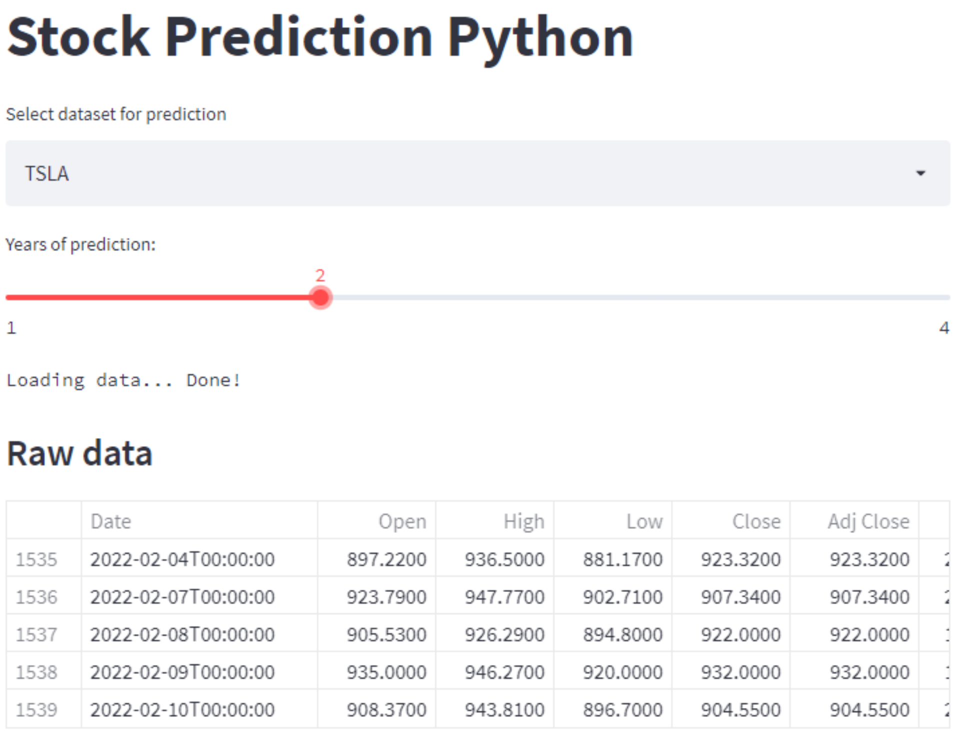
Preview of the website



Stock Prediction Python

Select dataset for prediction

TSLA ▾

Years of prediction:

2

1 4

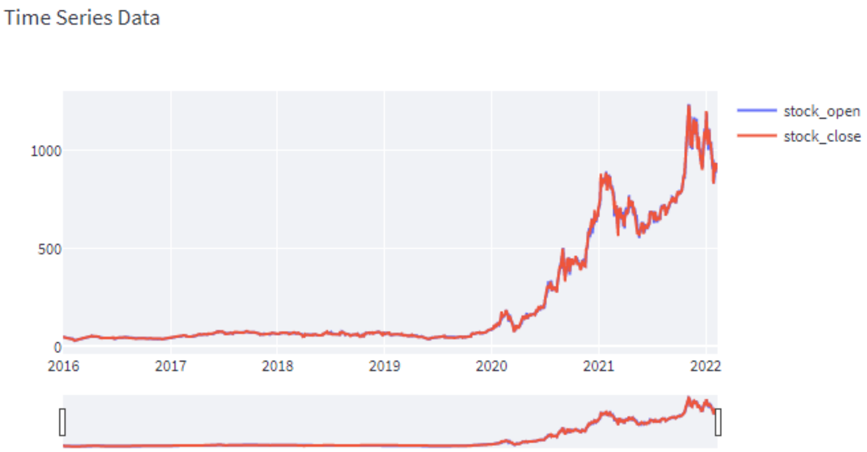
```
Loading data... Done!
```

Raw data

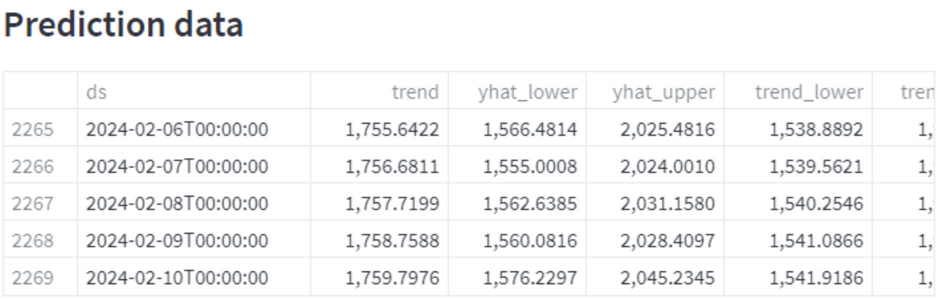
	Date	Open	High	Low	Close	Adj Close	
1535	2022-02-04T00:00:00	897.2200	936.5000	881.1700	923.3200	923.3200	2
1536	2022-02-07T00:00:00	923.7900	947.7700	902.7100	907.3400	907.3400	2
1537	2022-02-08T00:00:00	905.5300	926.2900	894.8000	922.0000	922.0000	1
1538	2022-02-09T00:00:00	935.0000	946.2700	920.0000	932.0000	932.0000	1
1539	2022-02-10T00:00:00	908.3700	943.8100	896.7000	904.5500	904.5500	2

Features:

- Dropdown bar to select Stock ticker symbol
- Slide bar to select prediction data from 1 to 4 years later
- Raw Data table taken from yahoo finance API



Time Series Data



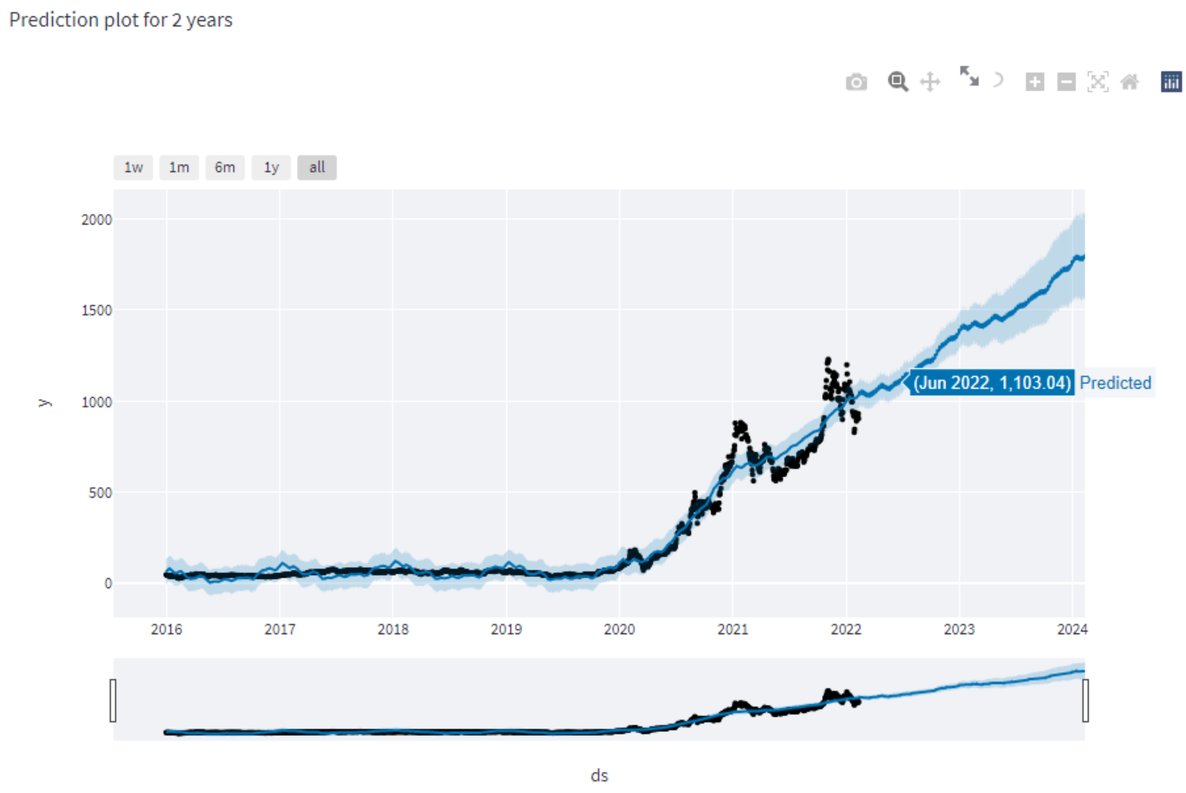
Prediction data

	ds	trend	yhat_lower	yhat_upper	trend_lower	tren
2265	2024-02-06T00:00:00	1,755.6422	1,566.4814	2,025.4816	1,538.8892	1,
2266	2024-02-07T00:00:00	1,756.6811	1,555.0008	2,024.0010	1,539.5621	1,
2267	2024-02-08T00:00:00	1,757.7199	1,562.6385	2,031.1580	1,540.2546	1,
2268	2024-02-09T00:00:00	1,758.7588	1,560.0816	2,028.4097	1,541.0866	1,
2269	2024-02-10T00:00:00	1,759.7976	1,576.2297	2,045.2345	1,541.9186	1,

Features:

- Plotted graph from raw data retrieved from yahoo finance
- Prediction data table achieved using Machine learning from Prophet library

Prediction plot for 2 years

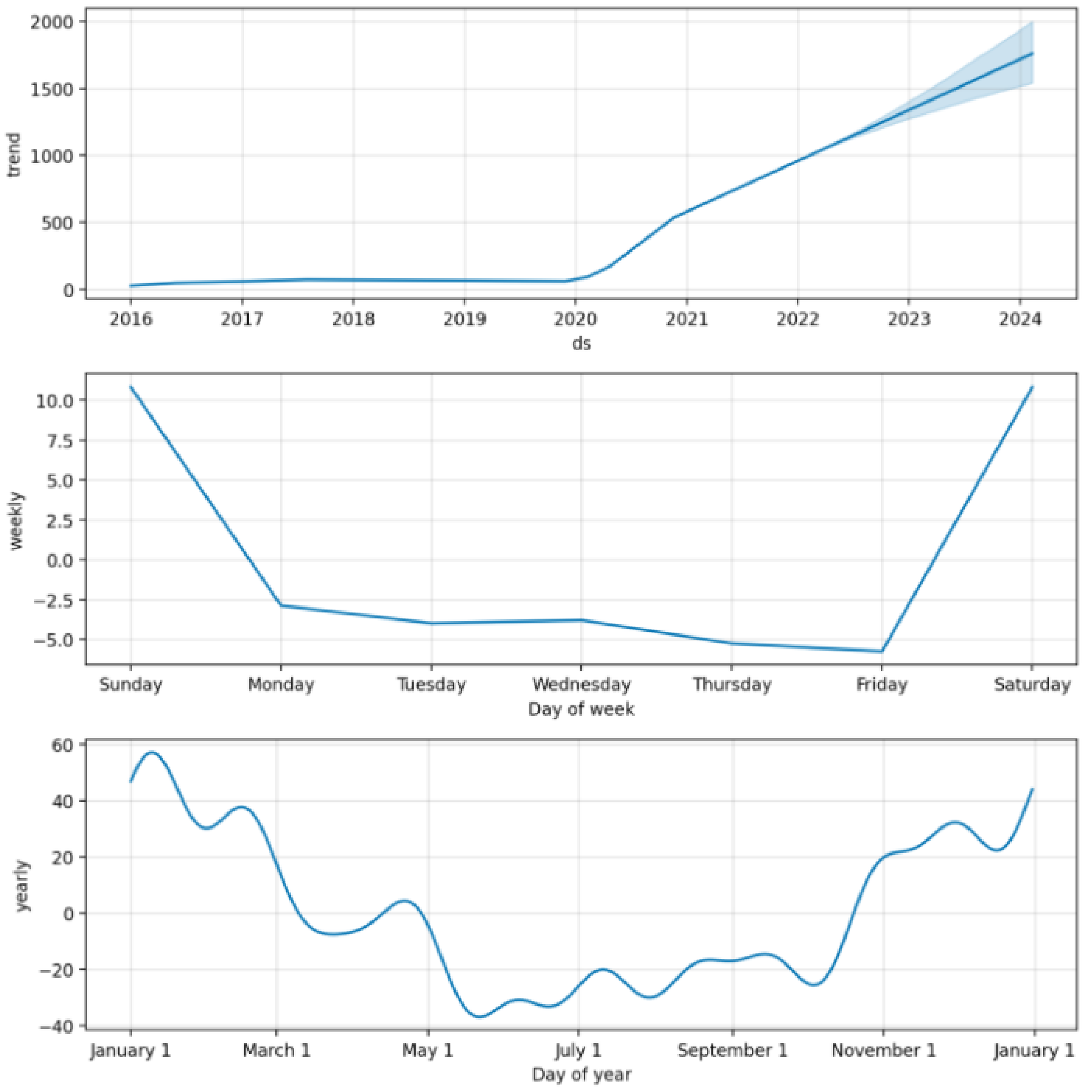


Mouse hovering on the month of June 2022 predicted price action on TSLA stock is $1,103.04.

Features:

- Selected 2 Years period of prediction
- Mouse able to hover on the graph trend to see predicted values

Prediction components



Features:

- Spilt prediction components of trend, weekly and yearly prediction data on graph within these specific parameters

SOURCE CODE

```python
##Stock Prediction
#commands to run file
#pip install yfinance prophet plotly
#streamlit run stockprediction.py

import streamlit as st
from datetime import date

import yfinance as yf
from prophet import Prophet
from prophet.plot import plot_plotly
from plotly import graph_objs as go

START = "2016-01-01"
TODAY = date.today().strftime("%Y-%m-%d")

st.title("Stock Prediction Python")

stocks = ("AAPL", "GOOGL", "MSFT", "TSLA", "NFLX", "AMZN", "FB") #7 stocks for now
selected_stock = st.selectbox("Select dataset for prediction", stocks) #dropdown slectbox to choose stocks

num_years = st.slider("Years of prediction:", 1, 4)
period = num_years * 365  #period of days


@st.cache #cache the yfinance download data so not req to redownload selected stocfks
def load_data(ticker):
    data = yf.download(ticker,START, TODAY)
    data.reset_index(inplace=True)
    return data

data_load_state = st.text("Load data...")
data = load_data(selected_stock)
data_load_state.text("Loading data... Done!")

st.subheader('Raw data')
st.write(data.tail())

#Plotting raw data table
def plot_raw_data():
    fig = go.Figure()
    fig.add_trace(go.Scatter(x=data['Date'], y=data['Open'], name='stock_open'))
    fig.add_trace(go.Scatter(x=data['Date'], y=data['Close'],
name='stock_close'))
    fig.layout.update(title_text="Time Series Data",
xaxis_rangeslider_visible=True)
    st.plotly_chart(fig)

plot_raw_data()

#Predicting with Prophet
df_train = data[['Date','Close']]
df_train = df_train.rename(columns={"Date":"ds", "Close":"y"})

m = Prophet()
m.fit(df_train)
future = m.make_future_dataframe(periods=period)
forecast = m.predict(future)

#Show and plot prediction
st.subheader('Prediction data')
st.write(forecast.tail())

st.write(f'Prediction plot for {num_years} years')
fig1 = plot_plotly(m, forecast)
st.plotly_chart(fig1)

st.write("Prediction components")
fig2 = m.plot_components(forecast)
st.write(fig2)
```