

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Stock Market Prediction Using Python

¹ Muskan Soni, ² Purva Gupta, ³ Nikhil Dhankar, ⁴ Priyasha Choudhary

¹²³⁴ Department of Computer Science and Engineering Shri Shankarachharya Technical Campus, Bhilai, Chhattisgarh, India Emails: muskansoni2301@gmail.com , pgupta91655@gmail.com , nikhildhankar85@gmail.com , priyasha611@gmail.com

1. ABSTRACT :

The project focuses on predicting stock market trends using advanced data analytics and machine learning tools in Python. By utilizing libraries such as Plotly for visualization, Prophet for time series forecasting, and YFinance for financial data extraction, this project aims to provide a user-friendly interface for analyzing stock trends. The model is trained on historical data to predict future stock prices, which can assist traders and investors in making informed decisions. The combination of open-source tools and statistical modeling delivers a robust approach to understanding market behavior.

2. Introduction

Stock market prediction has long been a challenging problem due to its volatile and dynamic nature. With the rise of machine learning and big data, there are now innovative methods to analyze vast amounts of financial data for making reliable forecasts. This project leverages the power of Python and its data science libraries to analyze historical stock prices and generate predictive models. The objective is to implement a system that can predict short-term trends and visualize stock movement, aiding users in interpreting market conditions.

3. Literature Review

Several studies have explored different methodologies for stock prediction including linear regression, neural networks, ARIMA models, and ensemble learning techniques. Facebook's Prophet model has emerged as a prominent tool for time series forecasting due to its interpretability and ease of use. Previous research shows that integrating multiple data sources and visual analytics enhances the quality of predictions. Our project builds upon these foundations by using Prophet in conjunction with real-time data fetched via YFinance and visualizing trends with Plotly.

4. Methodology

The methodology follows a systematic approach starting from data collection using YFinance, preprocessing the data for missing values and outliers, followed by implementing Prophet to forecast stock prices. The processed data is split into training and testing datasets. The model is then trained using Prophet, which considers trends, seasonality, and holidays. Results are visualized using interactive Plotly graphs for better understanding. Finally, the model's performance is evaluated using metrics such as RMSE (Root Mean Squared Error.



Stock market prediction methodology

5. Libraries Details

YFinance: Used to download historical market data directly from Yahoo Finance.
Prophet: Developed by Facebook, it is used for forecasting time series data based on additive models.
Plotly: An interactive graphing library used to create high-quality charts and plots.
Pandas/Numpy: For data manipulation and numerical computation.
Matplotlib/Seaborn: For basic plotting and exploratory data analysis during preprocessing stages.

6. Purpose

The purpose of this project is to build a reliable and easy-to-use system that predicts stock prices and helps users interpret financial trends through visual representation. The system aims to reduce the uncertainty involved in investment decisions by offering data-driven insights.

7. Result and evaluation

.

.

• The trained Prophet model was able to predict short-term stock trends with reasonable accuracy. The evaluation metrics showed a low RMSE, indicating good predictive performance. Visual plots generated by Plotly clearly represented the predicted trends alongside actual stock movements, providing intuitive feedback to users.

	Years of predictio	n:							
	1								
	Loading data	donel							
	Raw data	a							
	2,458 2024 1	00:00:00	164.855	166.26	161.12	163.06	163.06 1	9,666,40	0
	2,459 2024-1	0-11 00:00:00	163.33	165.27	162.5	164.52	164.52 1	0,946,00	0
	2,461 2024-1	0-14 00:00:00	164.91	167.62	164.78	166.35	166.35	9,981,80	ю
	2,462 2024-1	0-15 00:00:00	167.14	169.09	166.05	166.9	166.9 1	4,814,50	ю
re	cast data								
		trend	yhat_lower	yhat.	_upper	trend_lower	trend_up	oper a	dditive_terms
823	2025-10-11 00:00:00	221.321	201.8725	2	40.746	202.937	240.4	852	-0.4337
324	2025-10-12 00:00:00	221.4409	201.805	24	0.3741	202.9908	240	.705	-0.3611
325	2025-10-13 00:00:00	221.5608	201.5587	24	1.6542	203.0445	240.9	831	-0.7386
326	2025-10-14 00:00:00	221.6807	201.5988	24	1.4175	203.0993	241.1	.787	-0.6917
827	2025-10-15 00:00:00	221.8007	200.9774	- 24	1.5926	203.1543	241.3	1581	-0.6434
ecas	t plot for 1 years								
1m 250	6m ly all								
							-74		
50									
			2018	2	1020	2022		202	4



8. Discussion

While the model performs well in predicting general trends, it may not always capture sudden market movements caused by external events like economic shifts or company-specific news. Enhancing the model with sentiment analysis and additional features like trading volume or macroeconomic indicators could further improve accuracy. Nonetheless, the use of Prophet simplifies time series forecasting and makes it accessible for financial analysis.

9. Conclusion

The project successfully demonstrates the use of Python and its libraries for stock market prediction. With a clean data pipeline, robust forecasting model, and dynamic visualization, it offers a practical tool for stock analysis. This foundation can be expanded with more complex models or integrated into real-time trading platforms for further development.

10. REFERENCES :

- 1. Taylor, S. J., & Letham, B. (2018). Forecasting at scale. The American Statistician, 72(1), 37-45.
- 2. Yahoo Finance. (n.d.). Retrieved from https://finance.yahoo.com
- 3. Plotly documentation. https://plotly.com/python/

4. Prophet documentation. https://facebook.github.io/prophet/docs/5. McKinney, W. (2010). Data structures for statistical computing in Python. Proceedings of the 9th Python in Science Conference, 51–56.