📈 Stock Price Forecasting with Prophet –Venu Yerramsetti

This project forecasts Apple (AAPL) stock prices using **Prophet**, a powerful open-source time series forecasting library. Historical data is fetched from Yahoo Finance via `yfinance`, and forecasts are visualized using `matplotlib`.

🔧 Technologies Used

- Python
- yfinance
- Prophet
- Matplotlib
- Pandas

🗂 Project Structure

File Name	Description
`stock_forecast.py`	Python script for downloading data, training model, forecasting
`forecast_plot.png`	Image file showing the forecasted stock price
`requirements.txt`	List of Python dependencies
`README.md`	Project overview and instructions

📸 Output

The forecast plot will be saved as `forecast_plot.png`.
It will also be displayed interactively using `matplotlib`.

💡 About Prophet

Prophet is a forecasting tool built by Meta (Facebook) for time series data. It supports modeling:

- Seasonality (yearly, weekly, daily)
- Holiday effects
- Non-linear trends
- Missing data and outliers

🌐 How GitHub Fits In

This GitHub repository serves as a central hub for this project:

- **Hosts the source code:** All the project files are stored here.
- **Tracks project history:** You can see all the changes made to the code over time.
- **Shares the project with others:** Makes it easy for collaborators and others to access and contribute.

Important Note: GitHub does not execute your code directly. To run this project, you'll need to **clone the repository** and execute the code locally on your machine or within a cloud notebook environment.

🚀 Optional: Run in Google Colab or Binder

For a quick and easy way to run this notebook online without any local setup, you can use one of these free platforms:

Alternatively, you can upload the notebook directly to Google Colab and run all cells there.

▶ How to Run the Project Locally

```
# 1. Clone the Repository
git clone https://github.com/VenuYerramsetti/stock-price-forecasting-prophet
cd stock-price-forecasting-prophet

# 2. Set Up the Environment
pip install -r requirements.txt
```

```
# If needed, install manually:
pip install yfinance prophet matplotlib pandas

# If Prophet fails to install:
pip install pystan==2.19.1.1
pip install prophet

# 3. Run the Code
# Run the script:
python stock_forecast.py

# Or run the Jupyter Notebook:
jupyter notebook
# Then open `stock_forecast.ipynb` and run all cells
```

🖱️ Requirements

- yfinance==0.2.61
- prophet==1.1.6
- matplotlib==3.4.3
- pandas==1.3.3

💻 Code blocks Explanation

```
import yfinance as yf
from prophet import Prophet
import pandas as pd
import matplotlib.pyplot as plt
```

Imports the necessary libraries:

- yfinance to fetch stock data from Yahoo Finance.
- Prophet for time series forecasting.
- pandas for data manipulation.
- matplotlib.pyplot for plotting.

1. Download stock data

```
data = yf.download('AAPL', start='2018-01-01', end='2024-12-31')
```

- Downloads historical stock data for Apple (ticker symbol AAPL) from January 1, 2018 to December 31, 2024.
- The result, data, is a pandas DataFrame containing Open, High, Low, Close, Volume, and Adj Close prices for each trading day in that period.

2. Prepare data for Prophet

```
df = pd.DataFrame() df['ds'] = data.index df['y'] = data['Close'].values
```

- Creates a new DataFrame df formatted specifically for Prophet.
- ds stand for datestamp — it must be a column of datetime objects representing the dates of observations. Here it is set from the index of the original data DataFrame (which contains the trading dates).
- y is the target variable to forecast — here, the closing price of the stock on each date. Prophet expects the target variable column to be named y.
- This formatting is required because Prophet only accepts DataFrames with these two columns named exactly ds and y.

Check types (optional debug print)

```
print(type(df['y']), df['y'].shape)
```

- Prints the type and shape of the y column. This is a quick sanity check to ensure the target data is in the expected format (a pandas Series with the correct number of elements).

3. Initialize and fit model

```
model = Prophet() model.fit(df)
```

- Initializes a new Prophet forecasting model with default parameters.
- Fits the model to the historical stock data you prepared in df. Prophet uses this data to learn patterns, trends, and seasonality.

4. Make future dataframe & forecast

```
future = model.make_future_dataframe(periods=180)
forecast = model.predict(future)
```

- make_future_dataframe(periods=180) generates a DataFrame that extends the original dates (ds) by 180 future days. This DataFrame will be used to generate predictions.
- model.predict(future) computes the forecast for both historical and future dates, producing predicted values (yhat) along with confidence intervals and components (like trend and seasonality).

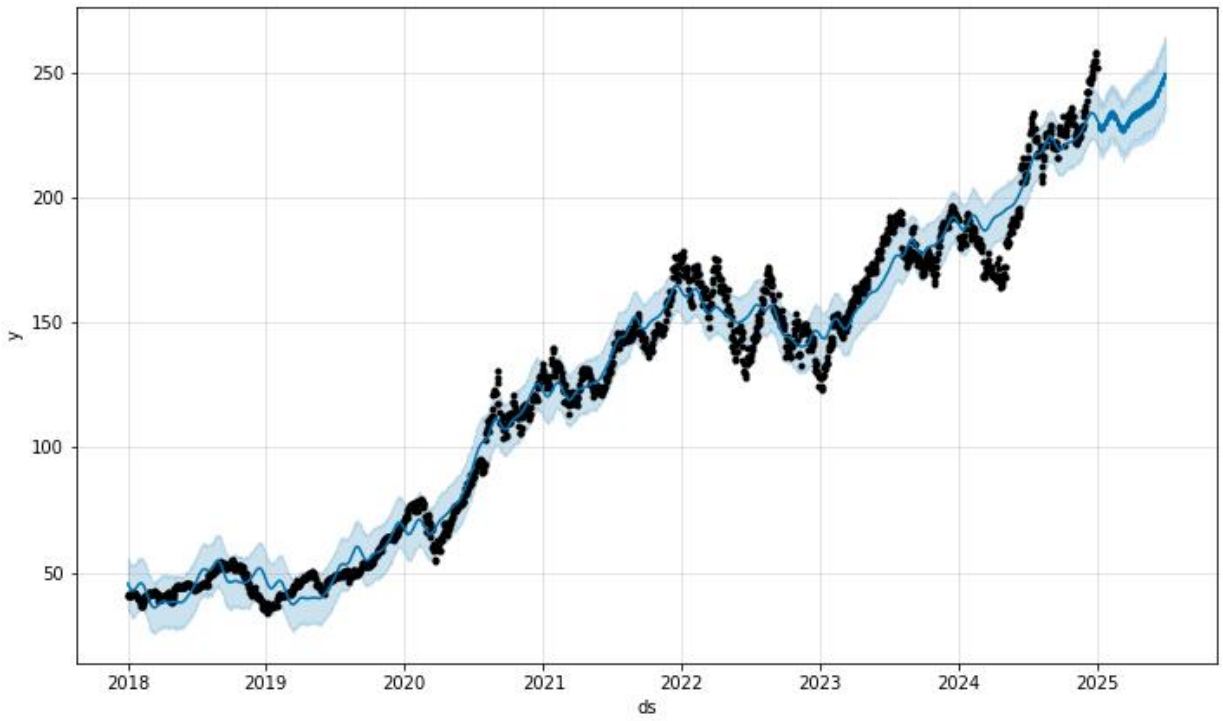
```
fig1 = model.plot(forecast)
fig1.savefig("forecast_plot.png")
plt.title("AAPL Stock Price Forecast")
plt.show()
```

- Creates a plot of the forecasted data including historical and predicted values using Prophet's built-in plotting method.
- Saves the plot as forecast_plot.png in your working directory.
- Adds a title to the plot with matplotlib.
- Displays the plot interactively on your screen.

6. Description of the Forecast Plot (forecast_plot.png)

This plot, generated by the Prophet library, visually represents the historical Apple (AAPL) stock prices alongside the model's 180-day future forecast. It provides a clear and concise summary of the time series analysis and the projected trajectory of the stock.

- **Key visual elements and their interpretation:**



(forecast_plot.png)

- **Black Dots (Historical Data Points):**
  - These dots represent the **actual daily closing prices of Apple (AAPL) stock** from early 2018 up to the end of 2024. They clearly show the stock's historical performance, including its significant growth, particularly from late 2019/early 2020 onwards, and the characteristic market fluctuations.
- **Dark Blue Line (Prophet's Trend/Forecast - `yhat`):**
  - This continuous dark blue line illustrates the **Prophet model's underlying trend and its point forecast**.
  - **In the historical period:** It demonstrates how well the model captures the long-term movement and smoother patterns within the past stock data, effectively averaging out daily volatility.
  - **In the future period (extending into 2025):** This line represents the model's **predicted daily closing prices** for the next 180 days. We can observe a continuation of the upward trend from late 2024 into the forecast horizon.
- **Light Blue Shaded Area (Uncertainty Interval - `yhat_lower` and `yhat_upper`):**
  - This crucial shaded region signifies the **model's confidence or credible interval** for its predictions (by default, 80%).
  - It shows the **range within which the actual stock price is expected to fall**.

- **Observation:** The band is relatively tighter for the historical data where the model has observed values. As the forecast extends into the future (into 2025), the **light blue band visibly widens**. This widening is a natural and important characteristic of time series forecasting, indicating that the **model's uncertainty increases further out in time**, which is particularly relevant for volatile assets like stocks.

7. Overall Insights from the plot

The visualization effectively highlights:

Strong Upward Trend: The dominant feature is the significant and sustained upward trend of AAPL stock over the observed period, which the Prophet model has successfully learned and projected into the near future.
Model Fit: The dark blue line generally follows the black dots well in the historical period, indicating a good fit of the Prophet model to the historical price movements.
Future Projection: The forecast for the first half of 2025 shows a continued upward trajectory, but the widening uncertainty band correctly advises caution regarding the precision of these longer-term predictions.

Summary of ds and y in Prophet: ds (datestamp): A column of dates (datetime objects) representing the time component of your data. Prophet requires this exact column name and type.

y: The numeric value you want to forecast (in your case, the daily closing price of AAPL). Prophet expects this to be a numeric column named