Time Series Analysis For Bitcoin Price Prediction Using Prophet

Team 271

Smartbridge - Artificial Intelligence

PROJECT REPORT

by

Pranav Vyas - 20BCT0279
Madhusmita Mukherjee - 20BCE2783
Devanshi Jajodia - 20BCE2749

Campus - VIT Vellore

June 2023

2. TABLE OF CONTENTS

3.Introduction

3.1 Overview

Bitcoin, which was established in January 2009, is a cryptocurrency that holds great value and can be traded on numerous global exchanges. With its high volatility, it presents a unique opportunity for price prediction, offering valuable insights for investors, traders, and researchers. Our project aims to work on the prediction system for Bitcoin using Prophet, which is a pre-trained model, to predict the price. There are various factors affecting the price of Bitcoin. A Prophet model is built that helps to define the price trend of Bitcoin in the future.Our project aims to predict future Bitcoin prices by applying time series analysis techniques using Facebook's Prophet forecasting model. Time series analysis involves studying historical data to identify patterns, trends, and seasonality. We will gather reliable historical price data for Bitcoin from sources like cryptocurrency exchanges or financial data providers, including dates and corresponding Bitcoin prices. To construct a predictive model, we will utilize Facebook's Prophet library, designed specifically for time series forecasting. Prophet incorporates different elements like trend, seasonality, and holiday effects to capture the underlying patterns in the data. The project will be divided into several steps:

- Data Collection
- Data Preprocessing
- Model Building
- Model Evaluation
- Future Price Prediction
- Visualization

Our project will end by thoroughly examining how well the model performs and its real-world implications for forecasting Bitcoin prices. It is essential to understand that although Prophet has shown success in different tasks of predicting time series data, it is necessary to consider additional factors and conduct additional analysis before relying solely on the model's predictions when making investment or trading choices.

3.2 Purpose

This project aims to use time series analysis and the Prophet forecasting model to predict future price trends of Bitcoin. It will analyze historical price data and consider factors like trends, seasonality, and holidays to provide useful information for investors, traders, and researchers in the cryptocurrency market. The Prophet model can provide insights into potential future price movements. This information can be valuable for investors, traders, and researchers looking to make informed decisions regarding Bitcoin. Investment Strategies and accurate price predictions can assist in formulating effective investment strategies. Investors can use the forecasted prices to determine optimal entry and exit points for buying or selling Bitcoin. Another purpose of this project is risk management. Price predictions can help in managing risks associated with Bitcoin investments. This can help protect investments from significant losses and enhance overall risk management practices. By leveraging the capabilities of the Prophet model, it aims to provide reliable forecasts that can assist investors, traders, and researchers in optimizing their strategies and maximizing their returns in the volatile cryptocurrency market. This project also helps in decision support as traders and individuals interested in cryptocurrency can utilize the forecasted prices as part of their decision-making process. By analyzing historical price data and comparing it with the forecasted prices, researchers can evaluate the accuracy of the Prophet model and explore potential correlations or patterns in Bitcoin's price dynamics. The use of prophet model eliminates the need for manual feature engineering and complex model selection, allowing users to focus on interpreting results and making decisions based on the forecasts. This automation has saved time and effort, making the forecasting process more efficient. Prophet is designed to handle large-scale time series data efficiently. It utilizes parallelization and other optimizations to handle thousands of time series and millions of observations. This scalability makes it suitable for applications where there is a need to forecast a large volume of data. Users can specify additional regressors, define custom seasonality patterns, and specify changepoints to capture specific events or patterns in the data. This flexibility enables users to tailor the model to their specific forecasting needs. Finally this project compares the predicted prices with the actual prices, and hence the accuracy and effectiveness of the model can be assessed.

4. Literature Survey

[1]. This paper explores the application of the Prophet forecasting model for time series analysis and prediction in the business economy. The paper begins by introducing the Prophet model, highlighting its flexibility and ease of use. They explain how the model decomposes time series data into trend, seasonality, and holiday effects, which allows for a better understanding of the underlying patterns. The authors also discuss how Prophet handles missing data and outliers, employing interpolation techniques for data imputation and robust statistical methods to minimize the impact of anomalies. Furthermore, the paper delves into the automatic detection of changepoints, which are significant shifts in the time series data. The authors explain how the Prophet identifies these changepoints, providing a means to adapt the forecasts accordingly. They also emphasize the model's ability to incorporate custom seasonality patterns, enabling the capture of unique seasonal variations specific to the business domain under consideration. The paper further highlights the uncertainty estimation feature of the Prophet model, which allows for the quantification of confidence intervals around the forecasted prices. The authors discuss the scalability of Prophet, highlighting its efficient handling of large-scale datasets through parallel processing and distributed computing. Overall, the paper provides a comprehensive overview of implementing Prophet forecasting model to time series analysis in the business economy.

[2] The authors present their methodology for forecasting stock prices and compare the performance of the Prophet model with and without incorporating ARIMA. The paper begins by explaining the limitations of traditional time-series forecasting models like ARIMs. They introduce the Prophet model, highlighting its ability to capture trend, seasonality, and changepoints. They then describe how they combine Prophet with ARIMA to leverage the strengths of both models. The authors outline the steps of their methodology like data preprocessing, model fitting using ARIMA, generating residuals, and using these residuals as an input to the Prophet model. They also discuss the process of parameter tuning and evaluation metrics used to assess the performance. The paper presents the experimental results of applying the proposed methodology to stock market data, comparing the accuracy of Prophet alone versus the combined Prophet-ARIMA model. The authors analyze the forecasting results, including mean absolute error (MAE) and mean squared error (MSE), to evaluate the effectiveness of the combined approach.

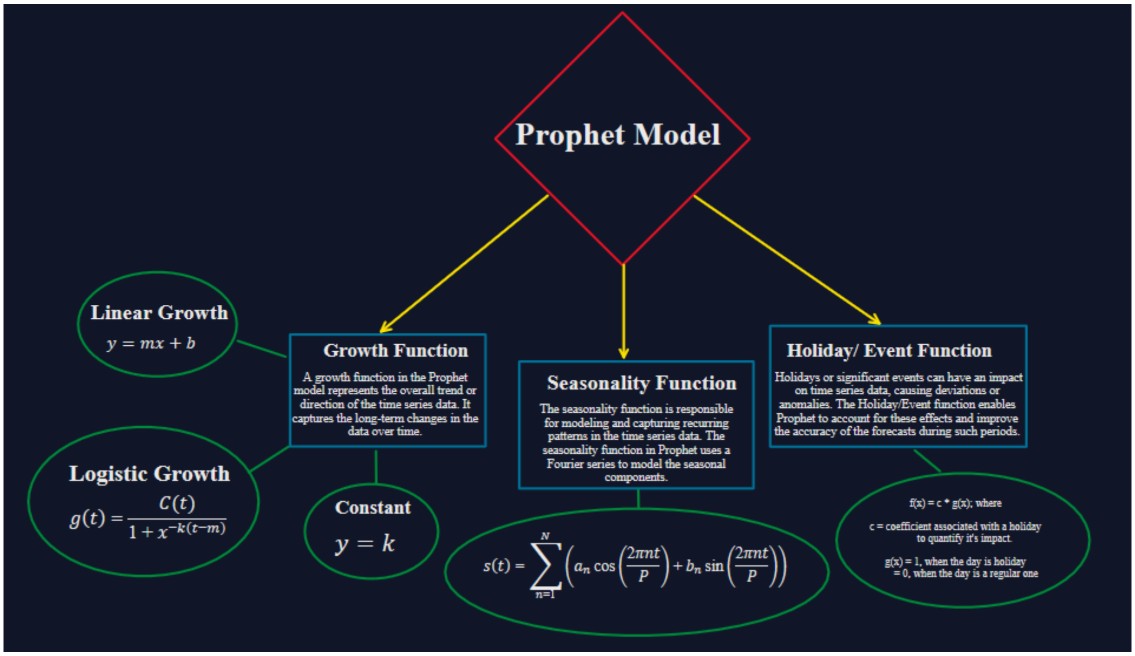
[3] This paper focuses on the application of Facebook's Prophet model for forecasting the price of Ethereum, a popular cryptocurrency. The methodology involves data collection of historical Ethereum price data, data preprocessing to handle missing values and outliers, and data transformation to the required format for Prophet. They then proceed to train the Prophet model using the transformed data, incorporating the model's time series analysis capabilities. Furthermore, the paper discusses the process of generating future predictions using the trained model and visualizing the forecasted values along with uncertainty intervals. They emphasize the importance of evaluating the forecast accuracy by comparing the predicted prices with the actual prices. The authors present the results of their experiments, evaluating the performance of the Prophet model for Ethereum price forecasting. They analyze metrics such as mean absolute error (MAE) or root mean squared error (RMSE) to assess the effectiveness of the model.

[4] This paper focuses on predicting lithium mineral resource prices in China using two different methods: Facebook Prophet (Fb-P) and Artificial Neural Networks (ANN). The authors present the methodology employed in their study. They begin by collecting historical data on lithium prices in China, including relevant features and factors affecting the prices. They preprocess the data to handle any missing values or outliers, ensuring the quality and integrity of the dataset. Next, the paper describes the implementation of both the Fb-P and ANN models for price forecasting. They discuss how Fb-P leverages time series decomposition, trend modeling, and seasonality detection to capture patterns in the data. In contrast, ANN is a machine learning-based approach that utilizes interconnected nodes to learn and predict future prices. The authors evaluate the performance of both models by comparing the forecasted prices with the actual prices. They analyze metrics such as mean absolute percentage error (MAPE) and root mean squared error (RMSE) to assess the accuracy and effectiveness of the models.

[5] The contents of this paper describe predicting stock prices using the Prophet model with hyperparameter tuning. The authors present the methodology they implemented to enhance the forecasting accuracy of the Prophet model. The authors describe the methodology, which involves data collection of historical stock prices and preprocessing to handle missing values or outliers. They then proceed to tune the hyperparameters of the Prophet model, such as the changepoint prior scale, seasonality parameters, and trend flexibility, to achieve the best possible performance. Furthermore, the paper discusses the process of training the optimized Prophet model and generating future price predictions. They evaluate the accuracy of the forecasted values by comparing them to the actual stock prices. Overall, the paper provides insights into the methodology implemented for stock price prediction using the Prophet model with hyperparameter tuning. It emphasizes the significance of fine-tuning the model's parameters to enhance forecasting accuracy and showcases the potential of the Prophet model in the domain of stock market forecasting.

5.Theoretical Analysis

5.1. Block Diagram



(Fig1. Prophet Model Components)

5.2. Software / Environments/ Libraries used:

1. **Google Colab**: It is a cloud-based integrated development environment (IDE) that allows users to write, run, and collaborate on Python code and machine learning projects.
2. **Pycharm**: PyCharm is an integrated development environment (IDE) specifically designed for Python development. It offers features like Go to Definition, Find Usages, and Rename Refactoring, making it easier to understand and modify code. It offers comprehensive support for web development using Python, HTML, CSS, and JavaScript.
3. **Prophet Library**: The Prophet library is an open-source time series forecasting library developed by Facebook to simplify the process of forecasting time series data with a focus on simplicity, flexibility, and high-quality forecasts. It provides uncertainty estimation for the forecasts by utilizing a Bayesian framework.
4. **Yfinance**: Yfinance is a Python library that provides a simple and convenient way to access historical and real-time financial data from Yahoo Finance. It allows users to retrieve stock prices, historical market data, dividend information, and more, making it a valuable tool for financial analysis.
5. **Pickle**: This modules enables object serialization and deserialization for data storage and retrieval.
6. **Languages used**: Python 3.10 , HTML, CSS, JavaScript
7. **Pystan**: A Python interface to the Stan probabilistic programming language. It allows users to define and fit Bayesian statistical models using the Stan modeling language through Python.
8. **Numpy**: A fundamental library in Python for scientific computing and data manipulation.
9. **Pandas**: Open-source Python library used for data manipulation and analysis.
10. **Matplotlib**: Python library for creating visualizations and plots. It provides a comprehensive set of tools for generating various types of plots, charts, and figures.

Hardware requirements of the project:

- CPU: A modern CPU with multiple cores, for eg, Intel Core i5 or higher
- RAM: 8 to 16 Gb or higher is recommended
- GPU: NVIDIA GeForce GTX 960 or higher is recommended
- Storage: 64 Gb to 500 Gb
- Operating System: Linux, Windows or MacOS

6. Experimental Investigation

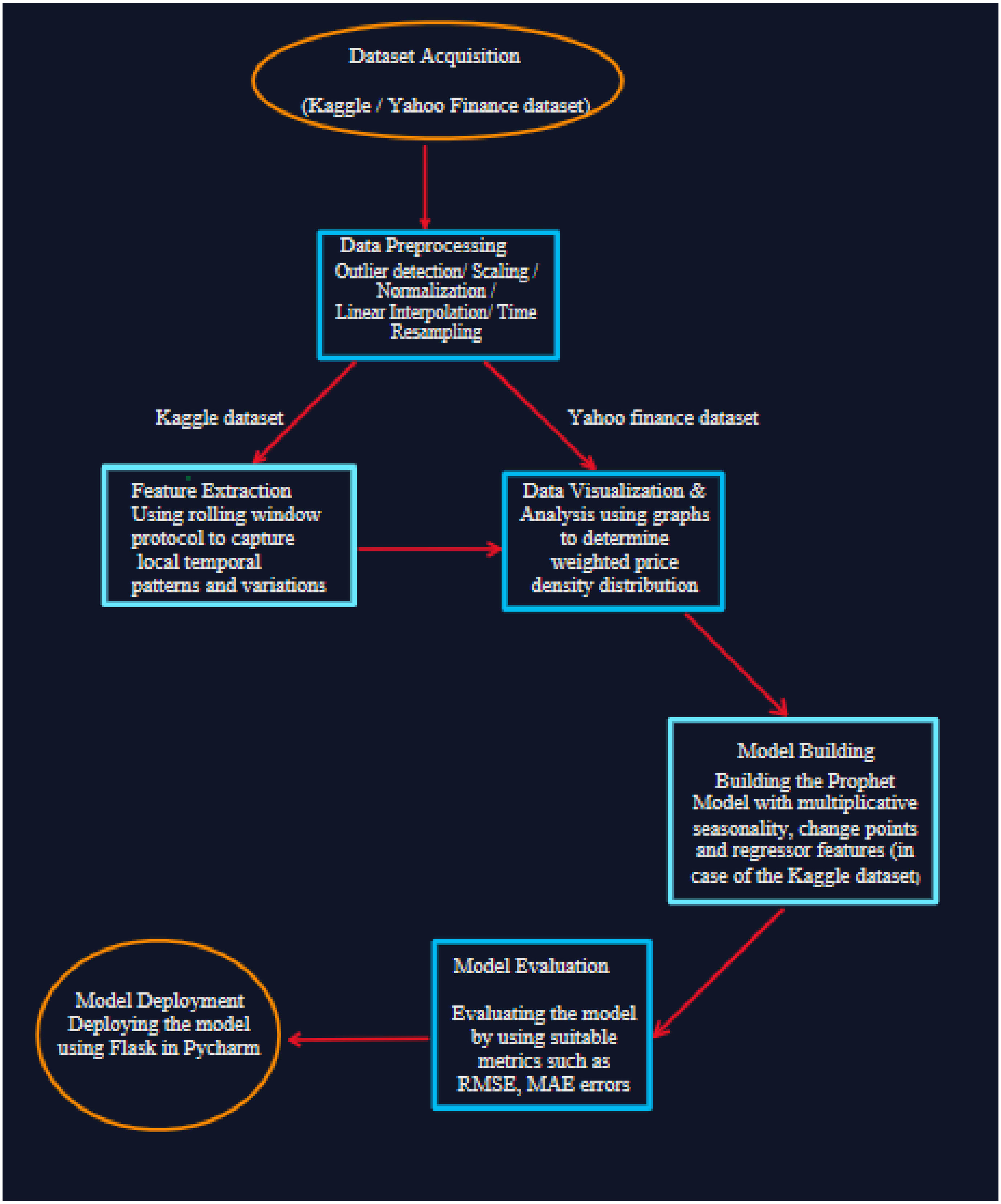
In this project, we perform Bitcoin Price Prediction using the Facebook Prophet Model. The dataset which we needed for performing this task was a time-series data with a sequence of data points denoting trading days and the corresponding weighted prices of bitcoins at the end of each training day. The first viable option was the Bitcoin Historical Dataset [6]. This dataset contains various feature variables such as open price (The opening price of Bitcoin at the start of trading day), high price (The highest price reached by Bitcoin during the trading day), low price (The lowest price reached by the Bitcoin at the end of the trading day), close price (The closing price of the Bitcoin at the end of the trading day) and the Weighted bitcoin price for multiple instances on a particular trading day. The timestamps for this dataset were in the Unix Time Format. Our first task was to convert the Unix Time Format to the standard ISO 8601 date format. We then performed the task of linear interpolation to handle missing values. Linear interpolation assumes a linear relationship between the data points and provides a simple and straightforward way to estimate values within the range of the known data. Time resampling was performed thereafter, to scale the dataset down to each trading day where the feature variables for each trading day were the aggregated results for multiple instances of that day. For analyzing time dependent patterns and data smoothing, we perform the task of feature extraction using the rolling windows protocol. The approach involves selecting a window size, moving it progressively over the time series, generally with an overlap. Different statistical calculations or transformations, such as mean, standard deviation, maximum, minimum, or more complicated operations like Fourier transforms or wavelet decompositions, can be carried out at each window point. We then split the data into training and validation sets and use the date and weighted price features along with the extracted features for training the prophet model. These extracted features act as regressors to the prophet model for better forecasting and predictions. The forecasted values are validated against the weighted prices to determine the mean absolute error and the root mean square error. A graphic visualization of the forecasted and weighted prices provides insights about weekly and yearly trends.

Since the Bitcoin Historical Dataset used above contains records of the prices up until March 2021, the model would not have access to recent trends and patterns and would not be able to accurately forecast the predicted price. For this reason, we use the Bitcoin USD Yahoo Finance Dataset [7]. This dataset provides a comprehensive record of Bitcoin's price movements, trading volume, market capitalization, and other relevant metrics over a specific time period. This dataset allows for the analysis of Bitcoin's price trends, volatility, and trading activity over time. It provides insights into market behavior and investor sentiment over the recent periods. Data resampling is performed on the dataset to include observations only after 2014. We check for null values on this dataset and implement data slicing to create a subset of the dataframe with the date and open prices as the feature variables. Data visualization is performed to understand the variations and trends of the open prices since 2015. The prophet model is then trained after instantiating it with some parameters. The first parameter is the seasonality mode which determines how the model incorporates recurring patterns over fixed intervals. Our model uses the multiplicative seasonality mode which assumes that the seasonal effect is multiplied with the trend component, holiday component and the error component. By accounting these components, the model captures cyclic patterns which cannot be explained by specific events. Seasonalities are computed using Fourier series which decompose the periodic function into a sum of sine and cosine function with varying amplitudes and frequency. During model fitting, Prophet estimates amplitudes of these functions by using the Hamiltonian Markov Chain Monte Carlo algorithm. This algorithm is used to sample from the posterior distribution of various prophet components and is then able to capture the uncertainty inherent in the parameter estimation process and provide not just point forecasts but also probabilistic forecasts that account for the range of possible outcomes. The fourier series used in the seasonal component of the prophet model is based on the following formula:

$$S(t) = A(1) * \sin(\omega*t) + B(1) * \cos(\omega*t) + ... + A(N) * \sin(N*\omega*t) + B(N) * \cos(N*\omega*t)$$

The second parameter used is the number of potential change points. Changepoints are the timestamps where the growth rate of the time series is expected to change. Our model uses a changepoint value of 10 indicating that the model will consider 10 potential changepoints. Once the task of model fitting is done, we make future dataframes for the next 90 timestamps for forecasting. We also visualize the change points across the dataset and perform cross-validation on the model which returns a dataframe that contains the predicted values and the actual values for each of the cross-validation periods. A performance matrix is constructed to analyze the results. Once the model is trained and validated, the next task involves saving the .pickle file of the model for deployment. We use Pycharm since it provides built in support for Flask, code navigation and web server settings configuration.

7. FlowChart



(Fig.2. Block Diagram)

Screenshots of Dataset & Visualizations.

yahoo/finance

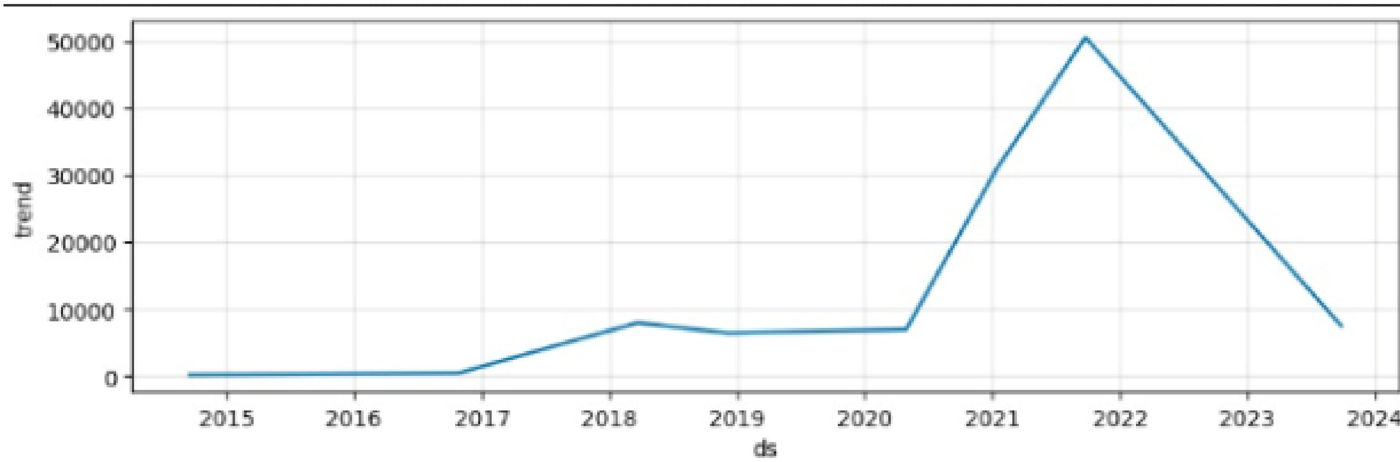
Search for news, symbols or companies

Currency in USD

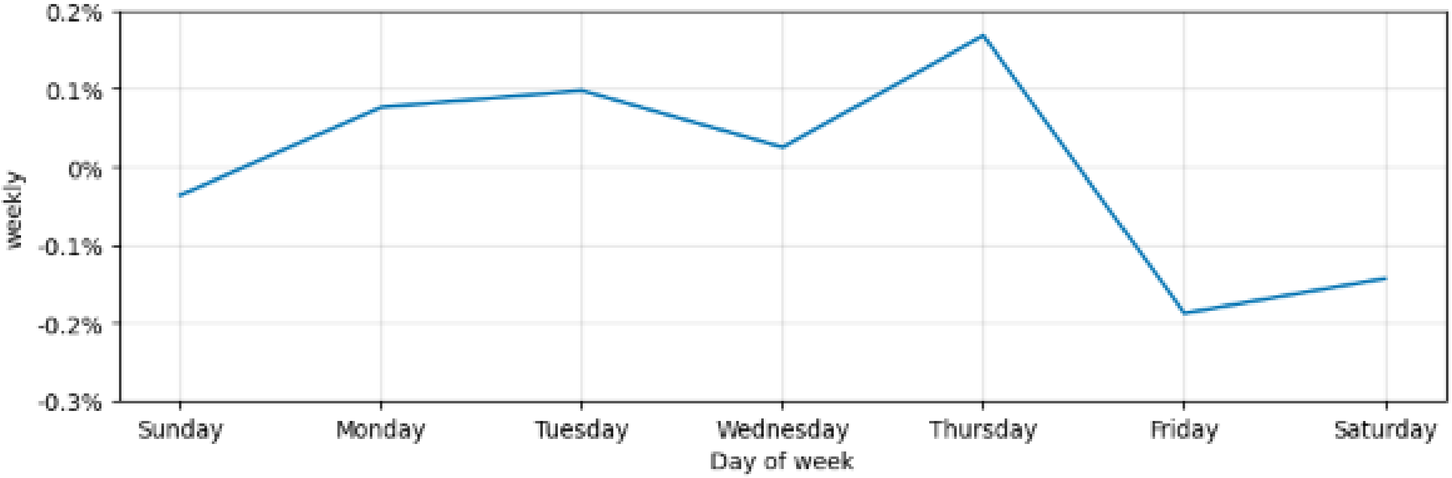
Download

Date	Open	High	Low	Close*	Adj Close**	Volume
Jun 30, 2023	30,446.09	31,238.34	30,363.27	30,658.47	30,658.47	17,310,715,904
Jun 29, 2023	30,086.19	30,796.25	30,057.20	30,445.35	30,445.35	13,180,860,821
Jun 28, 2023	30,696.56	30,703.28	29,921.82	30,086.25	30,086.25	14,571,500,779
Jun 27, 2023	30,274.32	31,006.79	30,236.65	30,688.16	30,688.16	16,428,827,944
Jun 26, 2023	30,480.52	30,636.03	29,955.74	30,271.13	30,271.13	16,493,186,997
Jun 25, 2023	30,545.15	31,041.27	30,327.94	30,480.26	30,480.26	12,703,464,114
Jun 24, 2023	30,708.74	30,804.15	30,290.15	30,548.70	30,548.70	12,147,822,496
Jun 23, 2023	29,896.38	31,389.54	29,845.21	30,695.47	30,695.47	24,115,570,085
Jun 22, 2023	29,995.94	30,496.00	29,679.16	29,912.28	29,912.28	20,653,160,491
Jun 21, 2023	28,311.31	30,737.33	28,283.41	30,027.30	30,027.30	33,346,760,979
Jun 20, 2023	26,841.66	28,388.97	26,668.79	28,327.49	28,327.49	22,211,859,147
Jun 19, 2023	26,335.44	26,984.61	26,312.83	26,851.03	26,851.03	12,826,986,222
Jun 18, 2023	26,510.46	26,675.93	26,325.89	26,336.21	26,336.21	9,565,695,129
Jun 17, 2023	26,328.68	26,769.39	26,174.49	26,510.68	26,510.68	11,090,276,850
Jun 16, 2023	25,575.28	26,463.17	25,245.36	26,327.46	26,327.46	16,324,646,965

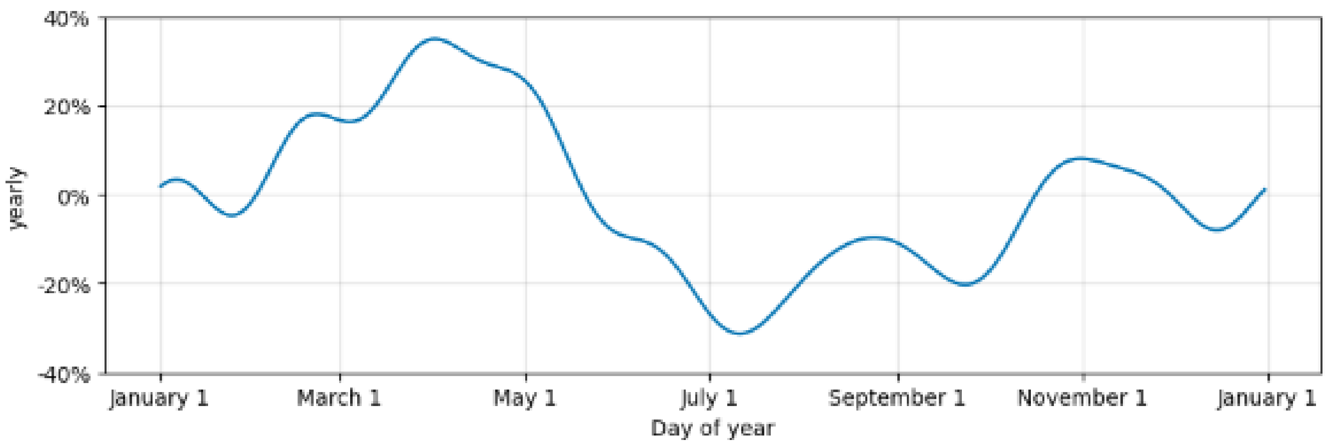
(Fig.3. Yahoo Finance Dataset Overview)



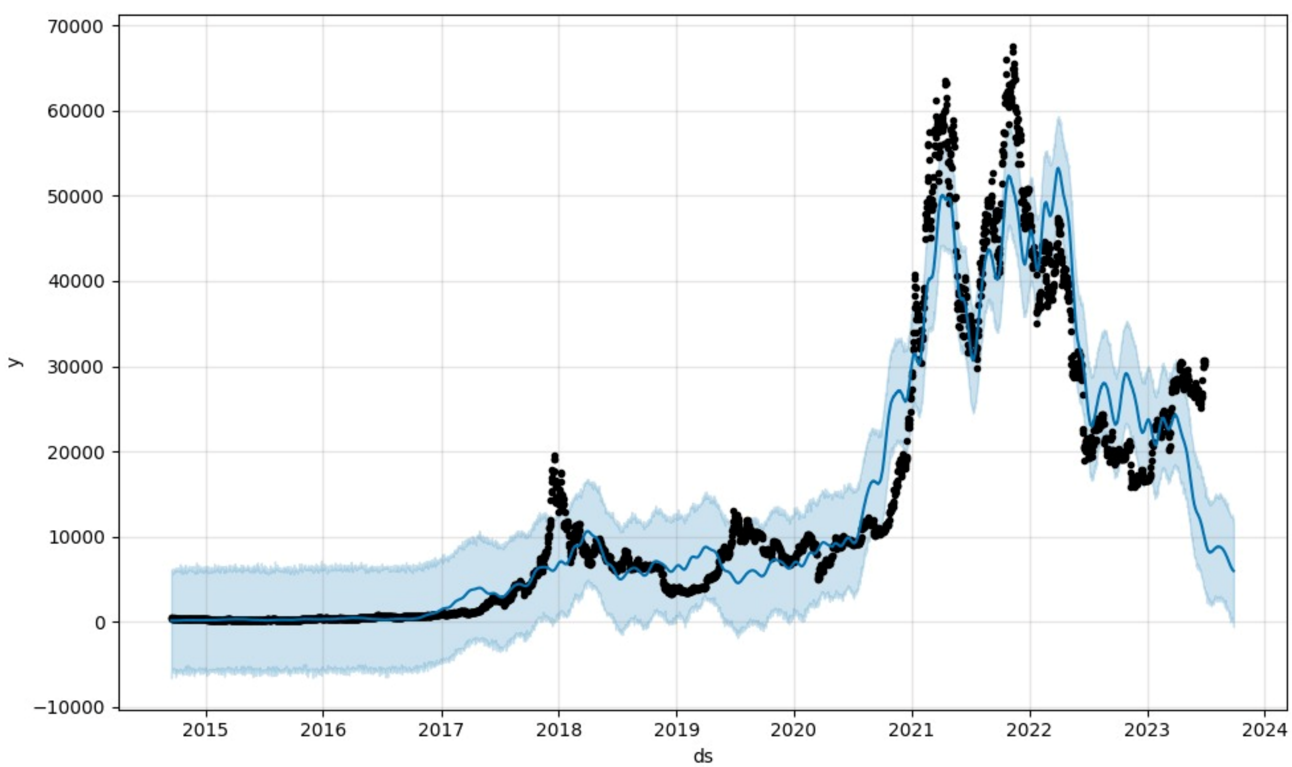
(Fig.4. Trend plot showing estimated overall trend over time)



(Fig.5. Weekly Seasonality plot)

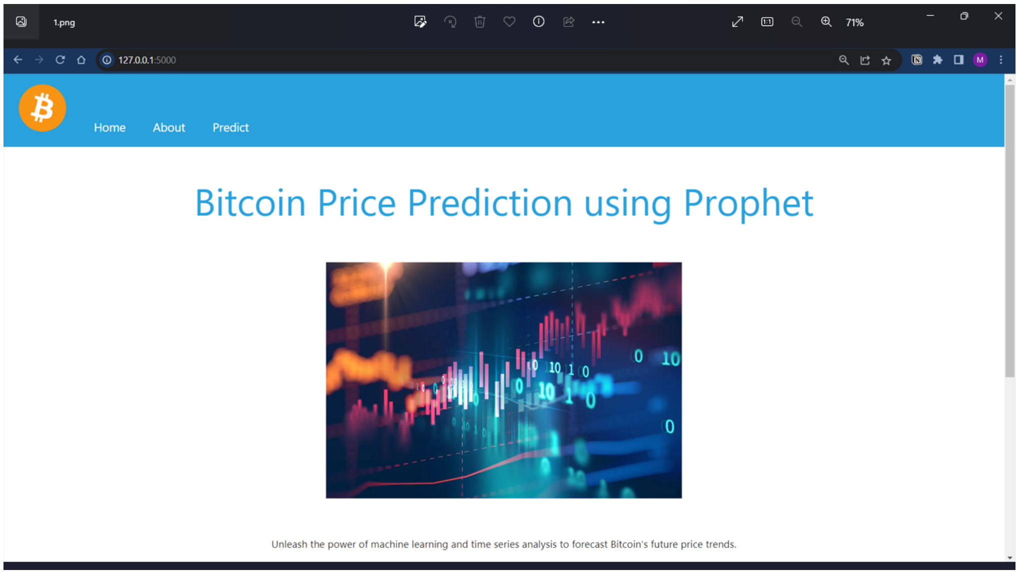


(Fig.6. Yearly Seasonality Plot)

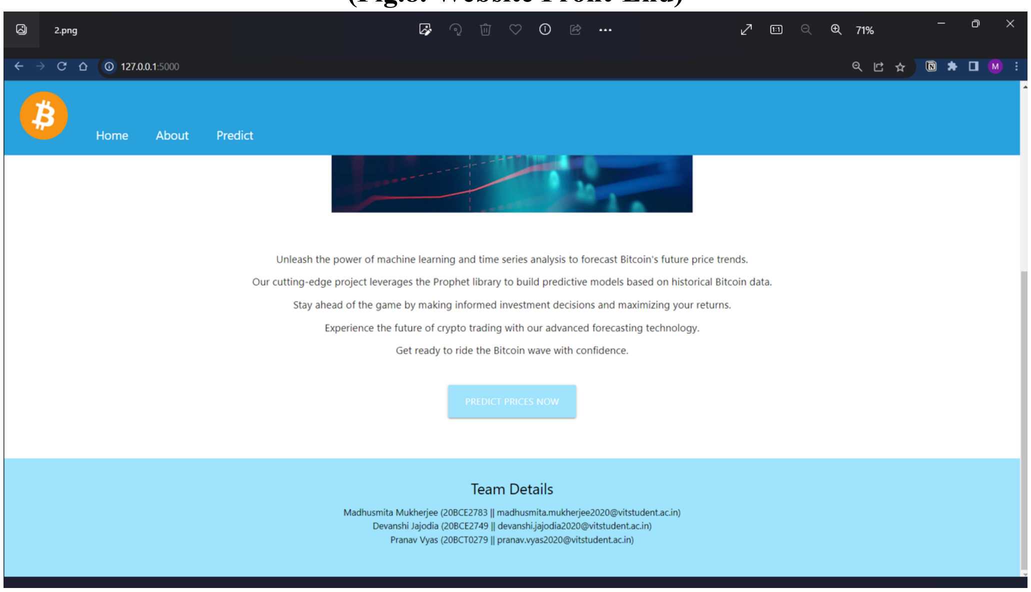


(Fig.7. Plot of weighted price, forecasted price and uncertainty intervals)

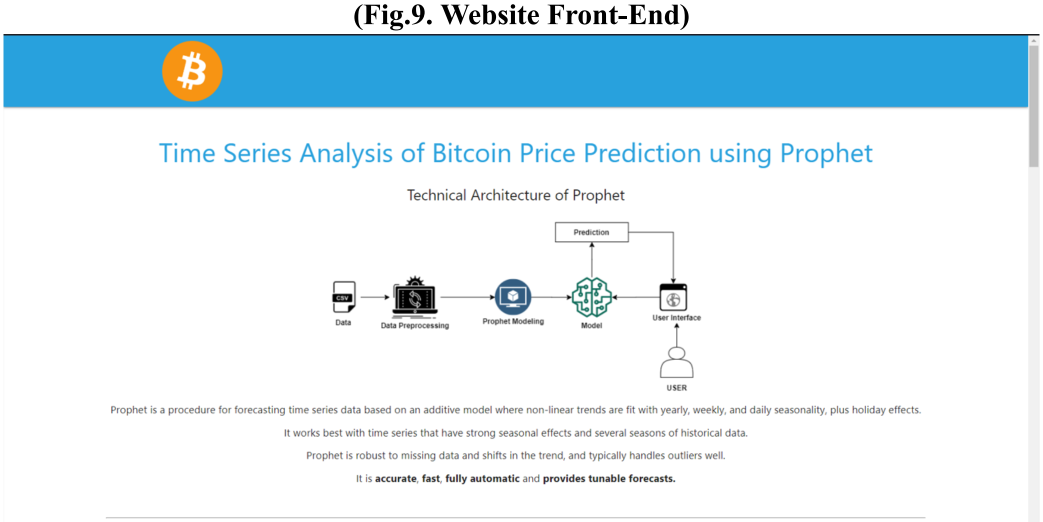
8. Results



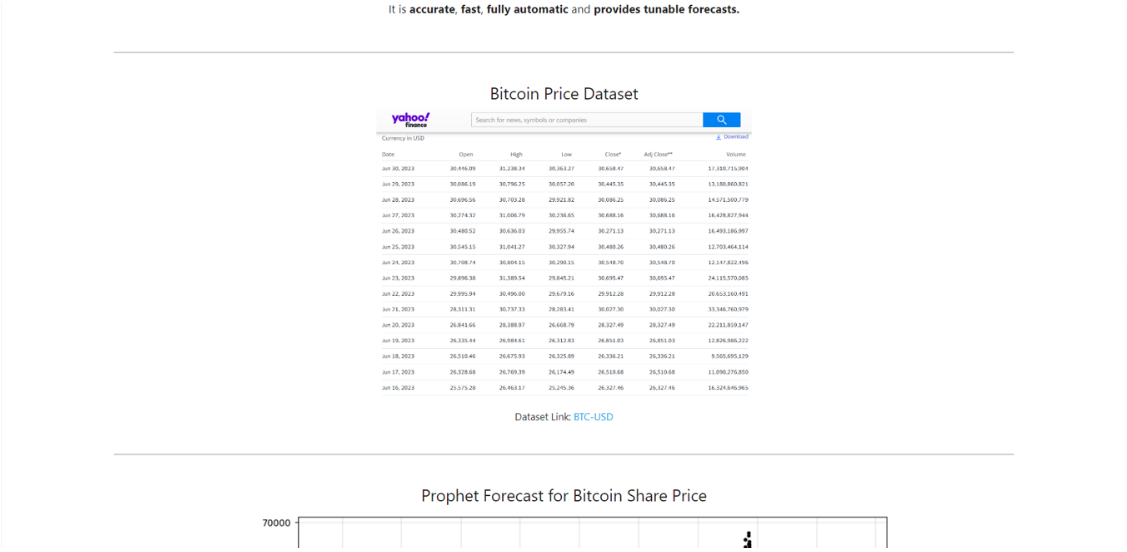
(Fig.8. Website Front-End)

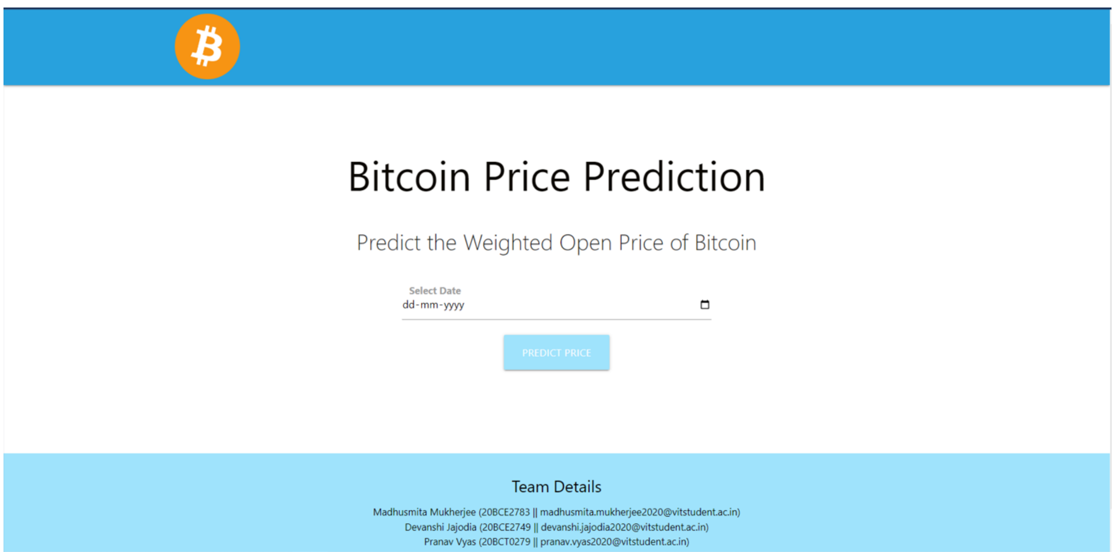


(Fig.9. Website Front-End)



(Fig.10.Website Front-End)





(Fig.11. Website Input Field)

(Fig.12. Website Prediction Screenshot)

9. Advantages and Disadvantages

Advantages:

- **Accurate Forecasting**: Time Series Prophet model is known for providing accurate forecasts for time series data. It can capture complex trends and non-linear patterns in the Bitcoin price data, enabling reliable predictions.
- **Flexibility**: The Prophet model is flexible and can handle various types of time series data, including Bitcoin prices. It can handle missing values, outliers, and irregular intervals in the data, making it suitable for real-world scenarios where data quality may not be perfect.
- **Intuitive Model**: The Prophet model is designed to be easy to use, even for users without extensive statistical or machine learning backgrounds. Its straightforward syntax and intuitive parameters make it accessible to beginners while still providing powerful forecasting capabilities.
- **Interpretability**: Prophet provides valuable insights into the underlying components of the time series, such as trend and seasonality which can help understand the driving factors behind Bitcoin price movements, aiding in decision-making and analysis.
- **Quick Prototyping**: The Prophet model is well-suited for rapid prototyping and iterative development. It can handle large datasets efficiently and provides fast model training, allowing for quick experimentation and iteration.
- **Time-Series Decomposition**: Prophet employs an additive time-series decomposition approach, breaking down the observed data into three components: trend, seasonality, and holiday effects. This decomposition enables the model to capture various patterns and underlying structures in the data, enhancing the accuracy of price predictions.
- **Uncertainty Estimation:** Prophet provides uncertainty estimation for its predictions, enabling users to quantify the confidence intervals around the forecasted prices. This feature is particularly valuable for decision-making processes, as it allows stakeholders to understand the level of risk associated with the predicted prices.

Disadvantages:

- **Limited Feature Engineering**: The Prophet model has a predefined set of features it can consider, such as trend, seasonality, and holidays. While these features are often sufficient for many time series forecasting tasks, they may not capture all the nuances and complexities of Bitcoin price movements, potentially limiting the model's predictive power.
- **Sensitivity to Parameter Tuning**: Although the Prophet model is designed to be user-friendly, it still requires careful parameter tuning to achieve optimal performance. Incorrectly set parameters may result in suboptimal forecasts or overfitting, requiring thorough experimentation and validation.
- **Inability to Capture Sudden Changes**: The Prophet model assumes that historical patterns will continue into the future and struggles to capture sudden and unexpected changes in the Bitcoin market, such as major news events, regulatory announcements, or market manipulation. It may take time for the model to adapt to new patterns, potentially impacting short-term predictions.
- **Lack of Long-Term Accuracy**: While Prophet can provide accurate short-term forecasts, its long-term predictions may be less reliable. Over extended forecast horizons, the model's accuracy may decrease due to the inherent volatility and unpredictability of the Bitcoin market.
- **Data Limitations**: The quality and availability of data can significantly impact the performance of the Prophet model. If the historical Bitcoin price data is limited or of poor quality, the model's ability to capture meaningful patterns and make accurate predictions may be compromised.
- **Limited Interpretability**: Although Prophet offers useful insights into trend, seasonality, and changepoints, its overall interpretability might be limited. The model provides decomposed components and visualizations but does not explicitly explain the relationships between input features, time series components, and the resulting predictions. This limitation might restrict the model's usefulness in scenarios where

10.Applications

- **Investment Decision-Making:** The ability to predict Bitcoin prices accurately can assist investors in making informed decisions regarding buying, selling, or holding cryptocurrency assets. By utilizing the Prophet model, investors can have a quantitative tool to assess potential price movements and adjust their investment strategies accordingly.
- **Risk Management:** Forecasting Bitcoin prices using the Prophet model can aid risk management strategies for individuals and businesses involved in the cryptocurrency market. By understanding future price trends, investors can implement risk mitigation measures, such as hedging strategies, diversification, or setting appropriate stop-loss levels.
- **Trading Strategies:** Traders can utilize Prophet-based Bitcoin price predictions to develop trading strategies that capitalize on short-term price fluctuations. By identifying patterns and trends, traders can make timely buy or sell decisions, improving their chances of profitable trades.
- **Market Analysis:** The Prophet model can provide valuable insights into the underlying factors driving Bitcoin price movements. Analysts can use the model's predictions to conduct market analysis, identify correlations with external events, and gain a deeper understanding of the cryptocurrency market's dynamics.
- **Portfolio Optimization:** Incorporating Bitcoin price forecasts into portfolio optimization models can help individuals and institutions manage their cryptocurrency portfolios more effectively. By considering the predicted returns and risks associated with Bitcoin, investors can optimize their asset allocation and balance risk-reward trade-offs.
- **Financial Planning:** Bitcoin price predictions can be integrated into financial planning models to assess the potential impact of cryptocurrency investments on long-term financial goals. Individuals can use these predictions to estimate future returns, plan for retirement, or evaluate the feasibility of specific financial objectives.
- **Regulatory Compliance:** Financial institutions and regulatory bodies can leverage Bitcoin price predictions to monitor and enforce compliance measures related to cryptocurrency transactions. By anticipating potential price fluctuations, regulators can identify suspicious or anomalous activities in the market.
- **Cryptocurrency Exchange Operations**: Exchanges can utilize the Prophet model to optimize their operations, such as order book management, liquidity provision, and risk assessment. Accurate price forecasts can assist in managing trade execution, reducing slippage, and improving overall market efficiency.

11.Conclusion

In conclusion, this project successfully implemented the Time Series Prophet model using the yfinance dataset to predict Bitcoin prices.The project began by collecting historical Bitcoin price data using the yfinance library, which provides access to a wide range of financial data, including cryptocurrency prices. The dataset was preprocessed to ensure data quality, addressing issues such as missing values, outliers, and data inconsistencies. The Time Series Prophet model was then trained on the preprocessed dataset, incorporating factors such as trend, seasonality, and holidays to capture the underlying patterns in the Bitcoin price data. The flexibility of the Prophet model allowed for the efficient handling of the yfinance dataset, enabling the extraction of meaningful insights for accurate predictions. Evaluation of the Prophet model's performance was conducted by comparing its predictions against the actual Bitcoin prices from the yfinance dataset. Metrics such as mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE) were utilized to assess the model's predictive capabilities and quantify the level of forecast accuracy achieved. The project findings shed light on the advantages and disadvantages of utilizing the Time Series Prophet model with the yfinance dataset for Bitcoin price prediction. The advantages included the model's ability to provide accurate forecasts, handle various time series data characteristics, offer an intuitive model structure, provide interpretability, and facilitate rapid prototyping. However, the limitations such as limited feature engineering, sensitivity to parameter tuning, challenges in capturing sudden changes, potential lack of long-term accuracy, and data limitations were also acknowledged.

12. Future Scope

This project opens up avenues for future research and development in the field of Bitcoin price prediction using the Time Series Prophet model. Several areas offer potential for improvement and expansion. Advanced feature engineering can explore additional factors impacting Bitcoin prices, such as sentiment analysis and macroeconomic indicators. Hybrid models, combining the Prophet model with other techniques like LSTM or ARIMA, can enhance accuracy and robustness. Adaptive models that quickly adapt to market dynamics can capture sudden changes and improve short-term predictions. Fine-grained analysis can focus on specific time periods or events for more precise forecasting. Ensemble approaches that combine multiple models can mitigate limitations and generate more reliable predictions. Real-time prediction capabilities can be developed using streaming data processing techniques. Evaluation and benchmarking against other models can provide insights into the Prophet model's performance. Future research in these areas will enhance the usability and effectiveness of the Time Series Prophet model for Bitcoin price prediction, benefiting decision-making and trading strategies in the dynamic cryptocurrency market.

13. Bibliography

[1] Navrátil, Miroslav & Kolková, Andrea. (2019). Decomposition and Forecasting Time Series in Business Economy Using Prophet Forecasting Model. Central European Business Review. 8. 26-39. 10.18267/j.cebr.221.

[2] C. R. Madhuri, M. Chinta and V. V. N. V. P. Kumar, "Stock Market Prediction for Time-series Forecasting using Prophet upon ARIMA," 2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 2020, pp. 1-5, doi: 10.1109/ICSSS49621.2020.9202042.

[3] P. Yuvarani, P. Bharani, B. Dharun and P. Dinesh, "Time Series Forecasting of Ethereum Price by FB-Prophet," 2023 4th International Conference on Signal Processing and Communication (ICSPC), Coimbatore, India, 2023, pp. 272-277, doi: 10.1109/ICSPC57692.2023.10125661.

[4] Jamaani, Fouad & Li, Xiaobin & Sengupta, Tuhin & Mohammed, Kamel. (2023). Forecasting the lithium mineral resources prices in China: Evidence with Facebook Prophet (Fb-P) and Artificial Neural Networks (ANN) methods. Resources Policy. 82. 10.1016/j.resourpol.2023.103580.

[5] A. Maheshwari, A. Malhotra, S. Tuteja, M. Ranka and M. S. A. Basha, "Prediction of Stock Prices using Prophet Model with Hyperparameters tuning," 2022 IEEE North Karnataka Subsection Flagship International Conference (NKCon), Vijaypur, India, 2022, pp. 1-5, doi: 10.1109/NKCon56289.2022.10126796.

[6] https://www.kaggle.com/datasets/mczielinski/bitcoin-historical-data

[7] https://finance.yahoo.com/quote/BTC-USD/history?p=BTC-USD

Appendix

The source code can be found at the github link below:
https://github.com/madhusmita1307/Time_Series_Analysis-Bitcoin_Price_Prediction_Prophet/tree/main