COURSE : PROBABILITY AND STATISTICS

Team Members :

Vidya Varshini - 230041013
Vaishnavi - 230041016
Namaswi - 230041023
Bhavana - 230041026
Pranava - 230041037

PROJECT: A Statistical Analysis of Stock Market Trends

Introduction:

Probability and statistics are essential tools in stock market analysis, helping investors deal with uncertainty and make better decisions. Probability allows us to estimate the chances of events like price increases or decreases. Statistics, on the other hand, helps analyze past stock data to spot patterns, trends, and relationships between factors like stock prices and market conditions.

This analysis gives useful insights into how these companies' stocks are performing, helping to understand their price changes, volatility, and overall market trends more clearly.

In this project,

We performed a detailed analysis of historical stock data for two major companies: Apple (AAPL) and Microsoft (MSFT).

The data covers the period from January 1,2022 to January 1, 2023.Our primary focus was on three key aspects:

- Descriptive Statistics: Analyzing the overall behavior of stock prices, including average price, volatility, and key statistical metrics.
- Data Visualization: Using various charts and graphs to highlight significant trends and changes in stock prices over time.
- Probability Distribution Fitting: Applying statistical models to understand patterns in stock price movements and evaluate their consistency with known probability distributions.

Data Collection:

We used the `yfinance` library to download daily historical data for the stock symbols (AAPL and MSFT) between the dates `2022-01-01` and `2023-01-01`. The data includes:

- Date(Vary depending on required time period duration)
- Adjusted Close Price (used as stock price)
- Volume traded - It is the total number of financial assets that are bought and sold over a specific period of time.

We have written codes in python language,so that when we imported the datasets using finance library we calculated basic descriptive statistics(Mean,Variance,Standard Deviation Skewness,Kurtosis) and then we wrote codes for plotting different graphs(Line graph,Pie chart,Bar graph,Frequency

Polygon and Histogram,Cumulative Frequency Curves, Scatter Plot).For plotting these we imported matplotlib.pyplot library and seaborn library for making statistical graphics and visualizations easier.Then we fitted multiple probability distributions to the data using the method of moments(Normal Distribution,Uniform Distribution,Binomial Distributions,Poisson Distribution, Gamma Distribution,Exponential Distribution) and we wrote codes for plotting the distributions alongside the stock price histogram.

Imported Libraries:

```
import yfinance as yf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import norm, uniform, gamma, expon, skew, kurtosis
```

Descriptive Statistics in STOCK ANALYSIS:

Descriptive statistical tools like mean,median,mode,variance, standard variation,skewness, kurtosis and Coefficient of variation are essential in understanding the behavior of financial data.Ultimately, these methods help investors navigate the stock market’s ups and downs with better confidence and smarter decision-making.

Mean:

- A high mean suggests higher average prices over the given period.

- A low mean indicates that the stock has been trading at a lower average.

Median:

- It divides the dataset into two equal halves.
- If the mean is greater than the median, it suggests the presence of a few very high values (right skew or positive skew).
- If the mean is less than the median, it suggests the presence of very low values (left skew or negative skew).
- If the mean and median are close, the data may be symmetrically distributed (often, but not always, indicative of normal distribution).

Variance:

- High variance indicates that the stock prices are widely spread out from the average, meaning the stock experiences larger fluctuations or higher volatility.
- Low variance suggests the stock prices are closer to the average, indicating more stability with smaller price swings.

Standard deviation:

- Standard deviation gives an idea about the volatility or variability of the data.
- A low standard deviation means the values are clustered closely around the mean, indicating less volatility.
- A high standard deviation means the values are more spread out from the mean, indicating higher volatility indicating a more risk for traders.

Skewness:

- Skewness gives an idea about the direction of the data's asymmetry.
- Positive skew indicates that the stock has a few extreme high values, which could signal rare but significant price spikes.
- Negative skew indicates the presence of extreme low values, suggesting potential for sharp drops in prices.
- If skewness is close to zero, the distribution of stock prices is more symmetrical, implying balanced rises and falls in stock price.

Kurtosis:

- Kurtosis helps assess the probability of extreme outcomes (either very high or very low values).
- High kurtosis (leptokurtic) suggests the stock prices have more extreme deviations, meaning there could be frequent large price changes or volatility spikes.
- Low kurtosis (platykurtic) implies fewer extreme deviations, meaning the stock price is relatively stable and less prone to extreme jumps or drops.

Coefficient of Variance :

- The **coefficient of variance (CV)** helps us infer how much variation or dispersion exists relative to the mean of the data.
- A **higher CV** indicates greater variability, while a **lower CV** suggests the data points are more consistent and

closer to the mean. It's especially useful for comparing variability across datasets with different units or scales.

- A high CV indicates high variability in returns relative to the mean.

For calculating basic descriptive statistics:

```
# Calculate basic descriptive statistics
mean_price = stock['Stock Price'].mean()
variance_price = stock['Stock Price'].var()
std_dev_price = stock['Stock Price'].std()
skewness_price = skew(stock['Stock Price'])
kurtosis_price = kurtosis(stock['Stock Price'])
coefficient_of_variation = (std_dev_price / mean_price) * 100
```

Visualization of Stock Data:

We are visualizing the stock data through various graphs and diagrams to obtain various perspectives.

- **Line Graph**: Visualizes the stock price over time.
- **Pie Chart**: Shows the volume distribution by quarter, offering insights into seasonal trading patterns.
- **Bar Graph**: Illustrates the average stock price per month, highlighting any cyclical trends.
- **Frequency Polygon and Histogram**: Shows the distribution of stock prices and compares it to theoretical probability distributions.
- **Cumulative Frequency Curves**: Displays "less than" and "more than" cumulative stock price distributions.
- **Scatter Plot**: Plots stock price against trading volume, showing any relationship between the two.

To plot Line Graph - Stock Price over Time:

```
# Line Graph: Stock Price over time
plt.figure(figsize=(10, 6))
plt.plot(stock['Date'], stock['Stock Price'], color='blue')
plt.title(f'{stock_symbol} Stock Price Over Time')
plt.xlabel('Date')
plt.ylabel('Stock Price')
plt.xticks(rotation=45)
plt.show()
```

To plot Pie Chart - Proportion of volume traded by quarters - 3 months:

```
# Pie Chart: Visualizing the proportion of volume traded by quarters
stock['Quarter'] = stock.index.to_period("Q")  # Create a 'Quarter' column based on the date index
quarterly_volume = stock.groupby('Quarter')['Volume'].sum()  # Sum the volume for each quarter
# Step 4: Plot pie chart for volume distribution by quarters
plt.figure(figsize=(7, 7))
quarterly_volume.plot(kind='pie', autopct='%1.1f%%', colors=['lightblue', 'lightgreen', 'lightcoral', 'gold'],
                      legend=False, labels=quarterly_volume.index)
plt.title(f'{stock_symbol} Volume Distribution by Quarter')
plt.ylabel('')
plt.show()
```

To plot Bar Graph - Average stock Price per month:

```
# Bar Graph: Average Stock Price per Month
stock['Month'] = stock.index.month
avg_price_by_month = stock.groupby('Month')['Stock Price'].mean()

plt.figure(figsize=(10, 6))
plt.bar(avg_price_by_month.index, avg_price_by_month.values, color='lightblue', edgecolor='black')
plt.title(f'{stock_symbol} Average Stock Price by Month')
plt.xlabel('Month')
plt.ylabel('Average Stock Price')
plt.xticks(ticks=np.arange(1, 13), labels=['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
plt.show()
```

To plot Histogram and Frequency Polygon over the histogram:

```
# Frequency Polygon (overlay on histogram)
plt.figure(figsize=(10, 6))
plt.hist(stock['Stock Price'], bins=30, density=True, alpha=0.6, color='lightblue', edgecolor='black', label='Stock Price Data')
sns.kdeplot(stock['Stock Price'], color='red', label='Frequency Polygon')
plt.title(f'{stock_symbol} Histogram and Frequency Polygon')
plt.xlabel('Stock Price')
plt.ylabel('Density')
plt.legend()
plt.show()
```

To plot both less than and more than type Cumulative Frequency Curves:

```
# Cumulative Frequency (both less than and more than type)
sorted_prices = np.sort(stock['Stock Price'])
cumulative_freq_less_than = np.cumsum(np.ones_like(sorted_prices)) / len(sorted_prices)
cumulative_freq_more_than = 1 - cumulative_freq_less_than
plt.figure(figsize=(10, 6))
plt.plot(sorted_prices, cumulative_freq_less_than, label='Cumulative Less Than', color='blue')
plt.plot(sorted_prices, cumulative_freq_more_than, label='Cumulative More Than', color='red')
plt.title(f'{stock_symbol} Cumulative Frequency Curves')
plt.xlabel('Stock Price')
plt.ylabel('Cumulative Frequency')
plt.legend()
plt.show()
```

Scatter Plot : Stock Price vs volume:

```
# Scatter Plot: Stock Price vs Volume
plt.figure(figsize=(10, 6))
plt.scatter(stock['Volume'], stock['Stock Price'], alpha=0.5, color='purple')
plt.title(f'{stock_symbol} Stock Price vs Volume')
plt.xlabel('Volume')
plt.ylabel('Stock Price')
plt.show()
```

Probability Distribution Fitting:

We used the method of moments to fit several probability distributions to the stock price data:

- Normal Distribution: Fitted using the mean and variance of the stock prices. Provides a baseline comparison for how stock prices behave in a "normal" scenario.
- Uniform Distribution: Assumes all price values within a range are equally likely.
- Binomial Distribution: Models the price as an outcome of a series of independent trials, estimating parameters from stock price data.
- Poisson Distribution: Suitable for modeling rare events, fitted based on the average stock price.
- Gamma Distribution: Captures skewed data, often used to model variables that are always positive.This distribution often fits stocks with asymmetric price behaviors, where the likelihood of small values is higher but with a long tail of extreme high values. Such stocks may have a high potential for large price jumps.
- Exponential Distribution: Stocks fitting an exponential distribution may suggest that prices tend to decay rapidly, with a few instances of higher values. This could be typical in stocks with a declining trend or where low prices dominate.

Fitting Distributions using Method of moments:

```python
#  Fit distributions using method of moments

# 1. Normal distribution (only uses mean and variance)
mean_norm, std_dev_norm = norm.fit(stock['Stock Price'])
# 2. Uniform distribution (mean and variance define it)
max_uniform,min_uniform = uniform.fit(stock['Stock Price'])
# 3. Gamma distribution (uses mean and variance)
shape_gamma, loc_gamma, scale_gamma = gamma.fit(stock['Stock Price'], floc=0)
# 4. Exponential distribution (mean equals 1/λ)
lambda_exp = 1 / mean_price
```

Plotting the distributions alongside the stock price histogram:

```python
# Step 4: Plot the distributions alongside the stock price histogram
# Generate x values for the fitted distributions
x = np.linspace(stock['Stock Price'].min(), stock['Stock Price'].max(), 1000)
# Plot stock price histogram
plt.figure(figsize=(10, 6))
plt.hist(stock['Stock Price'], bins=30, density=True, alpha=0.6, color='lightblue', edgecolor='black', label='Stock Price Data')
# Plot fitted normal distribution
plt.plot(x, norm.pdf(x, mean_norm, std_dev_norm), label=f'Normal (mean={mean_norm:.2f}, std={std_dev_norm:.2f})', color='red')
# Plot fitted uniform distribution
plt.plot(x, uniform.pdf(x, min_uniform, max_uniform - min_uniform), label=f'Uniform (min={min_uniform:.2f}, max={max_uniform:.2f})',color='blue')
# Plot fitted gamma distribution
plt.plot(x, gamma.pdf(x, shape_gamma, loc_gamma, scale_gamma), label=f'Gamma (shape={shape_gamma:.2f})', color='brown')
# Plot fitted exponential distribution
plt.plot(x, expon.pdf(x, scale=1/lambda_exp), label=f'Exponential (λ={lambda_exp:.2f})', color='orange')
```

To evaluate the goodness of fit for each distribution:

```python
# Normal distribution moments
print(f"Normal - Mean: {mean_norm:.2f}, Variance: {std_dev_norm**2:.2f}")
# Uniform distribution moments
mean_uniform = (min_uniform + max_uniform) / 2
variance_uniform = (max_uniform - min_uniform)**2 / 12
print(f"Uniform - Mean: {mean_uniform:.2f}, Variance: {variance_uniform:.2f}")
# Gamma distribution moments
mean_gamma = shape_gamma * scale_gamma
variance_gamma = shape_gamma * scale_gamma**2
print(f"Gamma - Mean: {mean_gamma:.2f}, Variance: {variance_gamma:.2f}")
# Exponential distribution moments
mean_exp = 1 / lambda_exp
variance_exp = 1 / lambda_exp**2
print(f"Exponential - Mean: {mean_exp:.2f}, Variance: {variance_exp:.2f}")
```

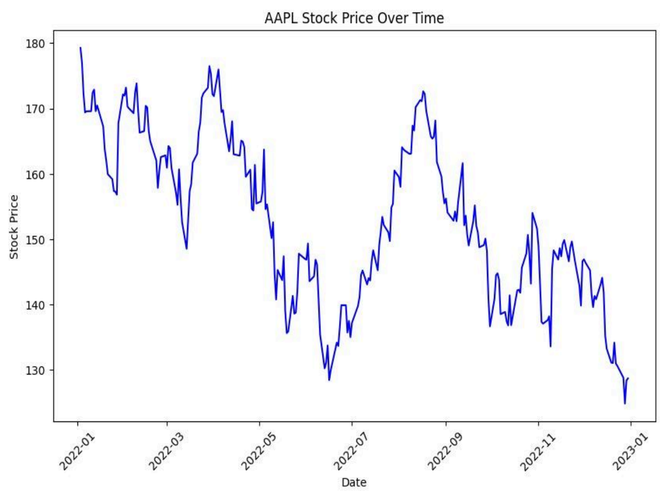
Based on the data we have taken,

For Apple (AAPL) company :

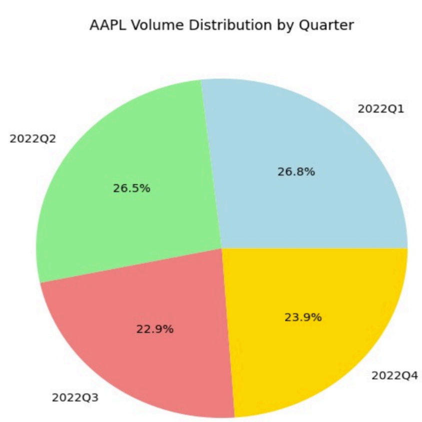
Basic Descriptive statistics:

```
Basic Descriptive Statistics for AAPL:
Mean: 152.94, Variance: 161.95, Std Dev: 12.73
Skewness: 0.01, Kurtosis: -1.01
Coefficient of Variance: 8.32
```

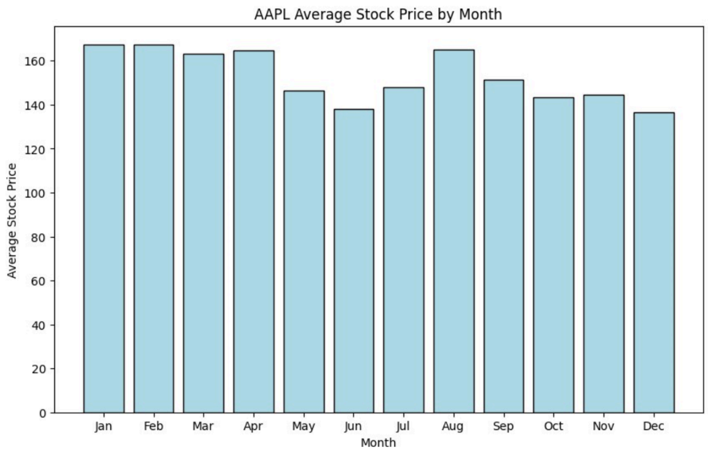
Line Graph:



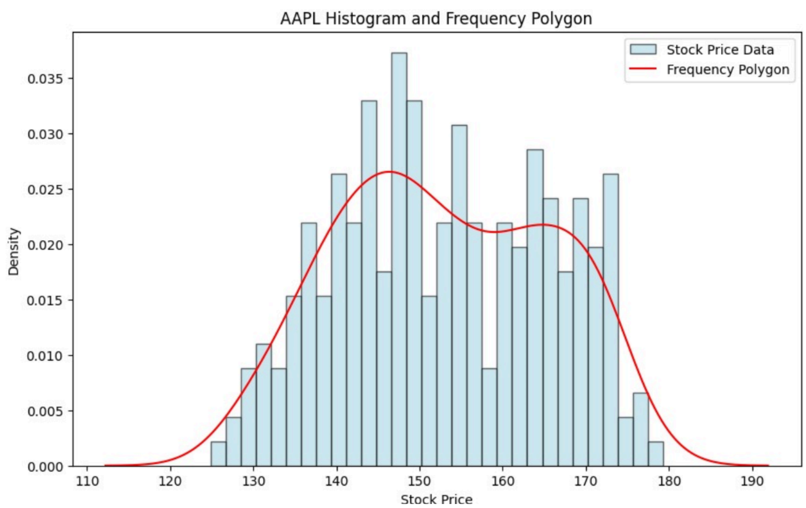
Pie Chart:



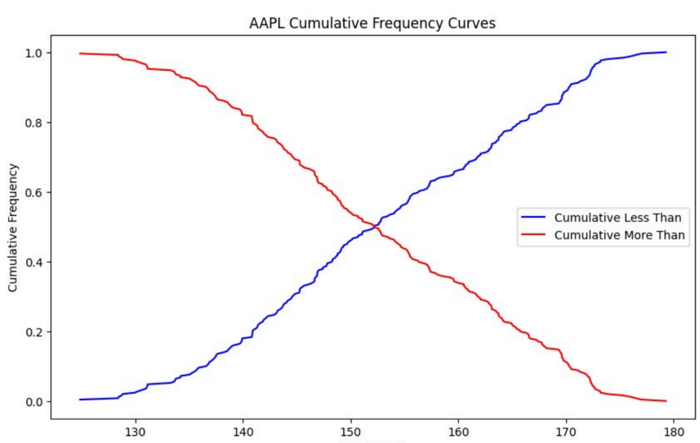
Bar Graph:



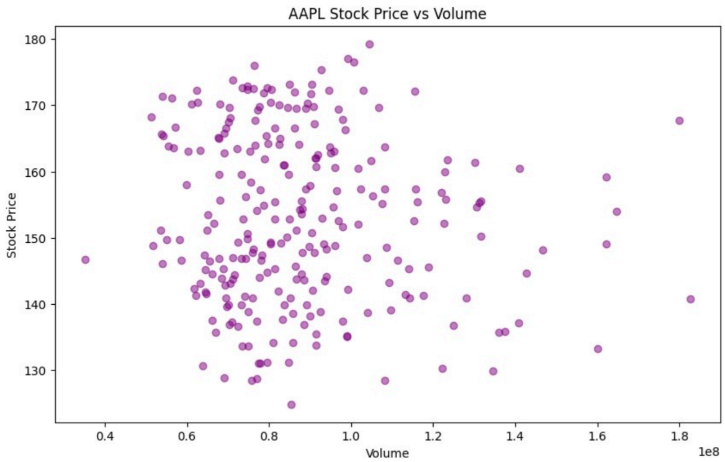
Histogram and Frequency Polygon:



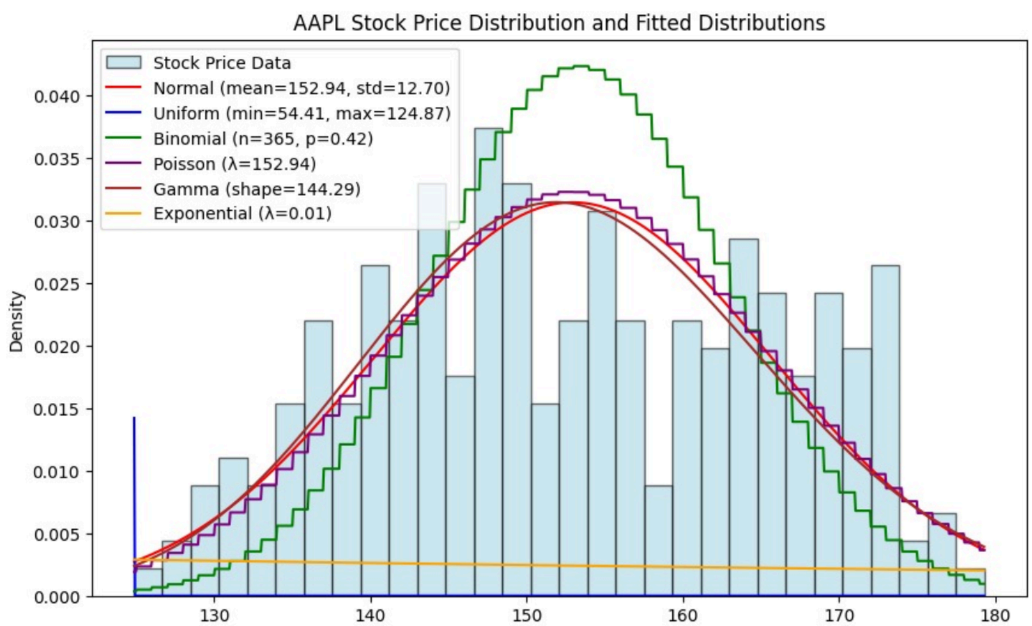
Cumulative Frequency Curves:



Scatter Plot:



Fitted Distributions over histogram of stock price vs density:



Fitting Distribution Moments:

```
Fitted Distributions Moments:
Normal - Mean: 152.94, Variance: 161.30
Uniform - Mean: 89.64, Variance: 413.69
Gamma - Mean: 152.94, Variance: 162.12
Exponential - Mean: 152.94, Variance: 23391.69
```

KEY INFERENCES:

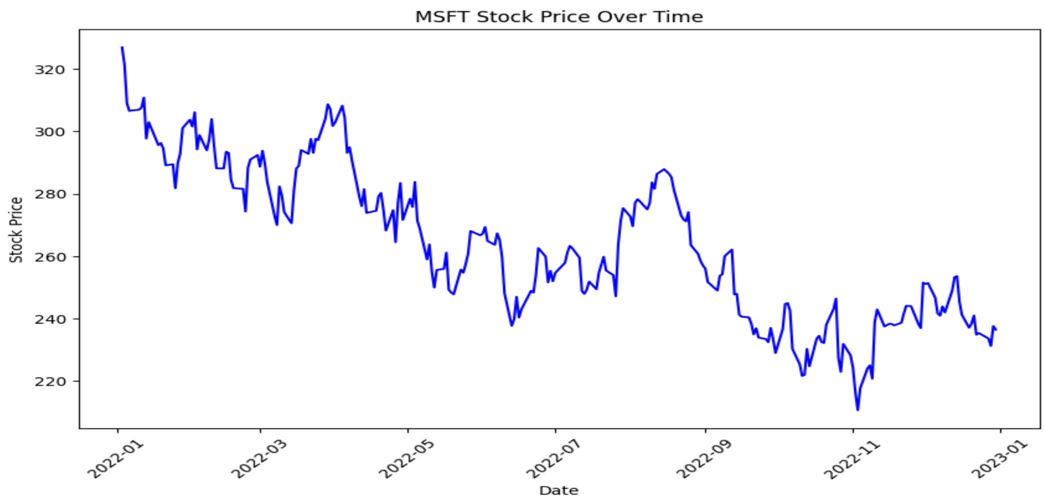
- The **Gamma distribution** fits best the data well as seen by the close match in mean and variance and even the normal distributions fit the data.
- The uniform distribution isn’t usually used for stock prices because it assumes all prices in a range are equally likely, which isn't how stock prices behave. Stock prices often move in patterns, with some prices happening more often than others.
- The variance reflects the spread of the data around the mean.
- Skewness close to 0 indicates the data is nearly symmetric.
- Kurtosis of -1.01 implies a flatter distribution than normal (less likelihood of extreme highs or lows).
- The Line graph shows Apple’s stock price trends in 2022. The price starts near $180, experiences several ups and downs, and steadily declines, ending the year around $130.This reflects a challenging year for AAPL, marked by market fluctuations and an overall downward trend.
- This pie chart shows the quarterly distribution of AAPL's trading volume in 2022. The volume was fairly consistent, with a slight increase in Q1 and Q2, while Q3 saw the lowest activity. Overall, there was no drastic variation in volume throughout the year.

For Microsoft (MSFT) Company:

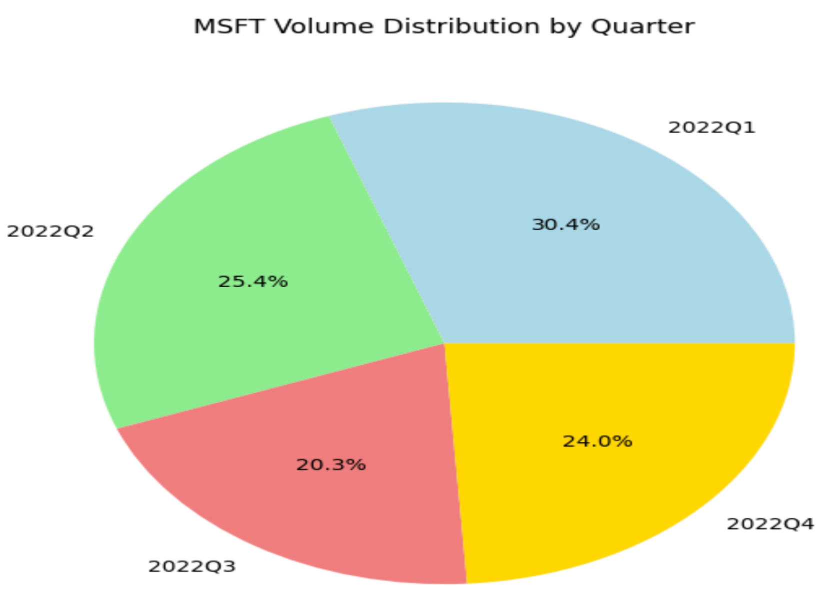
Basic Descriptive Statistics:

```
Basic Descriptive Statistics for MSFT:
Mean: 263.75, Variance: 608.04, Std Dev: 24.66
Skewness: 0.19, Kurtosis: -0.92
Coefficient of Variance: 9.35
```

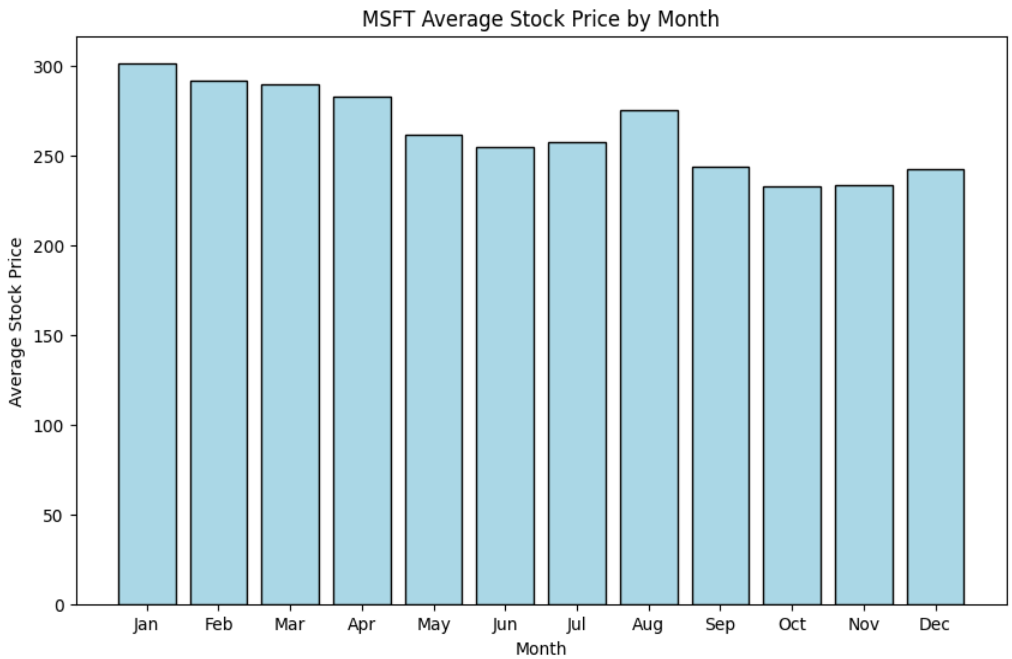
Line Graph:



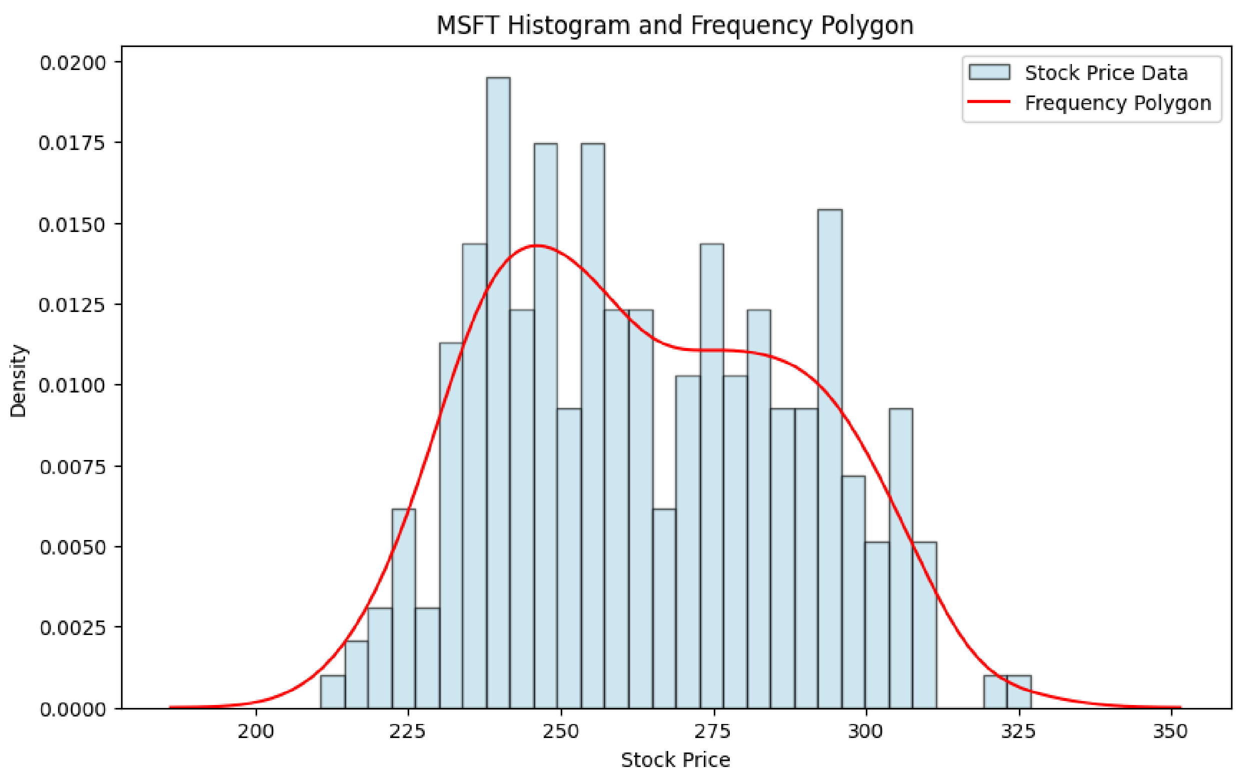
Pie Chart:



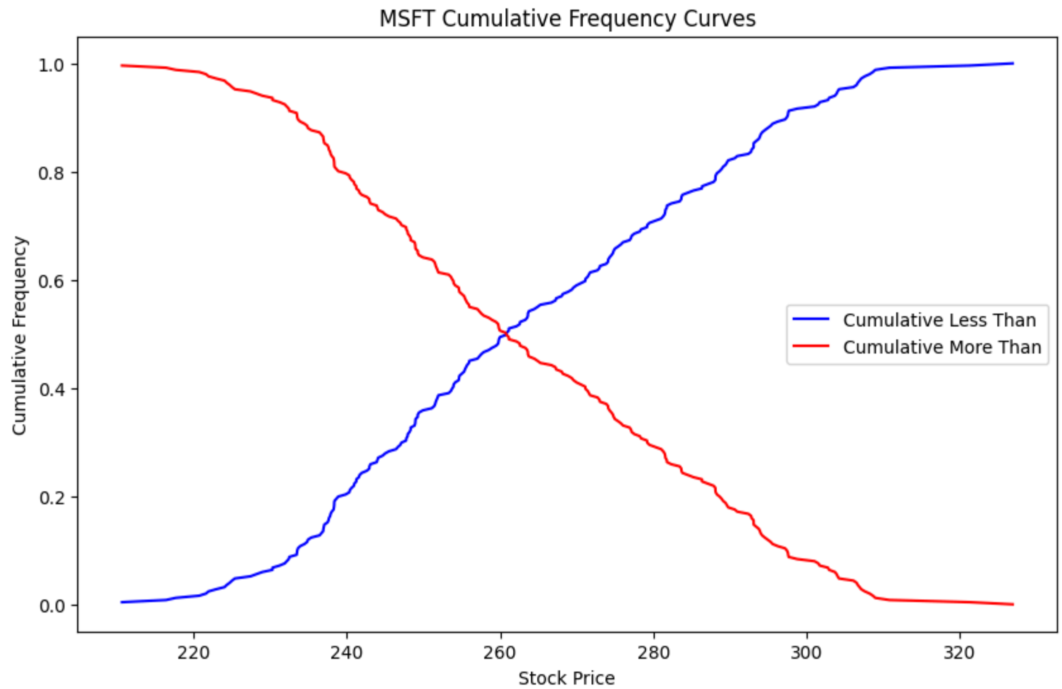
Bar Graph:



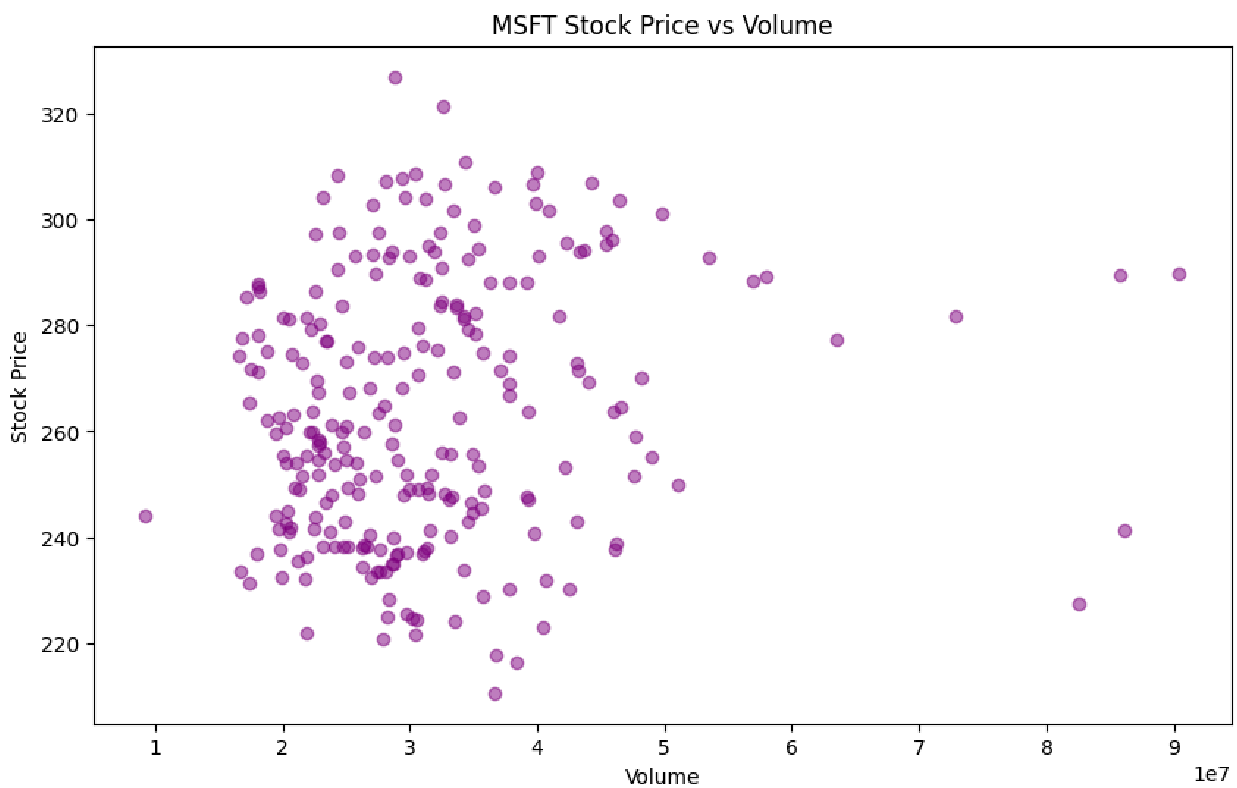
Histogram and Frequency Polygon:



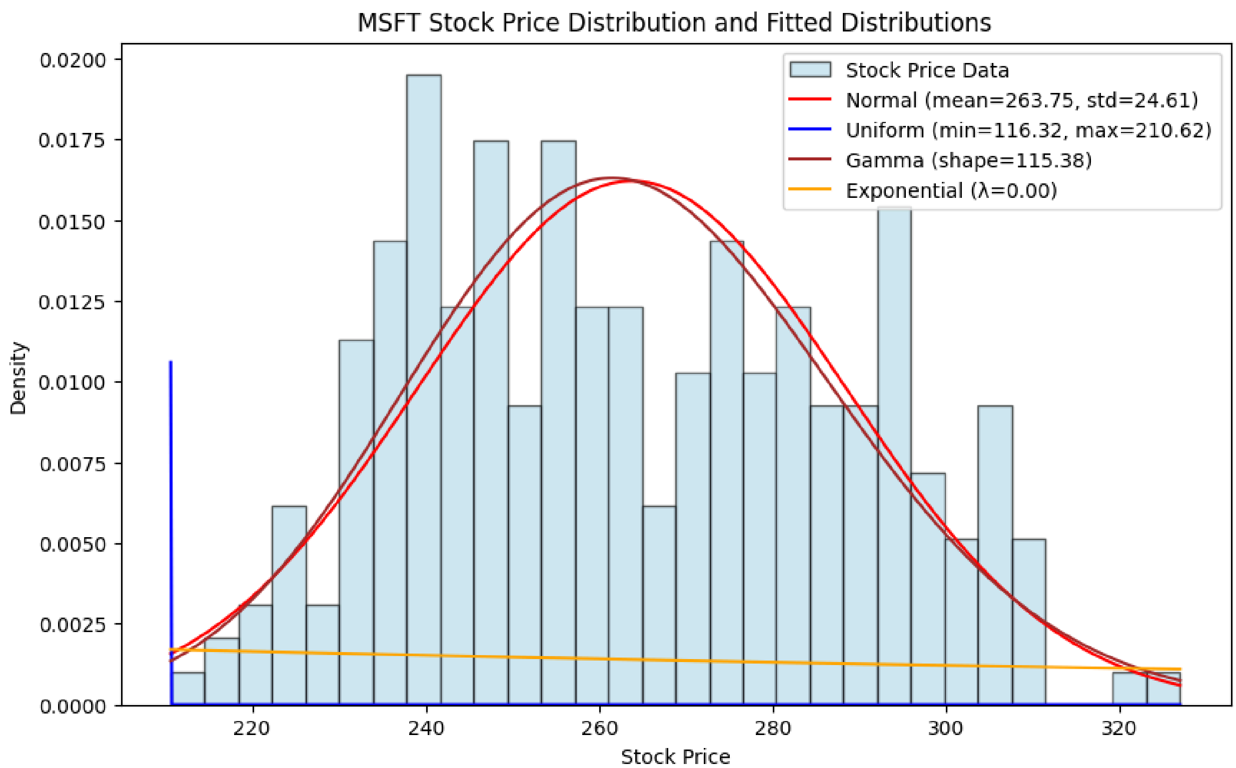
Cumulative Frequency Curves:



Scatter Plot:



Fitted Distributions over histogram of stock price vs density:



Fitted Distribution Moments:

```
Fitted Distributions Moments:
Normal - Mean: 263.75, Variance: 605.62
Uniform - Mean: 163.47, Variance: 741.00
Gamma - Mean: 263.75, Variance: 602.89
Exponential - Mean: 263.75, Variance: 69563.81
```

KEY INFERENCES:

- Variance reflects the spread of stock prices, with values deviating from the mean moderately.

- The skewness of 0.19 suggests the distribution is nearly symmetrical, with a slight tilt to the right (higher values are just a bit more common).
- Kurtosis of -0.92 implies a flatter distribution than normal (less likelihood of extreme highs or lows).
- The **Normal distribution** fits the stock price data best.The exponential and uniform distributions are poor fits.
- MSFT experienced a consistent **downward trend** throughout the year.There were some minor ups and downs in the starting of the year that later experienced a consistent decline.
- The pie chart suggests that trading activity was more concentrated in the first half of the year, with slight reductions in the latter half, highlighting potential seasonality or external market factors influencing trade volume across quarters.

Practical Implications:

- This analysis can be used to make informed decisions regarding stock purchases.Understanding which probability distribution a stock fits can aid in estimating future stock price behaviors and potential risks.
- Descriptive statistics provide a quick way to assess the basic performance and volatility of stocks. Higher

volatility stocks may offer higher potential returns, but at a greater risk.

- Seasonal or quarterly trends in volume can be used for strategic trading. For instance, traders might focus on times when trading volume is higher, suggesting more liquidity or potential market-moving events.
- This project demonstrates how statistical and visualization techniques can help analyze stock data, providing valuable insights for investors about stock trends, risk, and future price behavior. By combining descriptive statistics and distribution fitting, we can make both descriptive and predictive conclusions about the stock market.

CONCLUSION:

Through this project we have gained knowledge across the stock market analysis like basically we applied what we learned till date in this course to analyze stocks of companies Microsoft and Apple.We divided the tasks in writing code and gaining the required information in order to analyze the distributions.We also referred to some of the research papers and some other websites to get insights of the stock market analysis and to implement the codes.

LINK OF THE CODE:

Stock Analysis.ipynb