Salesforce 1A
Stock Price Prediction from Financial News

December 2025

AI Studio Final Project

BREAK THROUGH TECH

Meet Our Team

Shambhavi Bhandari
CSULA, CS & BME

Karla Nguyen
SJSU, CS

Chris Chen
UCB Applied Math

Judy Ojewia
U of U, CE

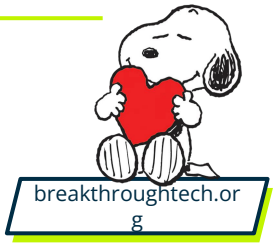
Our AI Studio Coach & Challenge Advisor

Leah Dsouza
AI Studio Coach

Atena Sadeghi
Challenge Advisor

Presentation Agenda

Introduction	5 minutes
Project Overview	5 minutes
Data Understanding and Preparation	5 minutes
Modeling and Evaluation	5 minutes
Final Thoughts	5 minutes



AI Studio Project Overview

This project explores the intersection of financial news and market prediction, aiming to forecast short-term stock price movements using machine learning and large language models (LLMs)

Our Goals

1. Build an accurate ML pipeline to use news sentiments and stock history to forecast next day stock trends
2. Evaluate model performance using accuracy (85% >)
3. Create visuals to show data patterns, sentiment trends, and model predictions

Business Question

- **Core Question:** Can financial news sentiment combined with historical data improve the accuracy of short-term stock price predictions?

- **Specific Goal:** Predicting the Next Day Opening Price for S&P 500.

- **Business Impact:**
    - Inform investment decisions (Risk Management).
    - Understand how public perception (News) impacts market value.

Business Impact

1. **Informs investment and risk decisions** ⇒ smarter financial or strategic investments
2. **Enhances market awareness** ⇒ better anticipate how public perception impacts market value
3. **Scalable for other industries** ⇒ analyzing news impact in sectors

Our Approach

Select Target

- Ticker: S&P 500
- Target: Opening price

Collect & Align Data

- Pull historical price data (yfinance)
- Combine with financial news headlines

Extract Sentiment via FinBERT

- Generate probabilities
- Compute daily sentiment scores

Engineer Features

- Lagged price metrics (returns, volatility, moving averages)

Train & Evaluate Models

- Tested different models → XGBoost
- Metrics: accuracy

Resources We Leveraged

1. Google CoLab
2. Python Libraries: yfinance, pandas, NumPy, matplotlib, Shap, gradio

Data Understanding and Data Preparation

Data Source

Market Price

- Source: Yahoo Finance (via yfinance)
- Daily OHLCV features:
  - Open, High, Low, Close, Volume
- Rolling averages, volatility

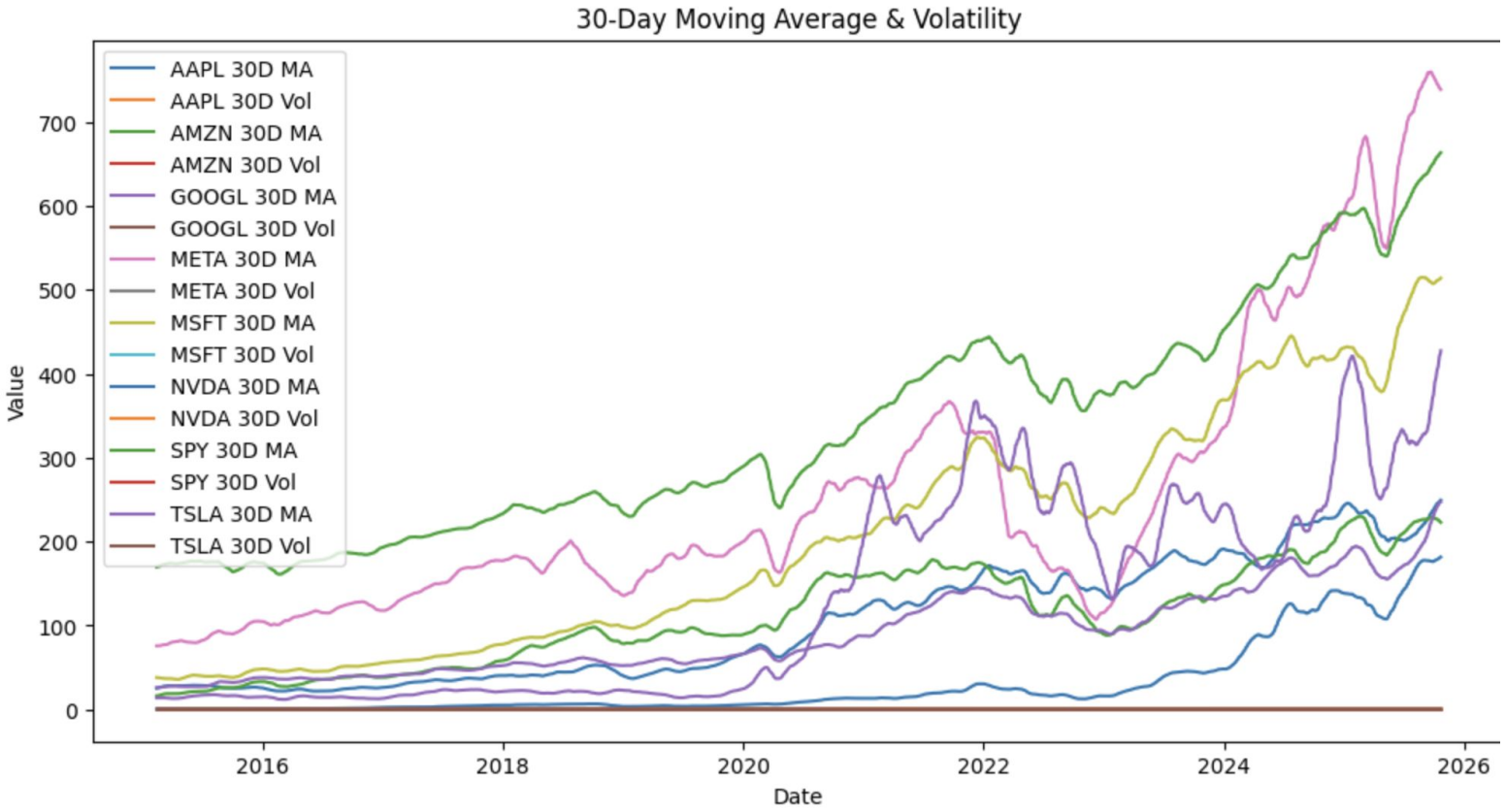
Financial News

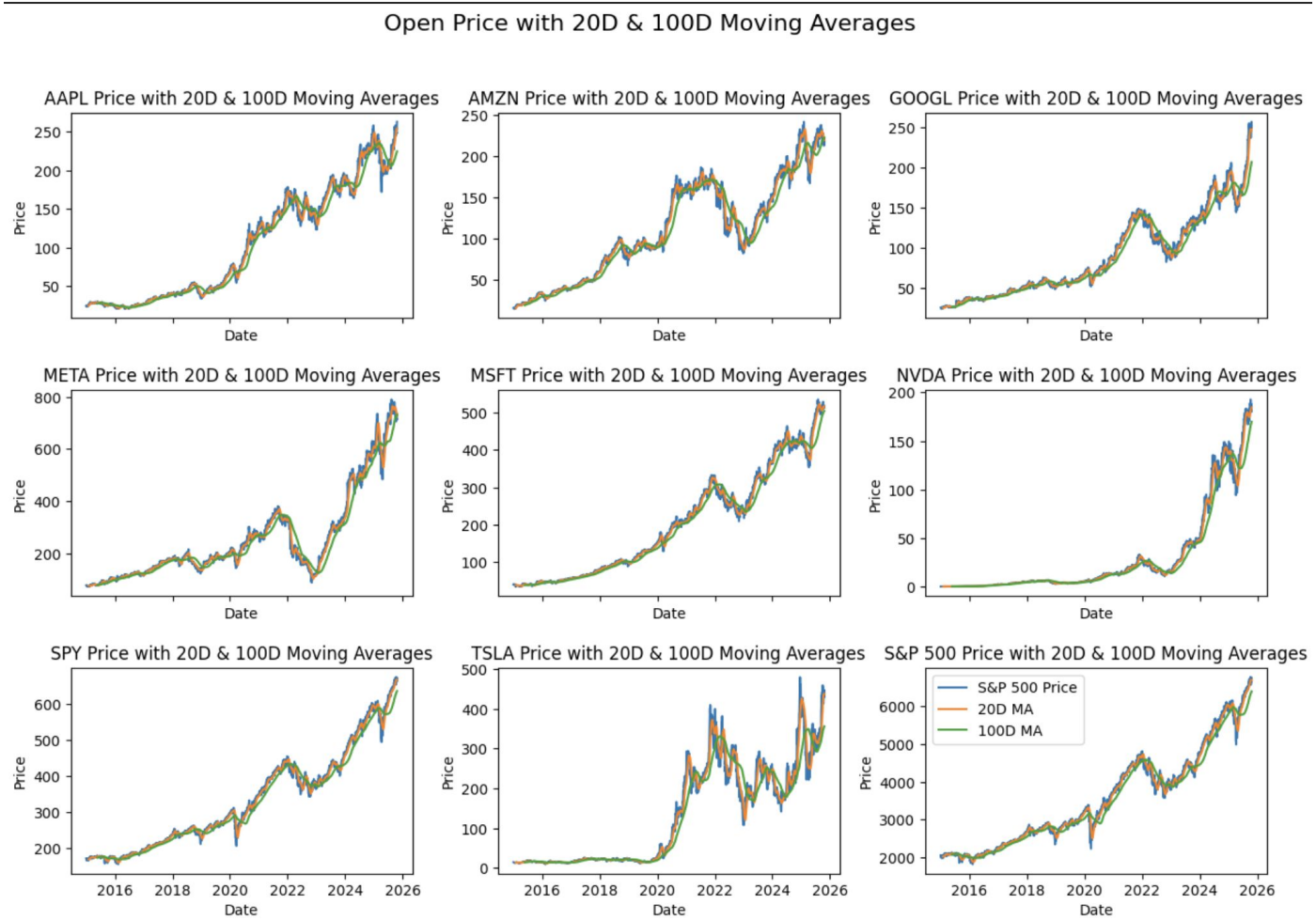
- Source: Kaggle (financial news dataset)
- Extracted features by Finbert from Prosus AI:
  - Average sentiment scores (positive / negative / neutral)
  - Number of headlines per day
  - Number of strongly positive headlines
  - Number of strongly negative headlines

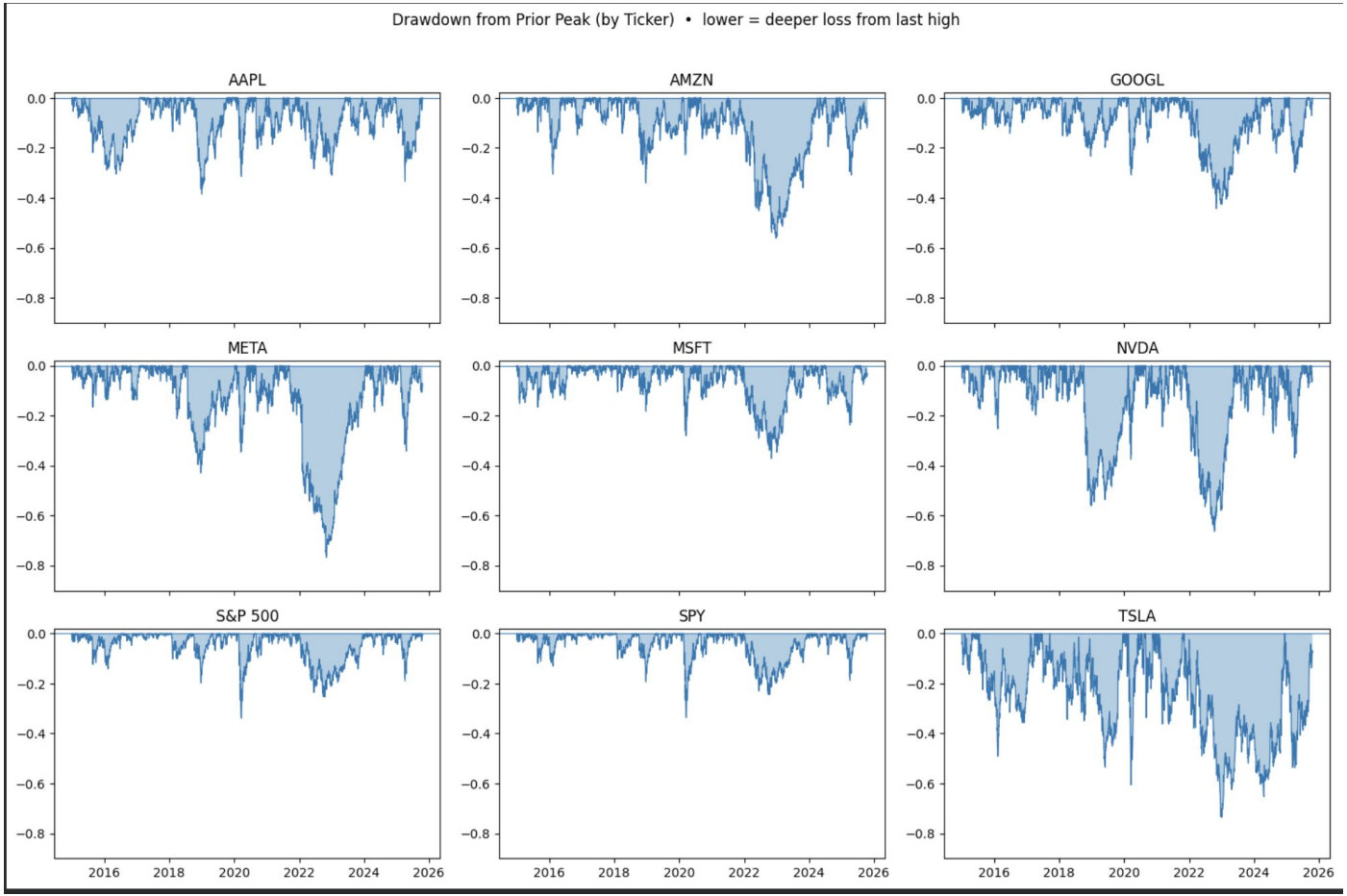
Data Overview and Preparation

1. Dataset Overview: yfinance & Kaggle
2. Preprocessing: performed data cleaning and categorization
3. Visualization: Creating stock performance metrics

Data Visualization

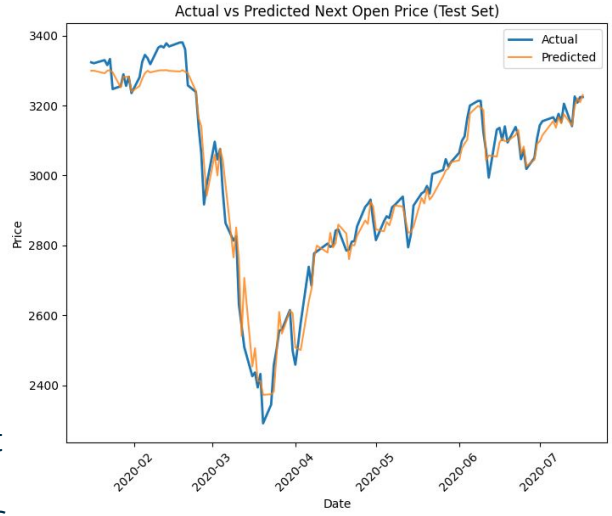


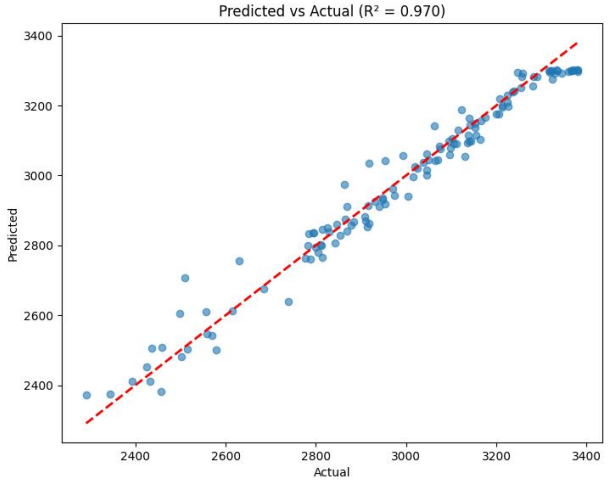

Open Price with 20D & 100D Moving Averages
AAPL Price with 20D & 100D Moving Averages
AMZN Price with 20D & 100D Moving Averages
GOOGL Price with 20D & 100D Moving Averages
META Price with 20D & 100D Moving Averages
MSFT Price with 20D & 100D Moving Averages
NVDA Price with 20D & 100D Moving Averages
SPY Price with 20D & 100D Moving Averages
TSLA Price with 20D & 100D Moving Averages
S&P 500 Price with 20D & 100D Moving Averages
S&P 500 Price
20D MA
100D MA
Price
Date
2016
2018
2020
2022
2024
2026

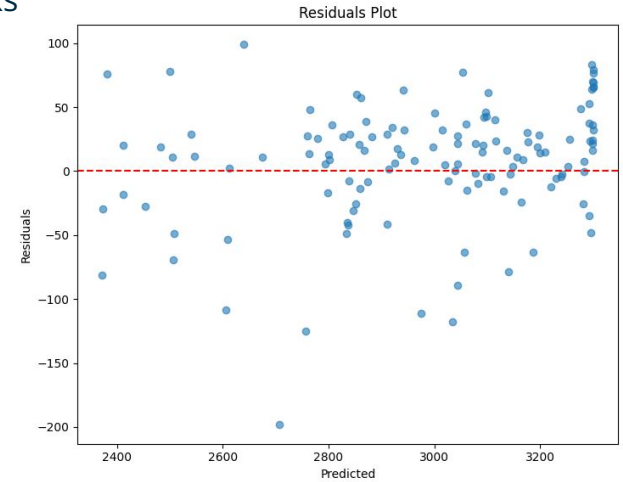

Drawdown from Prior Peak (by Ticker) • lower = deeper loss from last high
AAPL
AMZN
GOOGL
META
MSFT
NVDA
S&P 500
SPY
TSLA
0.0
−0.2
−0.4
−0.6
−0.8
2016
2018
2020
2022
2024
2026

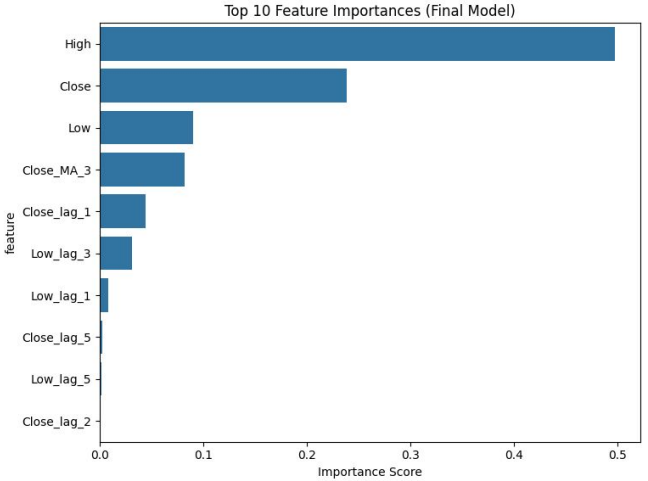
S&P 500 Prediction Choice

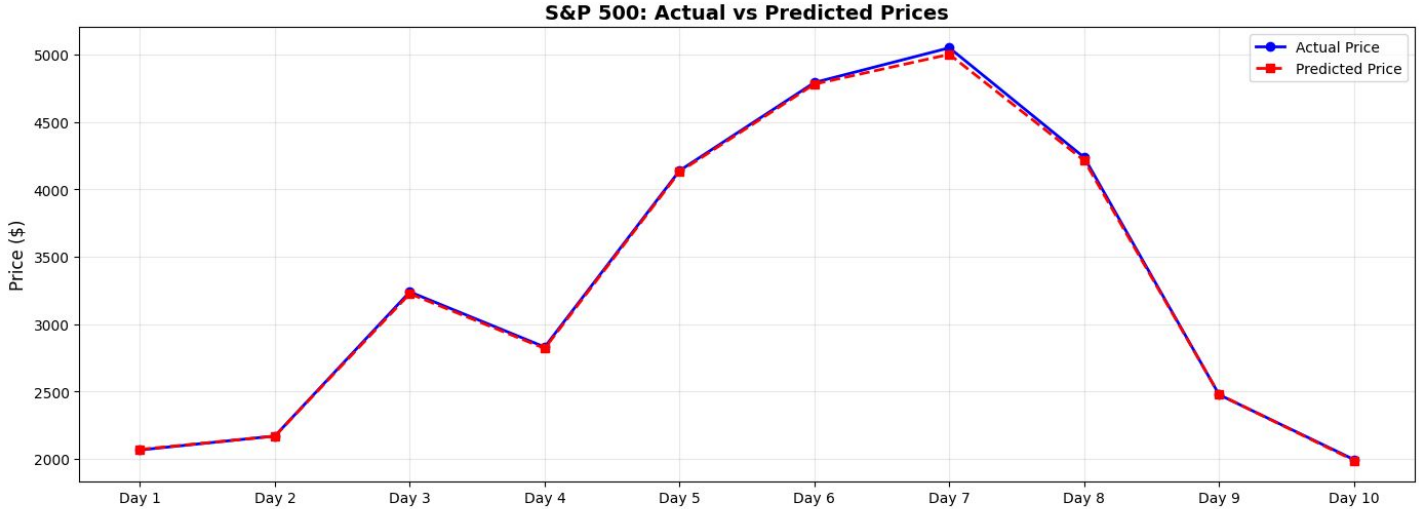
- Represents the overall market
- Less volatile than single stocks → more stable for modeling
- Strong availability of historic price data and related news coverage

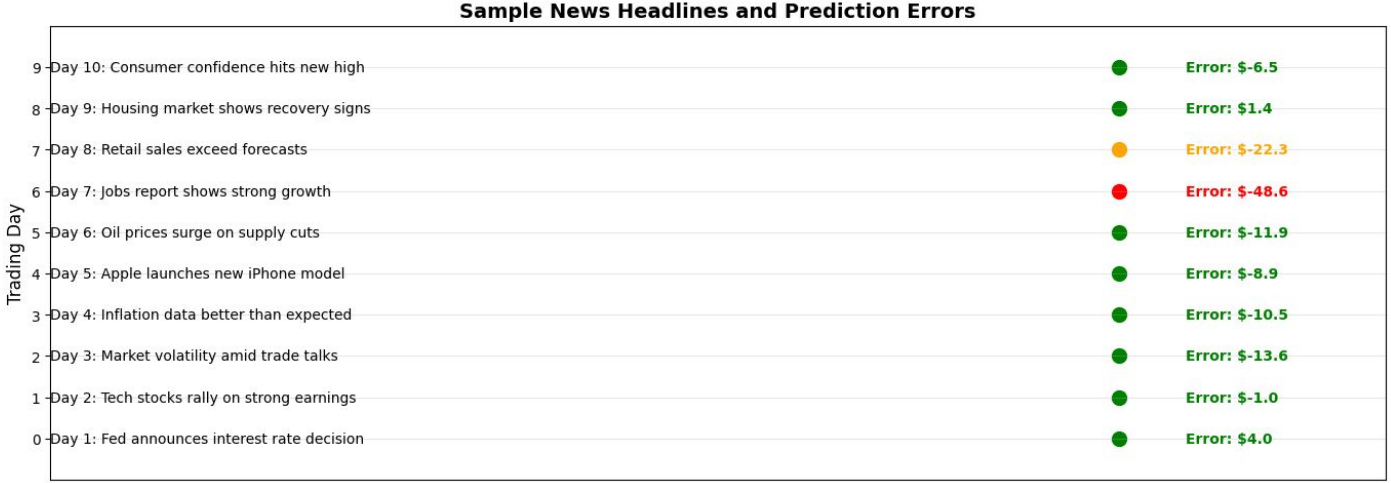









S&P 500: Actual vs Predicted Prices
Actual Price
Predicted Price
Price ($)
5000
4500
4000
3500
3000
2500
2000
Day 1
Day 2
Day 3
Day 4
Day 5
Day 6
Day 7
Day 8
Day 9
Day 10

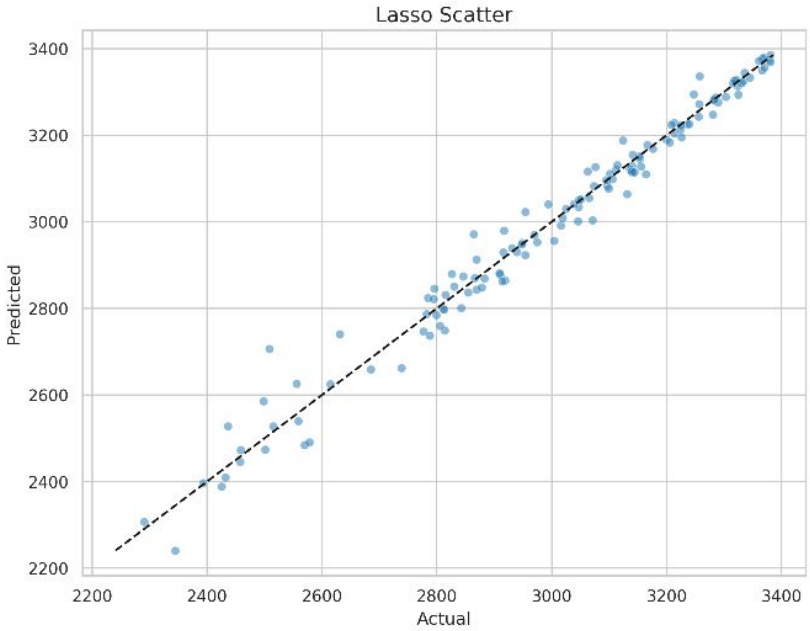

Sample News Headlines and Prediction Errors
Trading Day
9 Day 10: Consumer confidence hits new high
Error: $-6.5
8 Day 9: Housing market shows recovery signs
Error: $1.4
7 Day 8: Retail sales exceed forecasts
Error: $-22.3
6 Day 7: Jobs report shows strong growth
Error: $-48.6
5 Day 6: Oil prices surge on supply cuts
Error: $-11.9
4 Day 5: Apple launches new iPhone model
Error: $-8.9
3 Day 4: Inflation data better than expected
Error: $-10.5
2 Day 3: Market volatility amid trade talks
Error: $-13.6
1 Day 2: Tech stocks rally on strong earnings
Error: $-1.0
0 Day 1: Fed announces interest rate decision
Error: $4.0

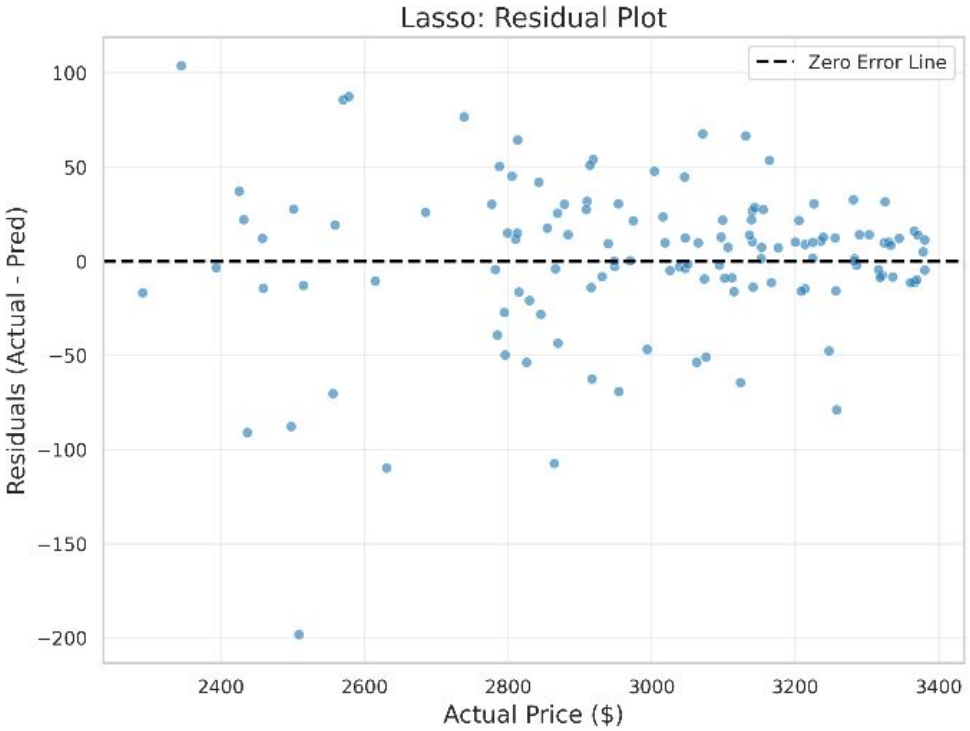
Modeling and Evaluation

Models Tested:

1. Lasso
2. Random Forest
3. XGBoost - Final model used

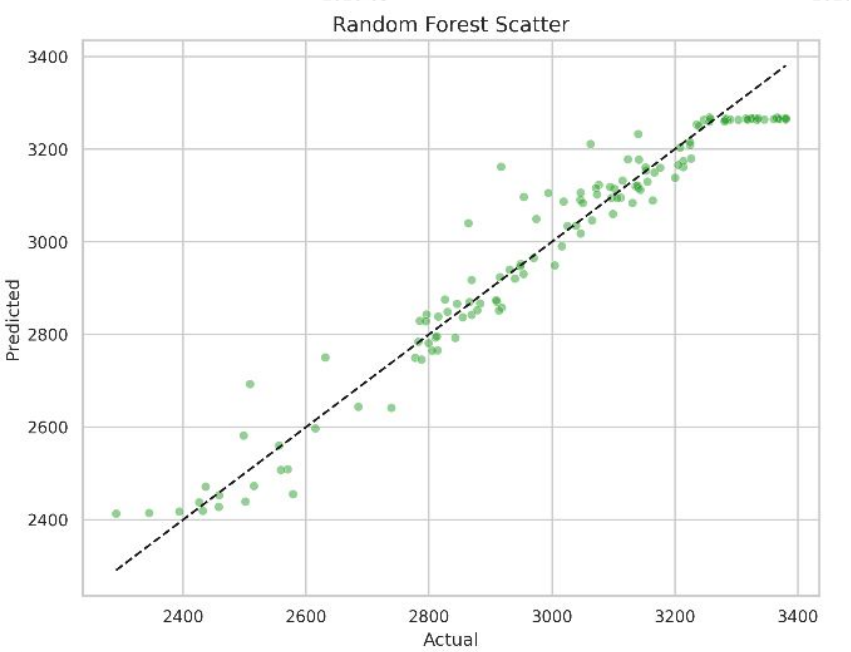
Lasso

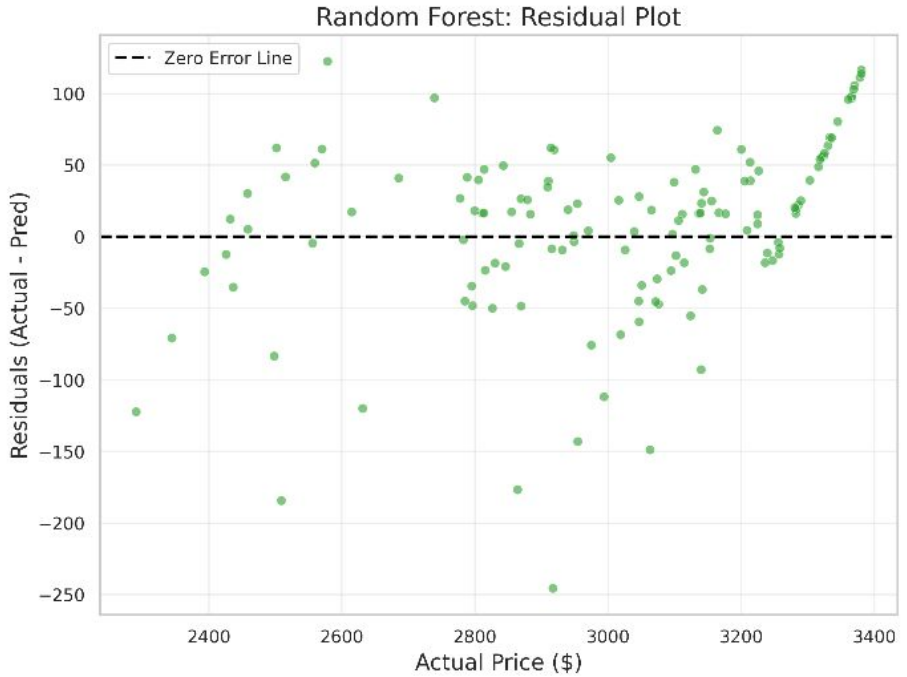




- Lasso regression shows a strong fit: predicted prices closely track actual prices along the 45° reference line.
- Residuals are mostly centered around zero with no clear trend, indicating low systematic bias.
- A few larger residuals suggest mild outliers, but they do not significantly affect overall model performance.

Random Forest





- Random Forest shows a generally strong fit, but predictions are slightly less tight around the 45° line than Lasso.
- Residuals fan out at higher prices and become mostly positive, indicating the model tends to underpredict the most expensive observations.
- The wider spread of residuals suggests Random Forest captures some non-linear patterns but still struggles with the upper end of the price range.

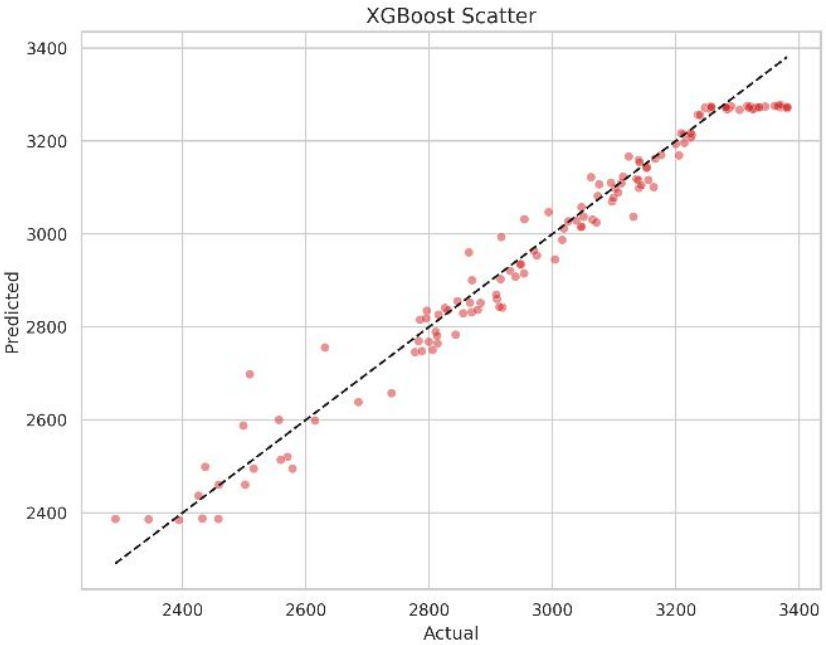
Model Comparison

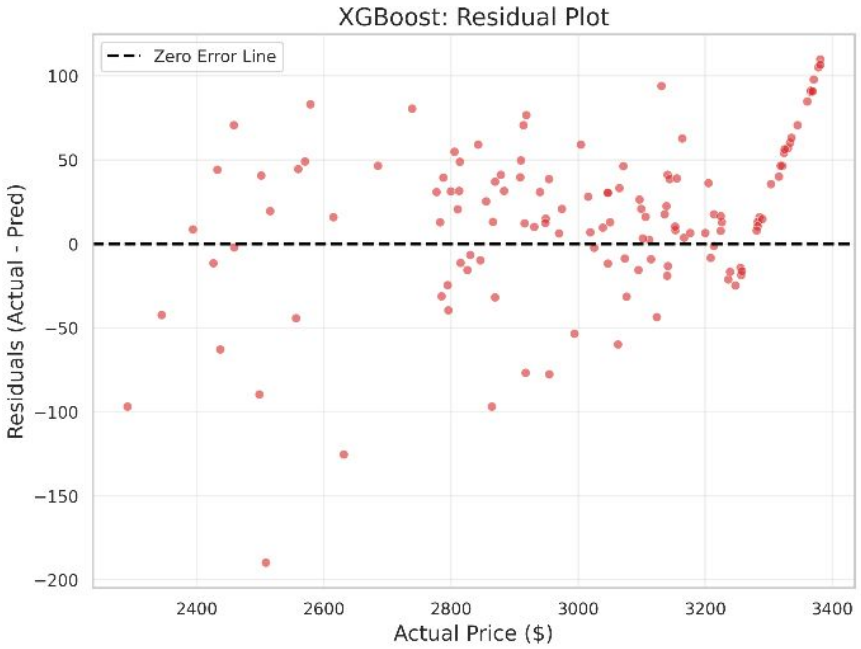
Model Name	Description	Results	Pros	Cons
Random Forest	Ensemble of many decision trees; captures non-linear patterns	Captured trend direction moderately well. Tended to underperform around volatile periods	Stable baseline, handles non-linear data, provides feature importance	Weaker than boosting models, smooth predictions, no time-dependency modeling
Lasso	Linear model with L1 penalty for feature selection	Strong linear fit: predicted prices track closely along the 45° line. Slightly wider spread at higher prices, but still stable.	Simple, easy to explain feature effects, fast to train and test	Cannot capture nonlinear relationships. Underperforms when price behavior depends on interactions or nonlinear market patterns.
XG Boost	A gradient-boosted decision tree model that builds trees sequentially, focusing on reducing previous errors. Widely used in financial ML due to its power and flexibility.	Best overall performance of all three models. Residuals show a mild underprediction for high-prices, but less severe than Random Forest.	Most accurate model in your experiment. Captures complex nonlinear price patterns well.	Less interpretable without SHAP. Slight over- and under-estimation in extreme price ranges.

Demo for all the models we tested

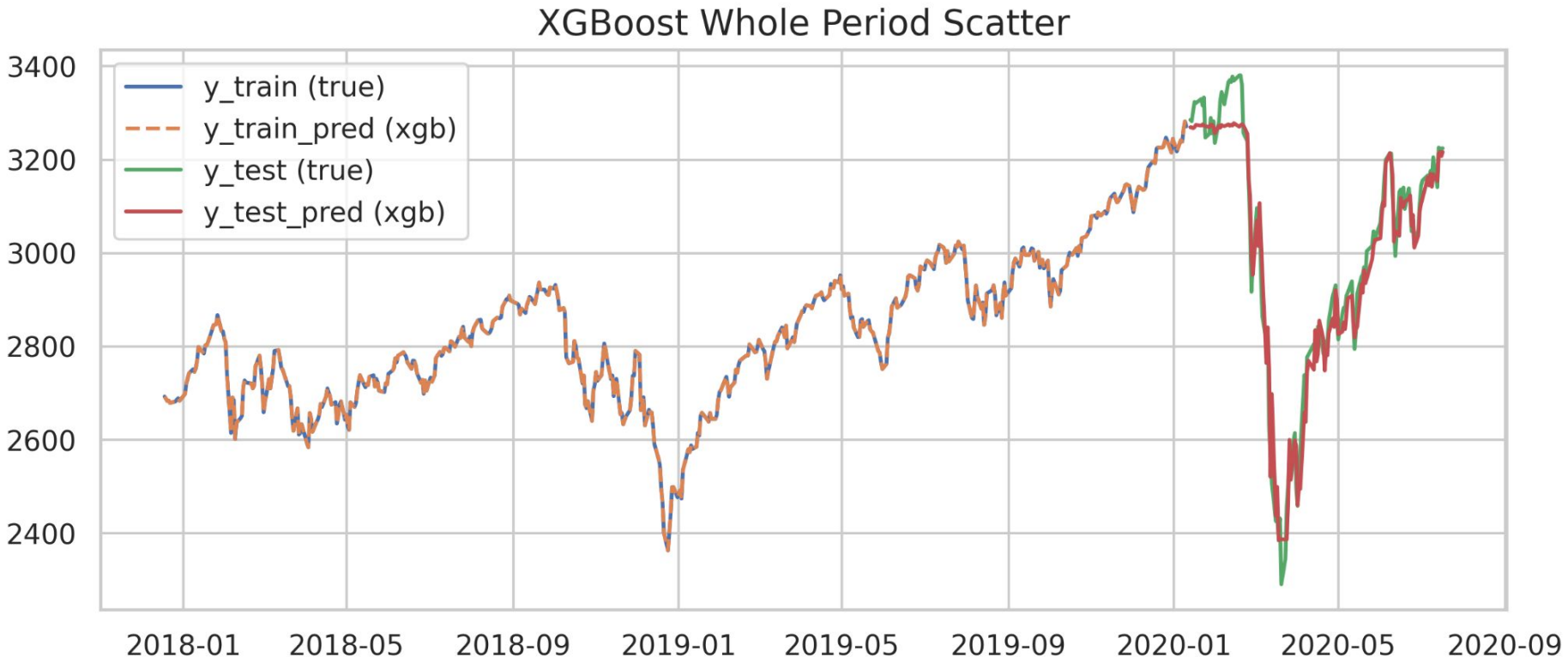
Trading Profit Website

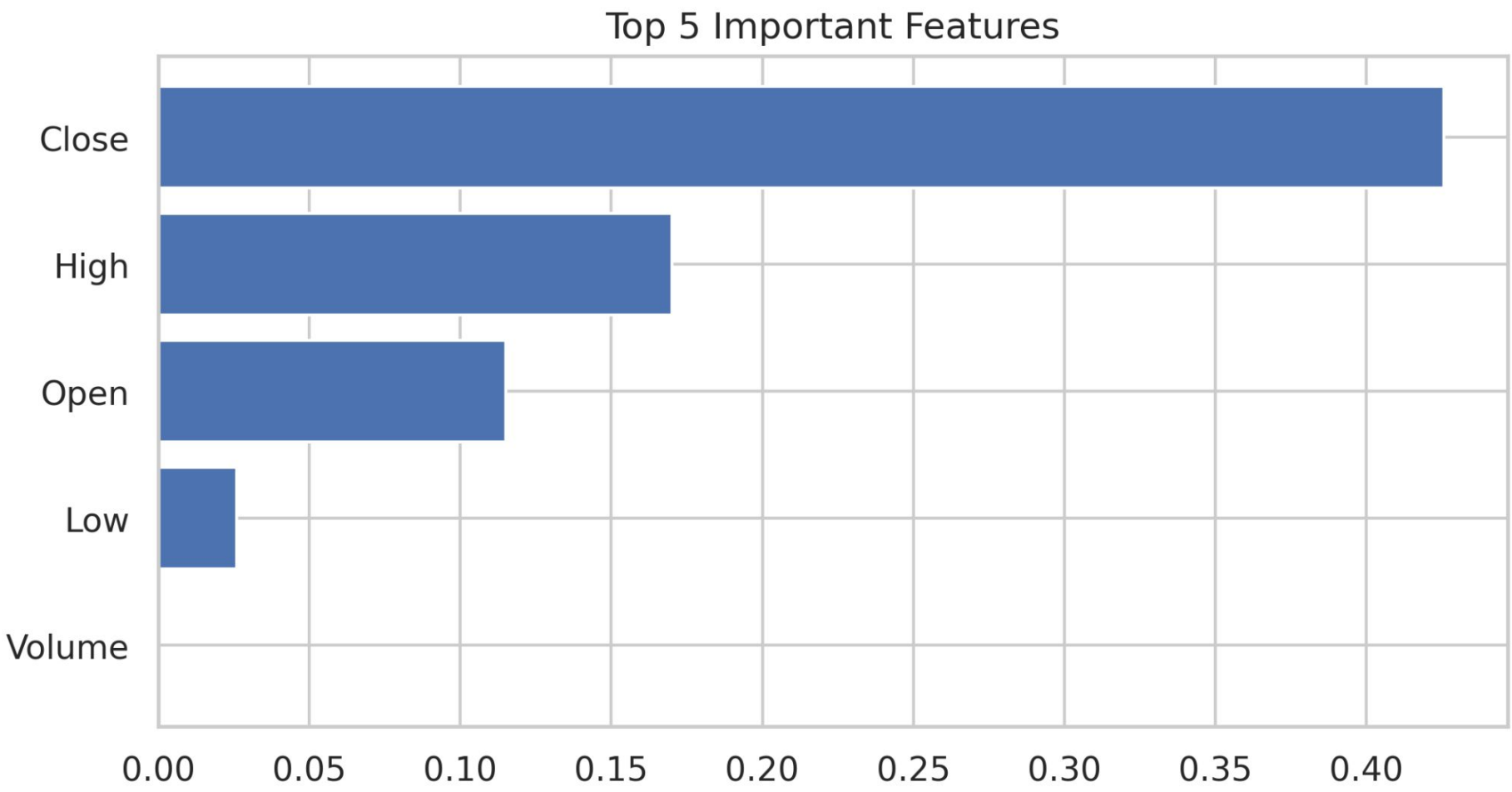
XGBoost



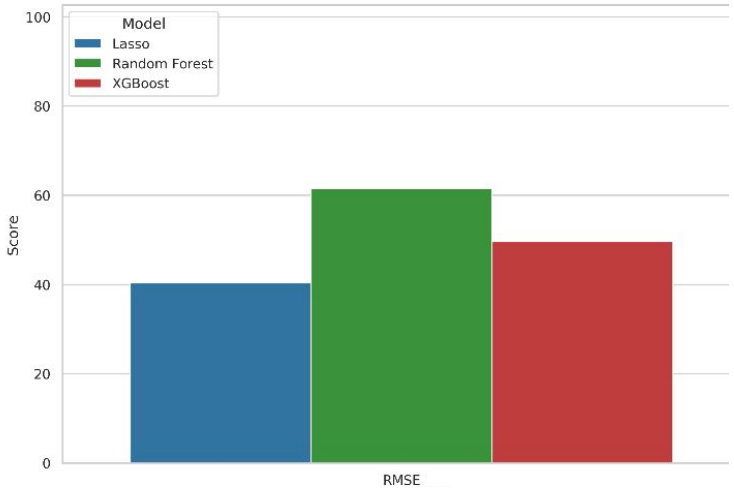


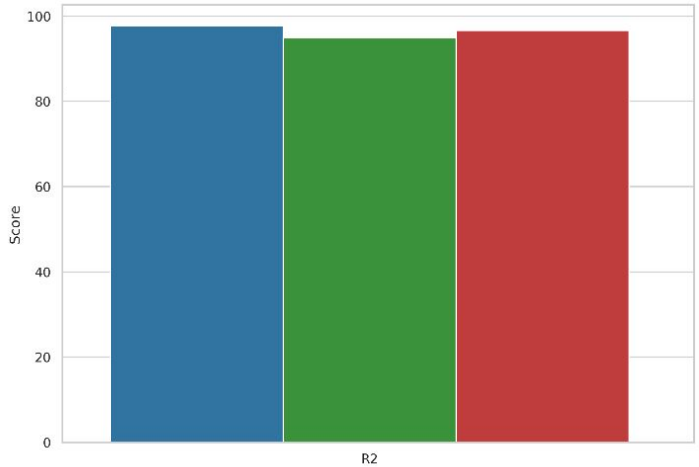
- XGBoost achieves a strong overall fit: most points lie close to the 45° line, indicating accurate price predictions.
- Residuals are centered around zero for most of the range but fan out at higher prices and become mostly positive, showing that XGBoost tends to underpredict the most expensive cases.
- A few mid-range observations show large negative residuals, suggesting occasional overreaction to local patterns and the presence of some outliers

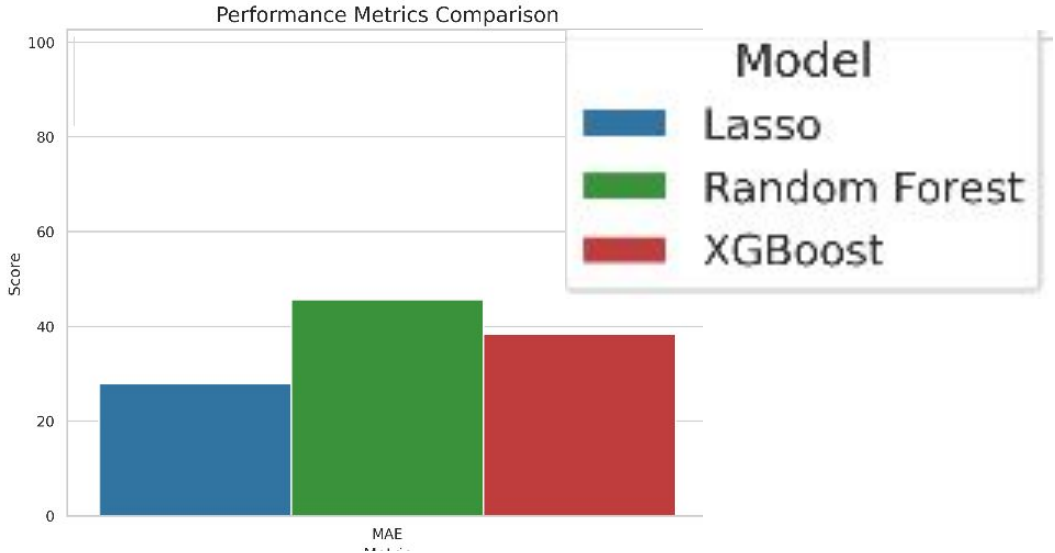

XGBoost Whole Period Scatter
y_train (true)
y_train_pred (xgb)
y_test (true)
y_test_pred (xgb)
3400
3200
3000
2800
2600
2400
2018-01
2018-05
2018-09
2019-01
2019-05
2019-09
2020-01
2020-05
2020-09


Top 5 Important Features
Close
High
Open
Low
Volume
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40

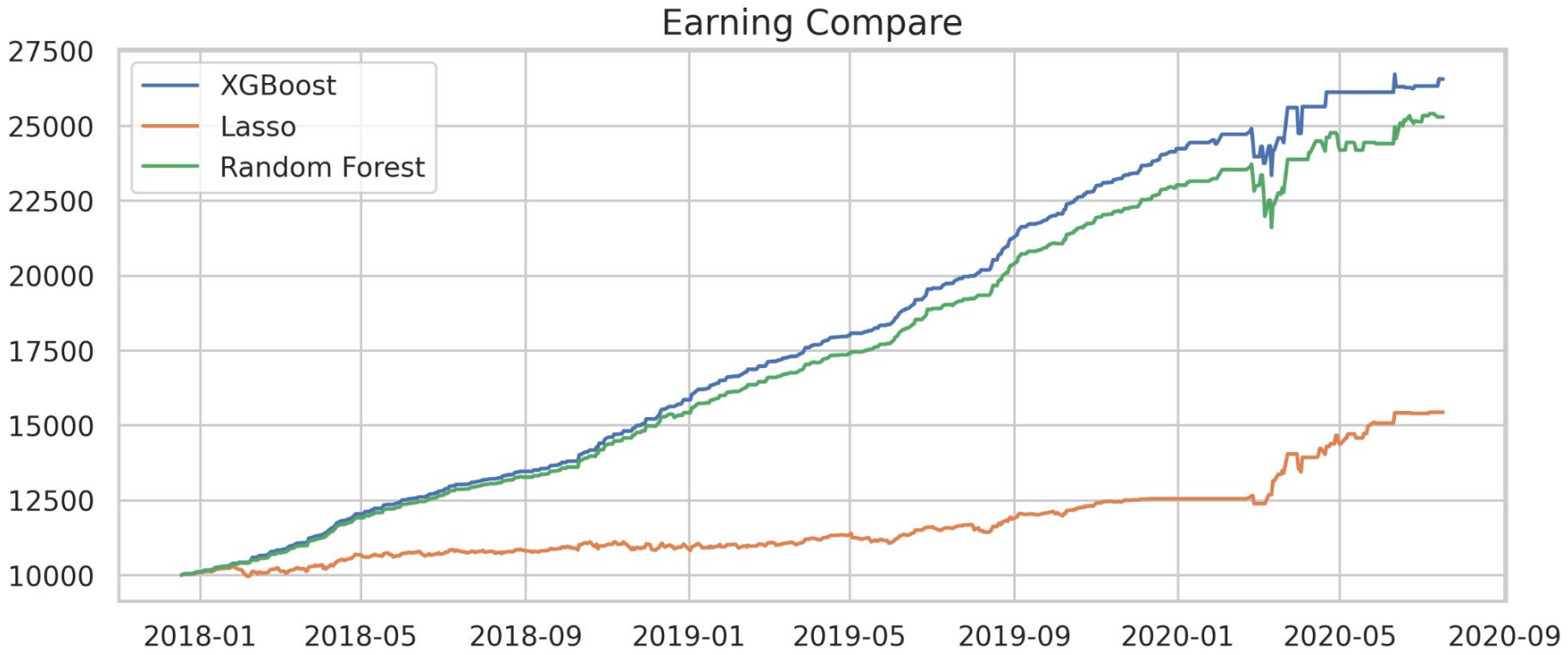
- SHAP results show that XGBoost relies most heavily on recent price information: *Close* and *High* (plus short-horizon rolling close) have the largest and most variable impact on predictions.
- High values of *Close* and *High* generally push predictions upward (red points on the positive SHAP side), while low values push them downward, matching financial intuition.
- Volume- and news-related features (headlines, positive sentiment) contribute only modestly, indicating that XGBoost mainly uses price and short-term trend signals, with limited incremental value from text features.

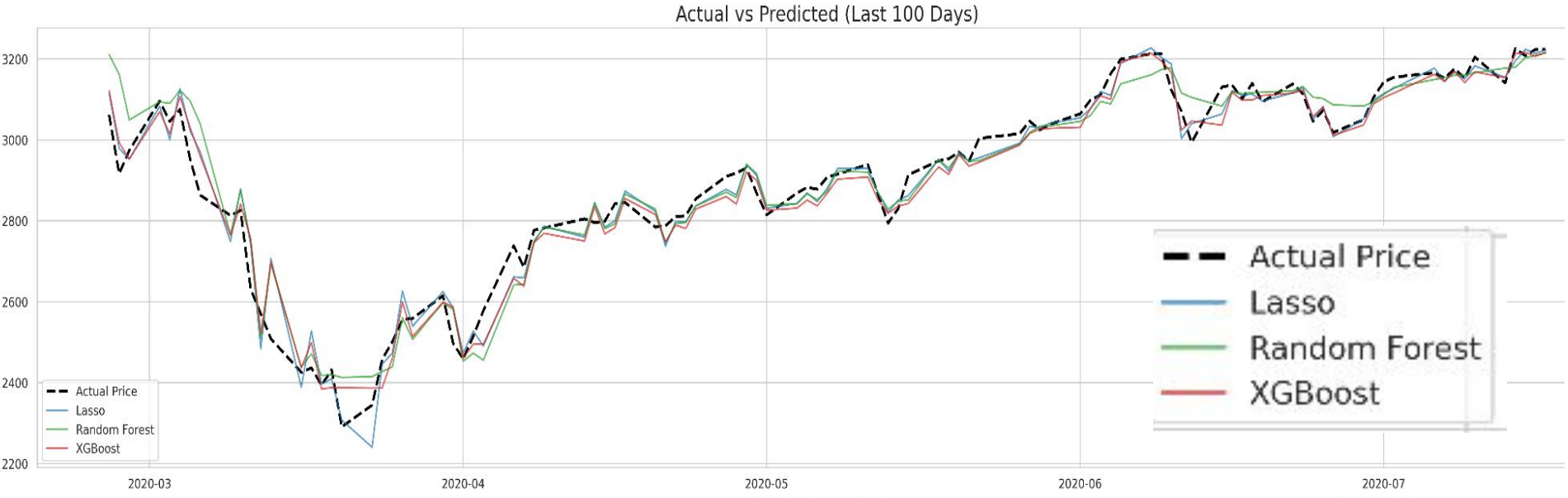






- Across all error metrics, Lasso performs best: it has the lowest RMSE and MAE and the highest R^2, indicating the most accurate and stable predictions
- XGBoost ranks second with slightly larger errors and a slightly lower R^2 than Lasso, but still clearly better than Random Forest.
- Random Forest shows the highest RMSE/MAE and the lowest R^2, suggesting it fits the data less well than the other two models in this setting.


Earning Compare
XGBoost
Lasso
Random Forest
27500
25000
22500
20000
17500
15000
12500
10000
2018-01
2018-05
2018-09
2019-01
2019-05
2019-09
2020-01
2020-05
2020-09



- All three models closely track the actual price over the last 100 days, capturing both the downward dip in March and the subsequent steady upward trend.
- The Lasso and XGBoost lines tend to stay closest to the dashed actual series, while Random Forest shows slightly larger deviations at some local peaks and troughs.
- Short-term fluctuations are generally well reproduced, with only small lags or under/over-shoots during the sharpest moves.

Final Model (XGBoost) Demo

Trading Profit Website

Final Thoughts

Potential Next Steps

- **Possible data leakage**
  We use day *t*'s close to predict day *t+1*'s open. Since these values are usually very similar, the model can "cheat" by relying heavily on Close, which SHAP confirms. This limits the model's true predictive power.

- **Weak impact of news features**
  News features contribute almost nothing, SHAP values are near zero. To make news meaningful, we may need a separate news model or a hybrid approach so news signals actually influence predictions.

- **Getting richer signals from text news**
  We currently use simple sentiment averages. Future work could apply NLP methods—topic extraction, event detection, or text embeddings—to provide richer information than just positive vs. negative counts.

What We Learned

- Technical skills: preprocessing, data cleaning, model fine-tuning, sentiment analysis integration, model evaluation
- Professional Skills: teamwork, project management, time management

Acknowledgements: Leah and Atena

Thank you!

Questions?