Journal:

02/12/2024: Team discussion regarding the topic selection and project work distribution among each other.

All 4 of us sat together and started discussing about the project deliverables and what is the expected results. Since the objective is revolving around Datasets and Machine Learning Algorithms, we decided to think of the topic on which we are going to work.

05/12/2024: Next team meeting on topic finalization

From the last meeting we were clear about the objective, and everyone came up with some topics which we can consider. My teammates suggested topics like Sentiment analysis and weather forecasting and prediction using AI/ML techniques. At the beginning, we all agreed to work on Sentiment Analysis from Twitter data.

10/12/2024: Topic change because we wanted to work on something different.

Sentiment analysis using twitter data was already done by illiyaz in Data Analytics project. He had a grip on some algorithms like LogisticRegression and Classifiers, but he wanted to work on some different algorithms which are related to time series forecasting. So, we decided to work on Stock price analysis and prediction using Regression models.

15/12/2024: Dataset collection and further analysis.

After going through many websites and datasets, some good datasets were selected by the team. Kshitij gave an idea for using one of the inbuilt libraries like y-finance which gives data for different stock prices. Which seemed to be a good idea at first. And we started to work on it, how to include y-finance and working on the coding part.

20/12/2024: Coding process

As kshitij was working on including the y-finance module to fetch the stock data, illiyaz was thinking how we can apply the Machine Learning Algorithms to the datasets which gives best results. Initially we decide to go with Linear Regression it was looking a better option, as it is simple to understand and implement and it is easier to interpret the output coefficients.

But later illiyaz thought This ML model works best when the dataset has Linear relations. But as we know stock price predictions and values are never linear. The relationships vary everyday. Hence He decided to go with other regression models.

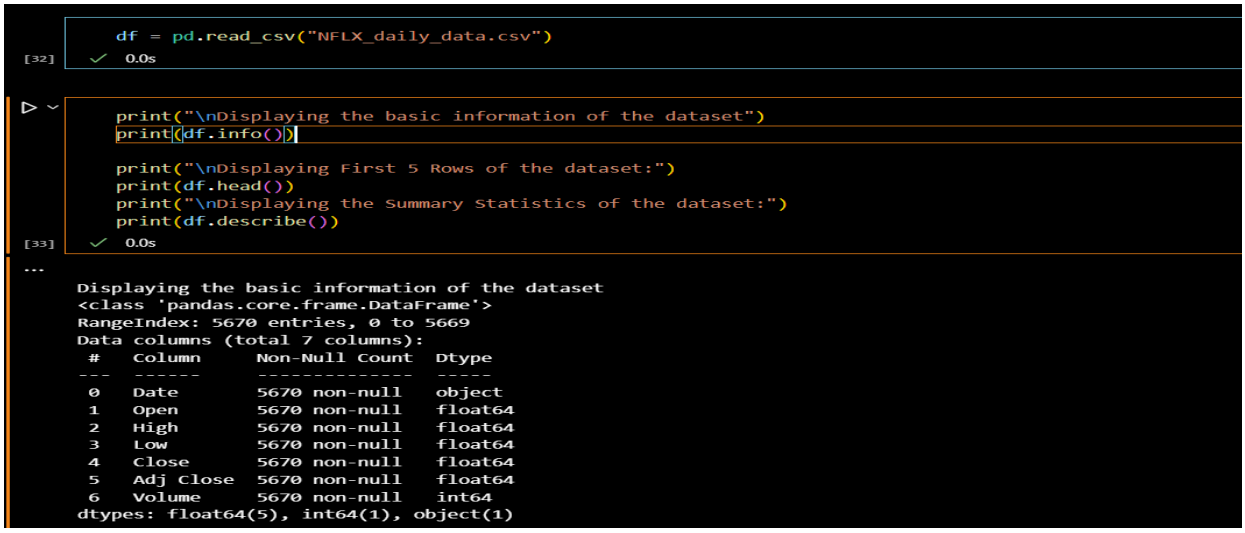
22/12/2024: Taking proper datasets(csv files) instead of y-finance module

Hitesh suggested that the y-finance module has some disadvantages but he still wants to work on that dataset but the objective of the project missed. As we need to collect data and preprocess that but also we need to keep checking and comparing the trends, so it would be better to get a proper dataset that has Stock values like Opening, Closing and Highest stock values for the day.

Shravani Ravindra Waghmare (23305665)

For dataset I searched datasets from different websites like (UCI Machine Learning Repository, Google Dataset Search, Data.gov and Kaggle) but I got the datasets from Kaggle that suits best for our project. Here is the link for the datasets (https://www.kaggle.com/datasets/zongaobian/netflix-stock-data-and-key-affiliated-companies). In this dataset We have stock prices of many different companies that are interrelated to each other. So we decided to work on the Netflix Stock Prediction along with other stocks(Amazon, Nvidia, SONY) which are highly interrelated with Netflix stocks. The Netflix is the main model of out project so if any changes happened in this model then all other model changed their stocking prediction.

23/12/2024: All of us took one dataset and started working on the datasets equally also we working on the code part. Illiyaz took Netflix Stock data imported required dependencies and loaded the dataset into VS-Code. Started displaying the dimensionality of it and important features.



```
[32]  df = pd.read_csv("NFLX_daily_data.csv")
      ✓ 0.0s

      print("\nDisplaying the basic information of the dataset")
      print(df.info())

      print("\nDisplaying First 5 Rows of the dataset:")
      print(df.head())
      print("\nDisplaying the Summary Statistics of the dataset:")
      print(df.describe())
[33]  ✓ 0.0s

...
Displaying the basic information of the dataset
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5670 entries, 0 to 5669
Data columns (total 7 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Date       5670 non-null   object
 1   Open       5670 non-null   float64
 2   High       5670 non-null   float64
 3   Low        5670 non-null   float64
 4   Close      5670 non-null   float64
 5   Adj Close  5670 non-null   float64
 6   Volume     5670 non-null   int64
dtypes: float64(5), int64(1), object(1)
```

As it is important to understand the important features and data types of the datasets, illiyaz decided to visualize his dataset. He checked the Machine learning model was working properly or giving accurate stocking accuracy.

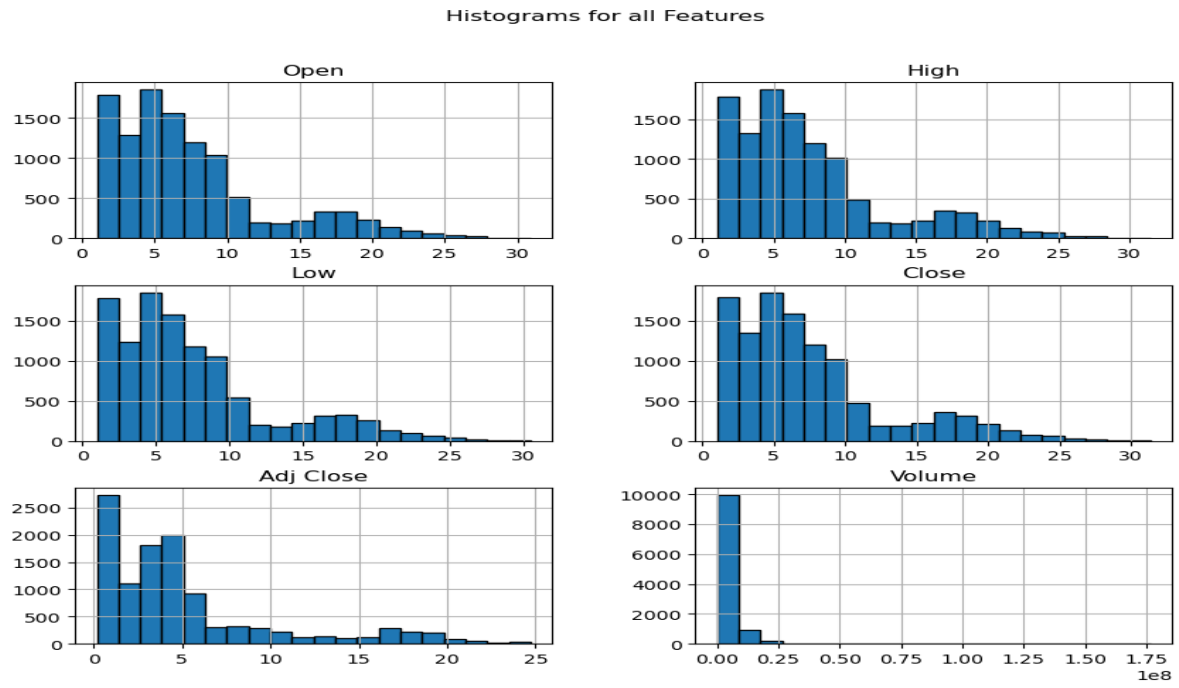
```
    print("\nChecking Missing Values in the dataset:")
    print(df.isnull().sum())
[34] ✓ 0.0s
...
Checking Missing Values in the dataset:
Date          0
Open          0
High          0
Low           0
Close         0
Adj Close     0
Volume        0
dtype: int64
```

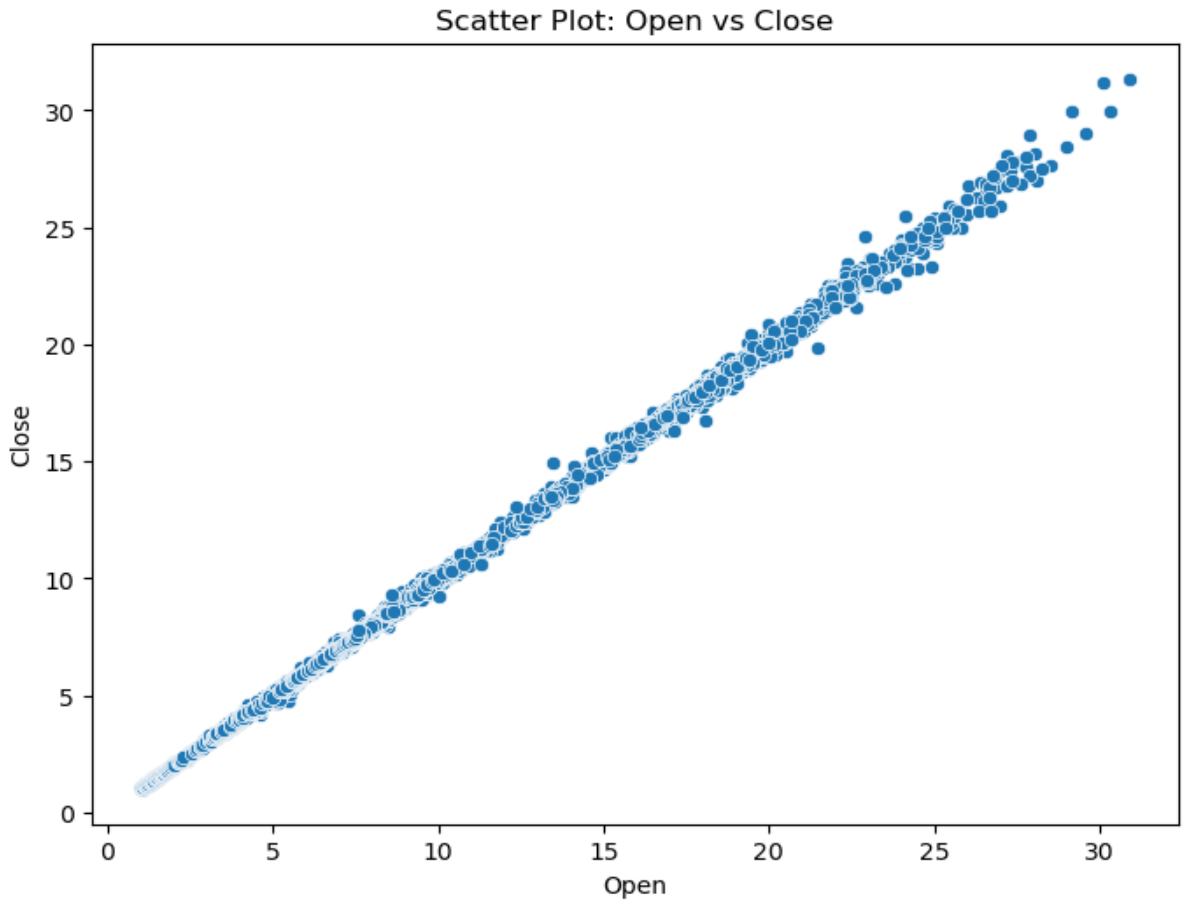
He found out that this is a clean dataset, which has no missing values. Hence there was no need to remove any rows or columns.

24/12/2024: Data Preprocessing and EDA(Exploratory Data Analysis)

He was sure about handling missing values, he thought removing outliers would be great so that we have less data to work on. But if we think about it, in stock predictions, we cannot remove any entries just because it is an outlier. We need each day data to check and predict the future stock prices, so we need data for all the days. So he started plotting graphs for data visualization.

First he plotted histogram for all the features and then Scatter plot for Opening and Closing values of the stock. This gives us the idea of the top and bottom constraints for the stock.



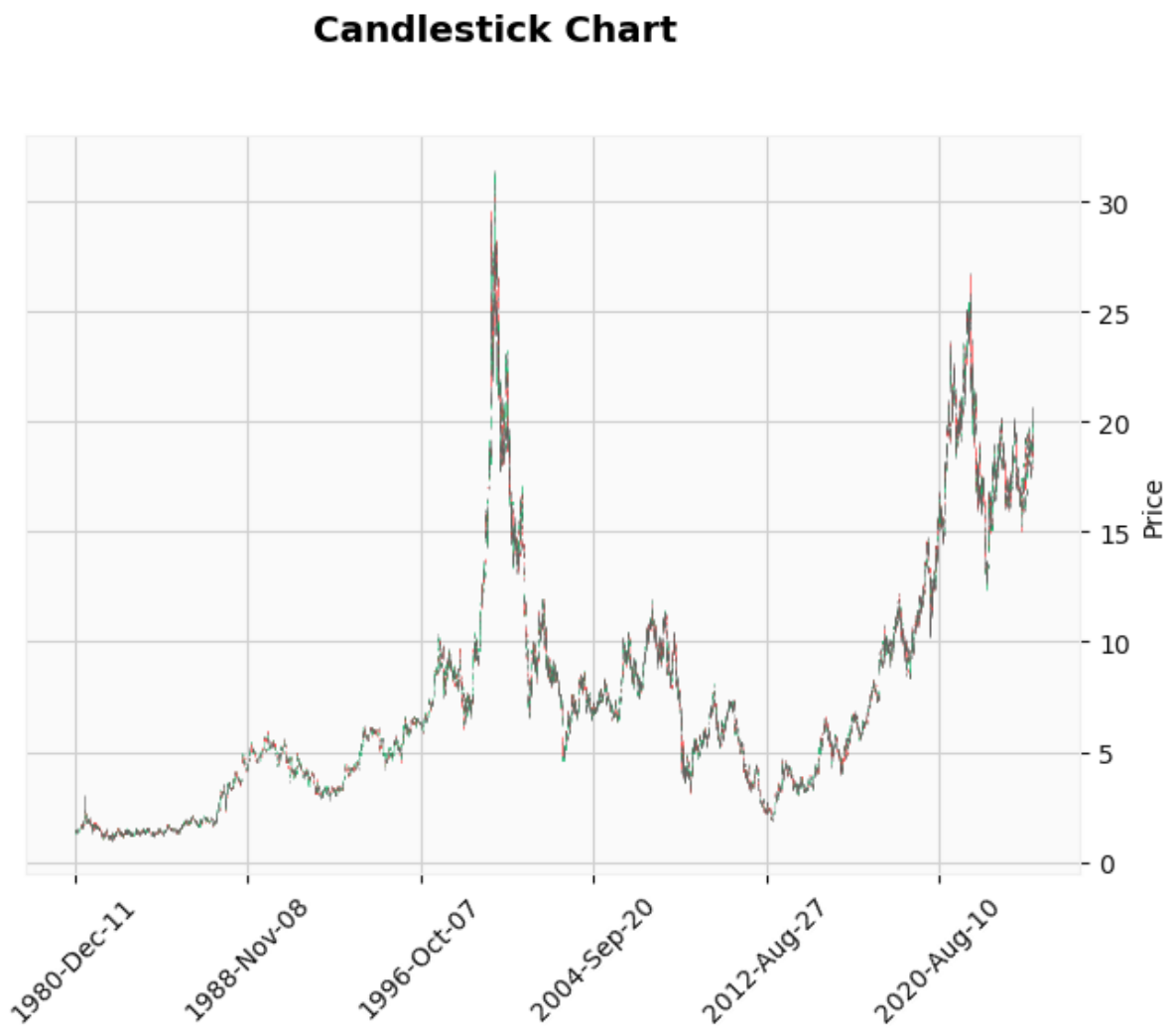


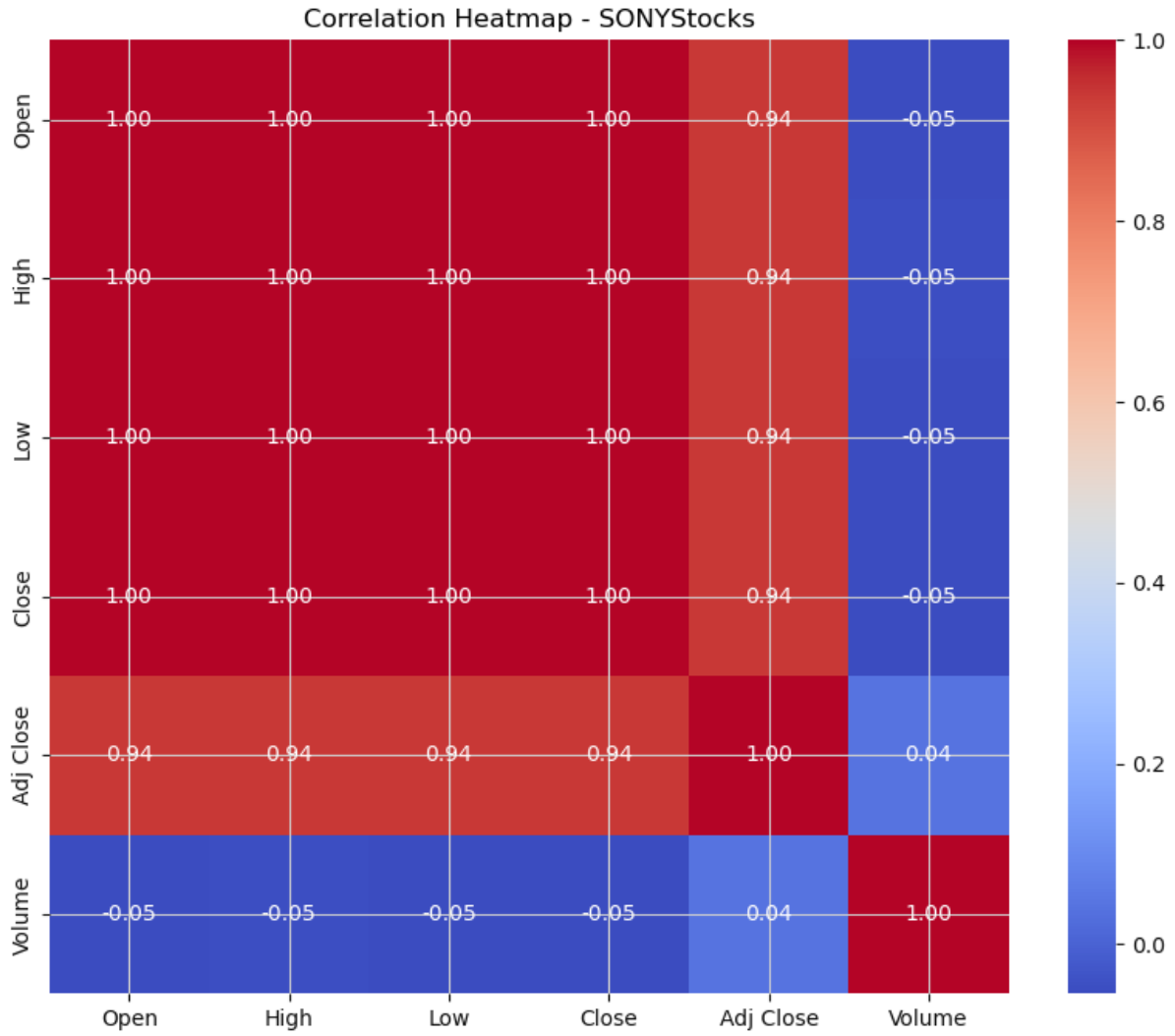
Then decided to plot pair plots, to understand the relations between all the features. But the most important relations pair were (Open, Close) and (High, Low)

Also Line chart for the “Closing Price” over the years were plotted. So we can compare how much a stock has increased over the years. This shows the stocking price from 2204 to 2024.

26/12/2024: Candlestick graph plotting and Heatmap

Illiyaz has knowledge about the candlestick graphs are the most accurate and best visualization for any stocks. So he researched about plotting candlestick graphs for datasets, and discussing with us and google search, he came across “mplfinance” module, which allows us to plot the candlestick graphs for the datasets. Hence he collected the syntax and applied for our dataset and got the candlestick graph.





28/12/2024: ML Algorithms application

This was the main and the difficult part of our project, where our team needs to decide what ML algorithms we should use and what is the best option for stock predictions. So we decided that each member will work on different ML Algorithms, will get the output results and graphs and then the best algorithm will be picked.

So I decided to work with Linear Regression model I searched about this model we run our project using this algorithm but I didn't get proper accuracy of stocking,so Hitesh decided to work on RandomForestClassifier, Kshitij decided to work on LSTM model. And illiyaz decided to work on prophet model. We compare every model so we decide to go through with Random Forest Classifier, LSTM and Prophet Model.

In this illiyaz transformed the data using MinMaxScaler, so that our ML algorithm works best. Once the scaling is done. After the MinMaxScaler he did splitting of the dataset into training and testing to calculate the Epoch. As the stock prediction is highly complex topic, we need to train the model with more data. Hence illiyaz decided to split the data into 80% training and 20% testing.

29/12/2024: Multiple error in code

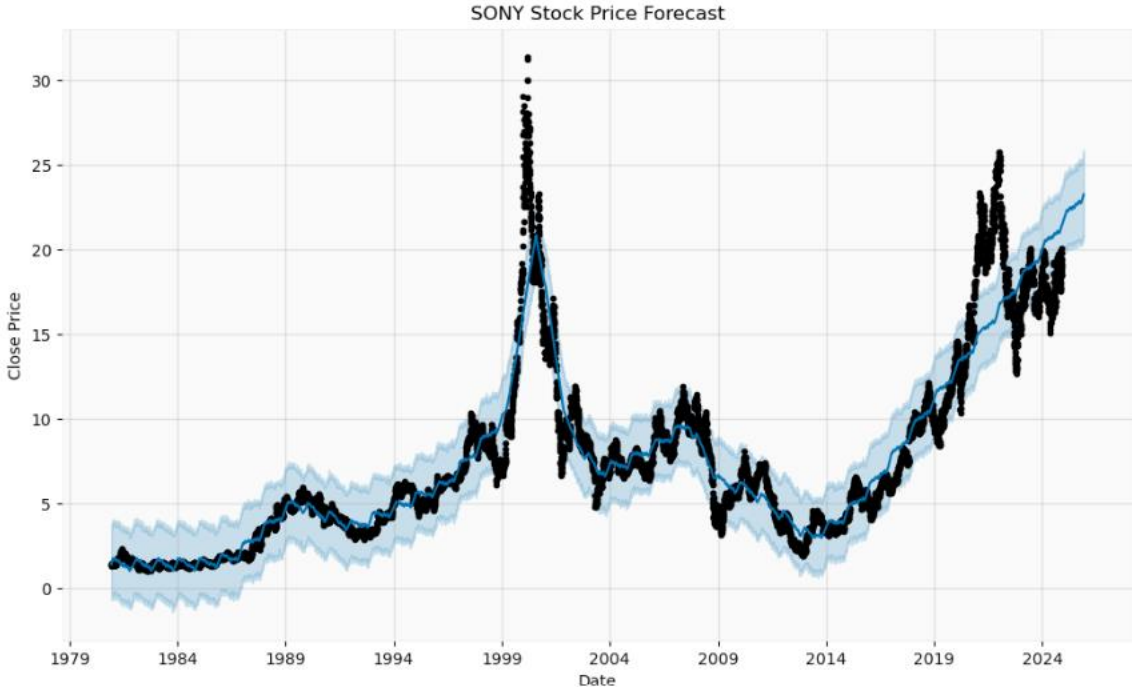
Illiyaz not having idea on how to use prophet model and how it works. So he decided to go through the module prophet and understand the concept. Prophet is an additive model-based method for time series data forecasting that fits non-linear trends with daily, weekly, and annual seasonality as well as holiday impacts. It is most effective when applied to time series with multiple seasons of historical data and significant seasonal influences. Prophet usually manages outliers effectively and is resilient to missing data and trend changes.

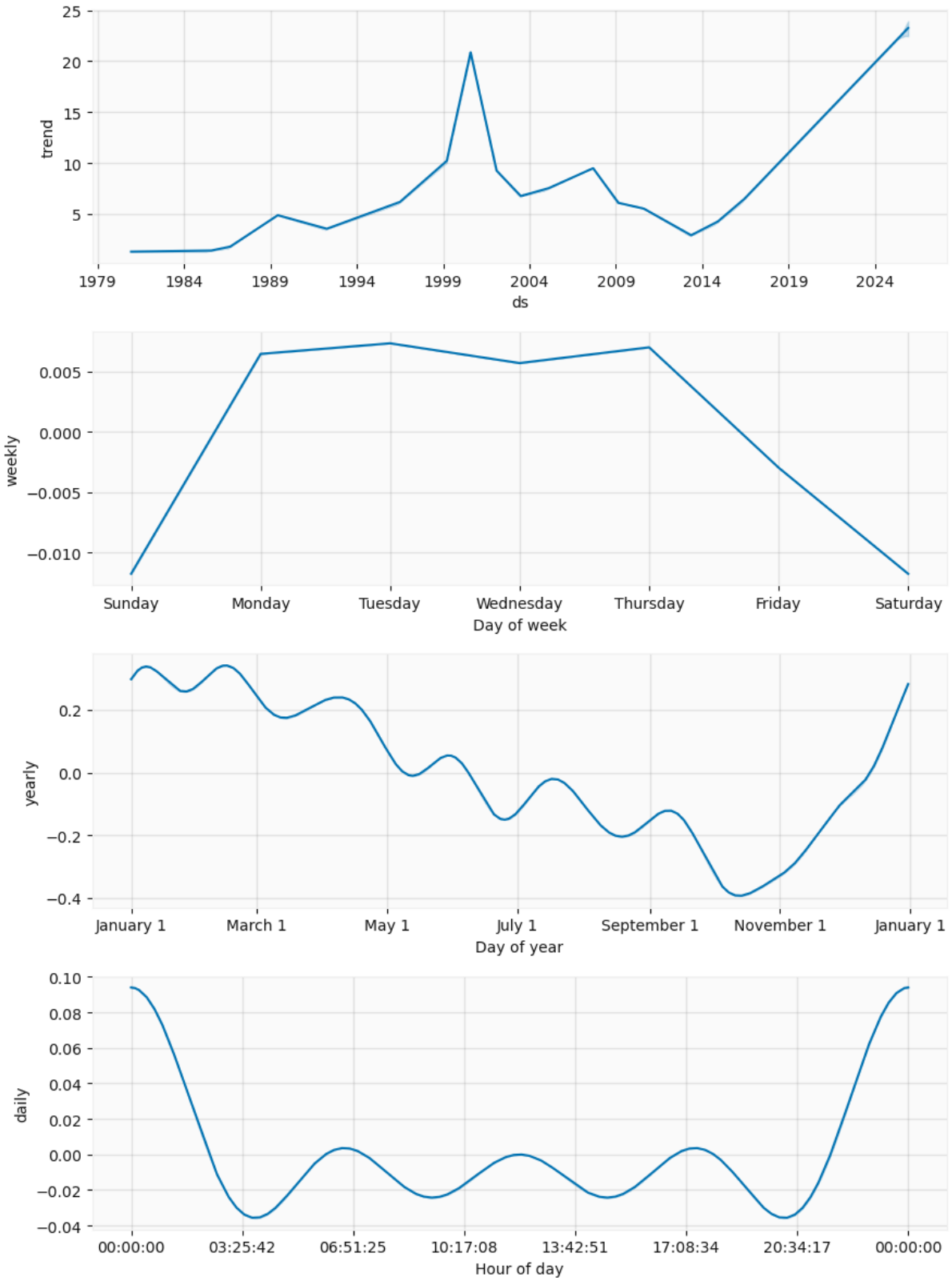
After understanding the prophet model, I got the syntax for this ML model from python official website. But got some errors while implementing it to my dataset. Then he came to knew that we have to do some datetime conversions and then proceed.

Illiyaz updated my dataframe accordingly and trained my model on this dataframe, and used "make_future_dataframe" which is inbuilt function of this prophet model, which gives us the prediction of future stock values and creates a new dataframe which predicts the stock prices for the next year.

I was also working on my dataset (SONY) using this prophet model and got Some graphs for the predicted trends were plotted and a new csv file will be saved which has the future trends and predictions.

```
#prophet ML algo for future stock prediction...

from prophet import Prophet

# Convert the 'Date' column to datetime format
df['Date'] = pd.to_datetime(df['Date'])

# Prepare data for Prophet (rename columns to 'ds' and 'y')
prophet_df = df[['Date', 'Close']].rename(columns={'Date': 'ds', 'Close': 'y'})

# Initialize the Prophet model
model = Prophet(daily_seasonality=True)

# Fit the model to the data
model.fit(prophet_df)

# Make a dataframe for future predictions
# Let's predict the next 365 days
future = model.make_future_dataframe(periods=365)

# Generate predictions
forecast = model.predict(future)
```


SONY Stock Price Forecast
Close Price
Date
30
25
20
15
10
5
0
1979
1984
1989
1994
1999
2004
2009
2014
2019
2024



02/01/2025: Working on PPT and report:

Everyone was ready with the code and outputs, but we decided to select best models which has better accuracy and predictions. Now we all knew what models were applied and which was the better working models. Hence we decided to work on report and ppt.

Everyone shared their inputs and research papers which we used for project this period. And Me and Hitesh was working on the PPT so we take information from illiyaz and kshitij and worked on the Introductory part, Literature Review, and Serched research papers that are related to our project Meanwhile Kshitij and illiyaz were working on methodology and results part. They shared all the necessary details of the methodologies so that they can append in the report.

03/01/2025: PPT

Presentation slides which involves the important points and highlights from the project was made by me and Hitesh. All the slides were added and necessary key points were highlighted.

04/01/2025: Report and ppt finalization:

Once the report and ppt were ready, we took a group call and reviewed all the work done by each member. Checked grammar mistakes and alignments.

05/01/2025: Aligning the code so that every dataset selected has been operated with similar techniques

After completing Report and PPT We decided that all the datasets used that must be aligned and the same set of operations should be applied so that there will be clarity of all the operations done and algorithms applied. And also the main reason for this is that the person who will check the code should understand and co-relate all the datasets and results.

Also we all uploaded the necessary files to the github and the link was added. Hitesh and Kshitij helped with the instructions file and the README file.

06/01/2025: Video presentation

All the team members did a teams call and illiyaz shared his screen and we started explaining the code and concept of our project. We also gave a demo by running the code.