**ADS-1 Assignment-1**

Name: Veera Raghunatha Reddy Naguru

Student Id: 22028322

Submitted to: University of Hertfordshire

Data link: <https://www.kaggle.com/datasets/aayushmishra1512/faang-complete-stock-data>

Repository link: <https://github.com/veeraraghu/22028322_ADS-1_Visualisation.git>

Code link in repository: <https://github.com/veeraraghu/22028322_ADS-1_Visualisation/blob/32b877079760fa4436d0de76d4f7826c5ce2c0cd/main_code.py>

**Visualization 1: Stock Performance over time between 2013 and 2020**

Chart, histogram

Description automatically generated

Over years from 2013 to 2020, Amazon stock price increased from 300$ to above 3000$. Amazon stock price raised above Google stock price around 2017. We can see an exponential growth in Amazon stock price after 2020 with a slight downfall.

If we see just before 2019 and after 2020 there is a common downfall in all stocks. This may be due to a social cause which might have affected the overall economy.

We can see that all stocks price increased over years and can be trusted to invest for long time.

**Visualization 2: Distributions of Daily Returns of each Company**

Chart, box and whisker chart

Description automatically generated

By looking at the distributions of returns of each company, we can notice that the loss and profit on daily returns are of similar percentile. It is very risky to trade these stocks daily, since the loss and profit on daily returns are unpredictable. So, it is not recommended to trade these stocks on daily basis.

**Visualization 3: Total Volume of Shares sold or traded by Company.**Chart, bar chart

Description automatically generated

By looking at the bar plot, we can say that the most traded stock is Apple with 447614.1584Millions of shares sold or traded between 18th May 2012 and 3rd Aug 2020. With this data we can say that the Apple Company is a true interest to many traders.

Since the Volume is very high, there is a chance that many traders are trading on daily return basis for this company stock.

Assignment Code:

*# -\*- coding: utf-8 -\*-*

*"""*

*Student Id:22028322*

*Name:Veera Raghunatha Reddy Naguru*

*FAANG Stock Data visualisation*

*Data collected from Kaggle and link to download data is given below:*

*https://www.kaggle.com/datasets/aayushmishra1512/faang-complete-stock-data*

*"""*

*# Imported required libraries*

*import pandas as pd*

*import numpy as np*

*import matplotlib.pyplot as plt*

*# Defining Functions*

*# These functions are made by keeping common stocks data caolumn names*

*# in reference.*

*def data\_trim\_by\_date(start\_date, end\_date, data):*

*'''*

*This function creates a Dataframe with records of data between given*

*start\_date and end\_date.*

*Parameters*

*----------*

*start\_date : STR*

*Enter starting date of your data to trim, in string format.*

*end\_date : STR*

*Enter end date of your data to trim, in string format.*

*data : pandas.DataFrame*

*data['Date'] is our target column to perfom boolean operation*

*with "start\_date" and "end\_date" to get selective data.*

*Make sure to check column names before calling this function.*

*Returns*

*-------*

*pandas.DataFrame*

*This function will return data recorded between*

*"start\_date" and "end\_date" with its index being reset .*

*'''*

*new\_data = data[(data['Date'] >= start\_date) & (data['Date'] < end\_date)]*

*new\_data = new\_data.reset\_index(drop=True)*

*return new\_data*

*def returns(data):*

*'''*

*This function will create a single column DataFrame with return percentage*

*or the change in share value ('Close') with the previous day share value.*

*Output data is in percentage.*

*Parameters*

*----------*

*data : pandas.DataFrame*

*data['Close'] is our target column to calculate daily returns.*

*Make sure to check column names before calling this function.*

*Returns*

*-------*

*returns : pandas.DataFrame*

*This function will return percentage change in Close value with*

*previous day Close value as a DataFrame.*

*'''*

*returns = data['Close'].pct\_change()\*100*

*return returns*

*def volume(data):*

*'''*

*This function will calculate the total volume of shares sold or traded by*

*the company that is recorded in the given data.*

*Parameters*

*----------*

*data : pandas.DataFrame*

*data['Volume'] is our target column to perform sum operation to get*

*total shares sold or traded by the company.*

*Make sure to check column names before calling this function.*

*Returns*

*-------*

*total\_volume : Float*

*This function will return total volume of shares sold or traded by*

*the company in millions as an integer.*

*'''*

*total\_volume = data['Volume'].sum() / 1000000*

*return total\_volume*

*# Reading companies stock data*

*amzn = pd.read\_csv('Amazon.csv')*

*apple = pd.read\_csv('Apple.csv')*

*fbook = pd.read\_csv('Facebook.csv')*

*google = pd.read\_csv('Google.csv')*

*netflix = pd.read\_csv('Netflix.csv')*

*# Looking at the top rows of all data*

*print('Amazon data', '\n', amzn.head(), '\n')*

*print('Apple data', '\n', apple.head(), '\n')*

*print('Facebook data', '\n', fbook.head(), '\n')*

*print('Google data', '\n', google.head(), '\n')*

*print('Netflix data', '\n', netflix.head(), '\n')*

*'''*

*Looking at all the top rows of data,*

*we can clearly say that the columns are similar in all the dataset.*

*But, the date values in all tables are different*

*'''*

*# Shape of Data of each company stock*

*print('Rows and columns in Amazon stocks data is ', amzn.shape)*

*print('Rows and columns in Apple stocks data is ', apple.shape)*

*print('Rows and columns in Facebook stocks data is ', fbook.shape)*

*print('Rows and columns in Google stocks data is ', google.shape)*

*print('Rows and columns in Netflix stocks data is ', netflix.shape, '\n')*

*'''*

*Since the number of records of all companies data is not same,*

*we need to find a timeperiod where all companies stock records are available*

*'''*

*# start\_date finds the starting date from where all companies data is recorded*

*start\_date = max(amzn['Date'].min(), apple['Date'].min(),*

*fbook['Date'].min(), google['Date'].min(),*

*netflix['Date'].min())*

*# end\_date finds the ending date until where all companies data is recorded*

*end\_date = min(amzn['Date'].max(), apple['Date'].max(),*

*fbook['Date'].max(), google['Date'].max(),*

*netflix['Date'].max())*

*print('We have stocks data recorded of all companies from ', start\_date,*

*'until ', end\_date, '\n')*

*# Creating new variables for data between the start\_data and end\_data*

*new\_amzn = data\_trim\_by\_date(start\_date, end\_date, amzn)*

*new\_apple = data\_trim\_by\_date(start\_date, end\_date, apple)*

*new\_fbook = data\_trim\_by\_date(start\_date, end\_date, fbook)*

*new\_google = data\_trim\_by\_date(start\_date, end\_date, google)*

*new\_netflix = data\_trim\_by\_date(start\_date, end\_date, netflix)*

*# Data is examined to check the start dates where data is recorded in of a year*

*# new\_amzn.to\_excel('new\_amzn.xlsx')*

*# Shape of Data of each company stock*

*print('Rows and columns in Amazon stocks data is ', new\_amzn.shape)*

*print('Rows and columns in Apple stocks data is ', new\_apple.shape)*

*print('Rows and columns in Facebook stocks data is ', new\_fbook.shape)*

*print('Rows and columns in Google stocks data is ', new\_google.shape)*

*print('Rows and columns in Netflix stocks data is ', new\_netflix.shape, '\n')*

*'''*

*We can see that all the datasets have same rows and columns*

*'''*

*# Plotting company share performance*

*plt.figure(figsize=(10, 6))*

*plt.plot(new\_amzn['Date'], new\_amzn['Close'], label='Amazon')*

*plt.plot(new\_apple['Date'], new\_apple['Close'], label='Apple')*

*plt.plot(new\_fbook['Date'], new\_fbook['Close'], label='Facebook')*

*plt.plot(new\_google['Date'], new\_google['Close'], label='Google')*

*plt.plot(new\_netflix['Date'], new\_netflix['Close'], label='Netflix')*

*plt.xlabel('Year')*

*plt.ylabel('Daily Close in US$')*

*plt.title('Stock Performance over time')*

*# These dates are the first working days of years from 2013 to 2020*

*# and where our stocks data is recorded.*

*plt.xticks(['2013-01-02', '2014-01-02', '2015-01-02', '2016-01-04',*

*'2017-01-03', '2018-01-02', '2019-01-02', '2020-01-02'],*

*[2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020])*

*plt.legend(loc='upper left')*

*plt.savefig('Stock\_Performance\_over\_time.png')*

*plt.show()*

*'''*

*By looking at the line plot,*

*we can say that the Amazon stock value is increased than the value of Apple*

*'''*

*# Calculationg returns*

*new\_amzn['Returns'] = returns(new\_amzn)*

*new\_apple['Returns'] = returns(new\_apple)*

*new\_fbook['Returns'] = returns(new\_fbook)*

*new\_google['Returns'] = returns(new\_google)*

*new\_netflix['Returns'] = returns(new\_netflix)*

*# Plotting distributions of return percentage of each company*

*plt.figure(figsize=(10, 10))*

*plt.suptitle('Distributions of Daily Returns of each Company', fontsize=20)*

*plt.subplot(3, 2, 1)*

*plt.hist(new\_amzn['Returns'], bins=20, label='Amazon',*

*density=True, color='c')*

*plt.title('Amazon')*

*plt.xlabel('Returns Percentage Distribution')*

*plt.grid(True)*

*plt.legend()*

*plt.subplot(3, 2, 2)*

*plt.hist(new\_apple['Returns'], bins=20, label='Apple',*

*density=True, color='g')*

*plt.title('Apple')*

*plt.xlabel('Returns Percentage Distribution')*

*plt.grid(True)*

*plt.legend()*

*plt.subplot(3, 2, 3)*

*plt.hist(new\_fbook['Returns'], bins=20, label='Facebook',*

*density=True, color='b')*

*plt.title('Facebook')*

*plt.xlabel('Returns Percentage Distribution')*

*plt.grid(True)*

*plt.legend()*

*plt.subplot(3, 2, 4)*

*plt.hist(new\_google['Returns'], bins=20, label='Google',*

*density=True, color='m')*

*plt.title('Google')*

*plt.xlabel('Returns Percentage Distribution')*

*plt.grid(True)*

*plt.legend()*

*plt.subplot(3, 2, 5)*

*plt.hist(new\_netflix['Returns'], bins=20, label='Netflix',*

*density=True, color='r')*

*plt.title('Netflix')*

*plt.xlabel('Returns Percentage Distribution')*

*plt.grid(True)*

*plt.legend()*

*plt.tight\_layout(pad=1)*

*plt.savefig('Distributions\_of\_Daily\_Returns\_of\_each\_Company.png')*

*plt.show()*

*print('All stocks have a 50% - 60% chance of daily profit')*

*# Calculating total volume of shares traded or sold*

*total\_volume = [volume(new\_amzn), volume(new\_apple), volume(new\_fbook),*

*volume(new\_google), volume(new\_netflix)]*

*companies = ['Amazon', 'Apple', 'Facebook', 'Google', 'Netflix']*

*# Plotting total volume of shares over company in millions*

*plt.figure(figsize=(10, 6))*

*bars = plt.bar(companies, total\_volume, color='g')*

*plt.xlabel('Company')*

*plt.ylabel('Shares sold/traded in millions')*

*plt.title('Total Volume Shares sold/traded by Company')*

*plt.bar\_label(bars)*

*plt.savefig('Total\_Volume\_Shares\_sold\_traded\_by\_Company.png')*

*plt.show()*

*print('Most traded stock is ', companies[np.argmax(total\_volume)], ' with ',*

*max(total\_volume), ' Millions of shares sold or traded.')*