test MSE: 0.0003331,
Accuracy: ~96.5%

JNJ

Train MSE: 0.0003117,

Test MSE: 0.0004725,
Accuracy: ~95.3%

JPM

Train MSE: 0.0002473,

Test MSE: 0.0003887,
Accuracy: ~95.9%

PG

Train MSE: 0.0002478,

Test MSE: 0.0005441,
Accuracy: ~94.9%

XOM

Train MSE: 0.0003113,

Test MSE: 0.0007872,
Accuracy: ~92.2%

NVDA

Train MSE: 0.0000415,

Test MSE: 0.0008594,
Accuracy: ~91.3%

PFE

Train MSE: 0.0003272,

Test MSE: 0.0005840,
Accuracy: ~94.6%

KO

Train MSE: 0.0006123,

Test MSE: 0.0008331,
Accuracy: ~91.4%

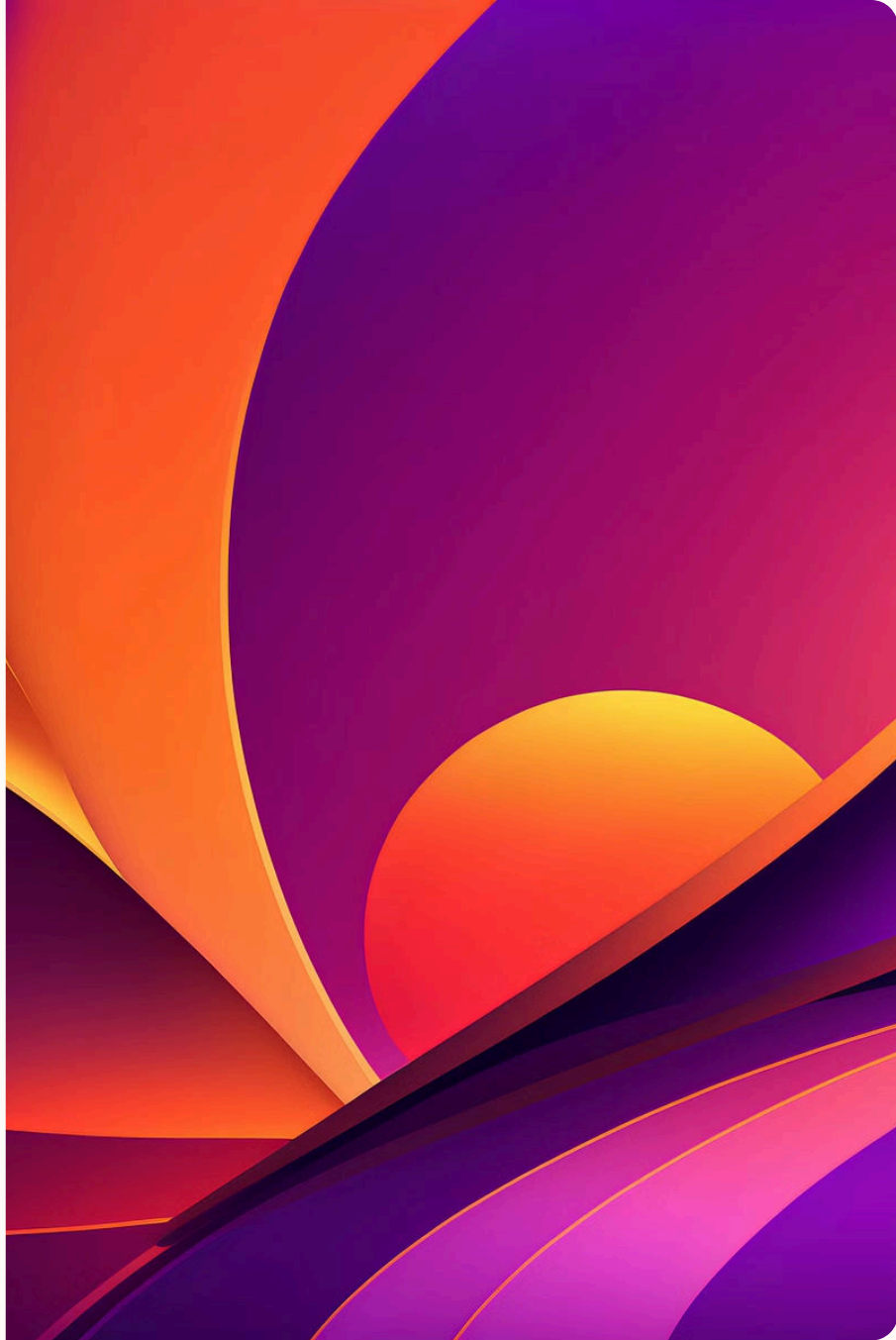
TSLA

Train MSE: 0.0001733,

Test MSE: 0.0005969,
Accuracy: ~94.5%

Problems Encountered

- Choosing the correct model (Neural Networks vs. Regression vs. Classification)
- Over and Under-fitting of models: Prophet and regression models had high MSE scores
- Long model training time (~20 mins)
    - until now!



Plans for Future Development

Enhance Model Complexity/Features

We could investigate the impact of additional financial indicators and socio-economic data. We would enhance model performance by expanding the feature set and evaluating the changes in predictive accuracy.

Handling Market Anomalies

Analyze the models' performance during periods of market instability and develop methods to improve robustness, such as incorporating anomaly detection mechanisms.

Add More Stocks

Expand the dataset by incorporating a wider range of stocks to run through the LSTM model. This expansion would enable the model to capture a broader spectrum of market behaviors and trends, potentially leading to more robust and reliable predictions.