6401: Visualization of Complex Data

Spring 2022

Instructor: R. Jafari, Ph.D

Lab #2

Student: Lydia Teinfalt

Initials: LT

Last updated: February 2, 2022

Contents

Abstract

Yahoo stock API was used as a data source to mine historical stock data for six large companies. These companies and their stock symbols are as follows: Apple (APPL), Oracle (ORCL), Tesla (TSLA), International Business Machines (IBM), Yelp (YELP), and Microsoft (MSFT). The dataset has of twenty-one years of stock prices from January 1st, 2000 through September 8th, 2021. For each company, we looked at six features: high stock price, low stock price, prices of stock at open and close of market, the number of shares traded e.g. volume and the adjusted closing stock price.

Introduction

In this lab, a program written in Python was used to calculate the following statistics: mean, variance, standard deviation, median and correlation matrix of stock prices between Apple, Oracle, Tesla, IBM, Yelp, and Microsoft in a span of 21 years. The mean is the average value found in a sample and is calculated using a formula displayed in Figure 1 where you sum up all values in the sample and divide it by the total number of values (via StatisticsHowTo).

$$\bar{x} = (\Sigma x_i) / n$$

Figure 1: Sample Mean

The variance tells you how spread out the data is from the mean in the sample and can be calculated per Figure 2 by taking the square of standard deviation of a random variable (Wikipedia):

$$\frac{1}{n-1}\sum_{i=1}^{n}\left(Y_i - \overline{Y}\right)^2$$

Figure 2: Variance

The median reflects the middle value of an ordered list. If the data x's values is sorted in numerical order with n elements, then median can be calculated with the following formula in Figure 3 (Wikipedia):

$$\text{if } n \text{ is odd, } \operatorname{median}(x) = x_{(n+1)/2}$$
$$\text{if } n \text{ is even, } \operatorname{median}(x) = \frac{x_{(n/2)} + x_{(n/2)+1}}{2}$$

Figure 3: Median

Standard deviation is the square root of the variance and measures how a value far from mean value of the sample. The formula from Figure 4 is from Wikipedia:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}, \text{ where } \mu = \frac{1}{N}\sum_{i=1}^{N} x_i.$$

Figure 4: Standard Deviation

Correlation between two random variables X and Z are equal to the covariance between these two variables divided to the product of the standard deviations of these variables which can be described by the following expression. (Jafari, 2022) Correlation coefficients' values range between -1 and 1.

$$Cor(X, Y) = \frac{Cov(X, Y)}{\sigma_X \sigma_Y}$$

Figure 5: Correlation Matrix

Methodology

Dataset

Python's Pandas_datareader package was used to connect to Yahoo's stocks API to retrieve company stock prices for six large companies in the U.S with the following stock symbols. APPL, ORCL, TSLA, IBM, YELP, and MSFT. Per Yahoo Finance, these stock symbols represent the following company names:

Stock Symbol	Company
AAPL	Apple Inc
ORCL	Oracle Corporation
TSLA	Tesla, Inc.
IBM	International Business Machines Corporation
YELP	Yelp, Inc.
MSFT	Microsoft Corporation

The dataset contains 21 years of daily stock prices for these companies from January 1st, 2000 through September 8th, 2021. Stocks are traded Monday – Friday. The features evaluated in the dataset are High ($), Low($), Open ($), Close($), Volume, and Adjusted Close($). High($) is the highest stock price the company had for the day, Low($) represents the lowest price the stock it traded for the day, Close($) is the stock price at market closing, Volume measures the number of shares the company sold in the day, and Adjusted Close ($) is the company's stock price after taking any actions like paying dividends or splitting the stock.

All companies and features were read into a single Pandas dataframe. A column storing the symbol was added to the dataframe to differentiate data between each company. Individual dataframes were used to construct four tables comparing stock prices for Apple, Oracle, Tesla, IBM, Yelp, and Microsoft. The first table had a comparison of the mean value for each feature (High, Low, Open Close, Volume, and Adjusted Close) represented as columns in our dataset. The second table calculated the variance of the features. The third table presented the standard deviation values. The fourth table displayed the median values of the dataset. Additional analysis after the tables were constructed required we calculated the maximum and minimum value for each feature. Finally, for each table, we calculated the company symbols that had the minimum and maximum value for attribute.

Lab Answers

Question 1

Using the pandas_datareader package connect to yahoo database and load the stock value for the following giant companies.

```
#Stock symbols for lab
stocks = ['AAPL', 'ORCL', 'TSLA', 'IBM', 'YELP', 'MSFT']

#Start Date
start_dt = '2000-01-01'

#End Date
end_dt = '2021-09-18'

#Source to read in stocks data
source = 'yahoo'

#Create empty dataframe
df1 = pd.DataFrame()

for i in range(len(stocks)):
  df = web.DataReader(stocks[i], data_source = source, start= start_dt, end = end_dt)
  df['Symbol'] = stocks[i]
  df1 = df1.append(df)
```

Question 2: Mean Value Comparison with Max / Min Values

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	22.76	22.28	22.52	22.53	433707785.90	21.43
ORCL	31.46	30.73	31.08	31.09	30164879.39	28.04
TSLA	103.39	99.01	101.28	101.33	31540116.84	101.33
IBM	124.87	122.71	123.74	123.8	6337327.69	90.14
YELP	39.04	37.48	38.26	38.25	2203685.35	38.25
MSFT	56.87	55.72	56.30	56.31	53555522.11	49.82
Maximum Value	124.87	122.71	123.74	123.8	433707785.90	101.33
Minimum Value	22.76	22.28	22.52	22.53	2203685.35	21.43

Question 3: Variance Comparison with Max / Min Values

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	1000.25	956.51	978.62	979.05	1.52E+17	968.29
ORCL	275.34	268.57	271.32	272.01	4.166E+14	271.08
TSLA	34144.91	31299.21	32788.05	32809.42	8.137E+14	32809.42
IBM	1254.58	1247.03	1250.86	1251.23	1.201E+13	1011.16
YELP	255.72	235.52	246.72	244.85	7.045E+12	244.85
MSFT	3200.97	3085.34	3143.26	3147.9	9.745E+14	3350.04
Maximum Value	34144.91	31299.21	32788.05	32809.42	1.52E+17	32809.42
Minimum Value	255.72	235.52	246.72	244.85	7.045E+12	244.85

Question 4: Standard Deviation Value Comparison with Max / Min Values

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	31.63	30.93	31.28	31.29	389872786.00	31.12
ORCL	16.59	16.39	16.47	16.49	20411399.99	16.46
TSLA	184.78	176.92	181.07	181.13	28525233.60	181.13
IBM	35.42	35.31	35.37	35.37	3465185.02	31.8
YELP	15.99	15.35	15.71	15.65	2654194.68	15.65
MSFT	56.58	55.55	56.06	56.11	31217778.07	57.88
Maximum Value	184.78	176.92	181.07	181.13	389872786.00	181.13
Minimum Value	15.99	15.35	15.71	15.65	2654194.68	15.65

Question 5: Median Value Comparison with Max/Min Values

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	11.23	10.96	11.12	11.11	320241600.00	9.53
ORCL	30.4	29.43	30.02	29.94	26392900.00	25.73
TSLA	46.4	45.00	45.79	45.78	25061000.00	45.78
IBM	121.02	118.75	119.74	119.9	5476124.00	91.32
YELP	35.96	34.88	35.50	35.4	1545100.00	35.4
MSFT	30.94	30.30	30.62	30.65	48910600.00	22.56
Maximum Value	121.02	118.75	119.74	119.9	320241600.00	91.32
Minimum Value	11.23	10.96	11.12	11.11	1545100.00	9.53

Question 6: Companies with Maximum and Minimum Mean for Each Attribute

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	22.76	22.28	22.52	22.53	433707785.92	21.43
ORCL	31.46	30.73	31.08	31.09	30164879.39	28.04
TSLA	103.39	99.01	101.28	101.33	31540116.84	101.33
IBM	124.87	122.71	123.74	123.8	6337327.69	90.14
YELP	39.04	37.48	38.26	38.25	2203685.35	38.25
MSFT	56.87	55.72	56.3	56.31	53555522.11	49.82
Company ...	IBM	IBM	IBM	IBM	AAPL	TSLA
Company ...	AAPL	AAPL	AAPL	AAPL	YELP	AAPL

Question 7: Companies with Maximum / Minimum **Variance** for Each Attribute

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	1000.25	956.51	978.62	979.05	1.520007892245867e+17	968.29
ORCL	275.34	268.57	271.32	272.01	416625249367809.8	271.08
TSLA	34144.91	31299.21	32788.05	32809.42	813688952073349.9	32809.42
IBM	1254.58	1247.03	1250.86	1251.23	12007507220798.12	1011.16
YELP	255.72	235.52	246.72	244.85	7044749407828.64	244.85
MSFT	3200.97	3085.34	3143.26	3147.9	974549667662495.9	3350.04
Company ...	TSLA	TSLA	TSLA	TSLA	AAPL	TSLA
Company ...	YELP	YELP	YELP	YELP	YELP	YELP

Question 8: Companies with Maximum / Minimum **Standard Deviation** for Each Attribute

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	31.63	30.93	31.28	31.29	389872785.95	31.12
ORCL	16.59	16.39	16.47	16.49	20411399.99	16.46
TSLA	184.78	176.92	181.07	181.13	28525233.6	181.13
IBM	35.42	35.31	35.37	35.37	3465185.02	31.8
YELP	15.99	15.35	15.71	15.65	2654194.68	15.65
MSFT	56.58	55.55	56.06	56.11	31217778.07	57.88
Company ...	TSLA	TSLA	TSLA	TSLA	AAPL	TSLA
Company ...	YELP	YELP	YELP	YELP	YELP	YELP

Question 9: Companies with Maximum / Minimum **Median** for Each Attribute

Name	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
AAPL	11.23	10.96	11.12	11.11	320241600.00	9.53
ORCL	30.40	29.43	30.02	29.94	26392900.00	25.73
TSLA	46.40	45.00	45.79	45.78	25061000.00	45.78
IBM	121.02	118.75	119.74	119.90	5476124.00	91.32
YELP	35.96	34.88	35.50	35.40	1545100.00	35.40
MSFT	30.94	30.30	30.62	30.65	48910600.00	22.56
Company Max Value	IBM	IBM	IBM	IBM	AAPL	IBM
Company Min Value	AAPL	AAPL	AAPL	AAPL	YELP	AAPL

Question 10: Correlation Matrix for Apple Company

Correlation Matrix for Apple

	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
High	1.00	1.00	1.00	1.00	-0.41	1.00
Low	1.00	1.00	1.00	1.00	-0.42	1.00
Open	1.00	1.00	1.00	1.00	-0.41	1.00
Close	1.00	1.00	1.00	1.00	-0.41	1.00
Volume	-0.41	-0.42	-0.41	-0.41	1.00	-0.41
Adj Close	1.00	1.00	1.00	1.00	-0.41	1.00

Question 11: Correlation Matrices for Oracle, Tesla, IBM, Yelp and Microsoft

Correlation Matrix for Oracle

	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
High	1.00	1.00	1.00	1.00	-0.55	1.00
Low	1.00	1.00	1.00	1.00	-0.57	1.00
Open	1.00	1.00	1.00	1.00	-0.56	1.00
Close	1.00	1.00	1.00	1.00	-0.56	1.00
Volume	-0.55	-0.57	-0.56	-0.56	1.00	-0.56
Adj Close	1.00	1.00	1.00	1.00	-0.56	1.00

Correlation Matrix for Tesla

	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
High	1.00	1.00	1.00	1.00	0.17	1.00
Low	1.00	1.00	1.00	1.00	0.16	1.00
Open	1.00	1.00	1.00	1.00	0.16	1.00
Close	1.00	1.00	1.00	1.00	0.16	1.00
Volume	0.17	0.16	0.16	0.16	1.00	0.16
Adj Close	1.00	1.00	1.00	1.00	0.16	1.00

Correlation Matrix for IBM

	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
High	1.00	1.00	1.00	1.00	-0.39	0.93
Low	1.00	1.00	1.00	1.00	-0.41	0.93
Open	1.00	1.00	1.00	1.00	-0.40	0.93
Close	1.00	1.00	1.00	1.00	-0.40	0.93
Volume	-0.39	-0.41	-0.40	-0.40	1.00	-0.43
Adj Close	0.93	0.93	0.93	0.93	-0.43	1.00

Correlation Matrix for Yelp

	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
High	1.00	1.00	1.00	1.00	0.27	1.00
Low	1.00	1.00	1.00	1.00	0.24	1.00
Open	1.00	1.00	1.00	1.00	0.25	1.00
Close	1.00	1.00	1.00	1.00	0.25	1.00
Volume	0.27	0.24	0.25	0.25	1.00	0.25
Adj Close	1.00	1.00	1.00	1.00	0.25	1.00

Correlation Matrix for Microsoft

	High($)	Low($)	Open($)	Close($)	Volume	Adj Close($)
High	1.00	1.00	1.00	1.00	-0.39	1.00
Low	1.00	1.00	1.00	1.00	-0.39	1.00
Open	1.00	1.00	1.00	1.00	-0.39	1.00
Close	1.00	1.00	1.00	1.00	-0.39	1.00
Volume	-0.39	-0.39	-0.39	-0.39	1.00	-0.40
Adj Close	1.00	1.00	1.00	1.00	-0.40	1.00

Conclusion

We looked at daily stock prices using Yahoo's API for six well known US companies Apple, Oracle, Tesla, IBM, Yelp and Microsoft. All are tech companies except for Yelp. We looked at a twenty-one-year span and conducted exploratory data analysis by comparing the values of the mean, variance, standard deviation, median. For each company, we look at the following stock prices: High, Low, Open, Close, Volume and Adjusted Close.

For mean, IBM had the highest value in the categories of High, Low, Open and Close. But in terms of volume, Apple had the highest mean. Tesla had the highest mean value for adjusted closing stock price. Apple had the lowest mean values for all features except for Volume, that taken by Yelp.

Looking at the variance table, Tesla had the maximum value for all categories except for Volume- this went to Apple. This speaks to the volatility of the Tesla stock prices on the market. On the low side for variance was Yelp for all categories so it is a consistent performer in the market.

For the standard deviation comparison, Tesla again held the highest values in all categories except for Volume – this went to Apple. Yelp help the lowest standard deviation values in the list of companies. These findings are consistent with the variance comparison and makes sense because standard deviation is the square root of variance.

In the median comparison table, IBM was the best performer for all attributes except for Volume. Apple is the clear winner in this category which speaks to the popularity of company. IBM beating out Tesla in the Median and Mean comparison shows that IBM has a longer history of trading on the stock market. Apple has the lowest median stock prices for all attributes except for volume which goes to Yelp.

High, Low, Open, Close and Adjusted Close stock prices are highly correlated features. For Apple, Oracle, Tesla, Yelp, and Microsoft, the correlation matrix shows positive correlation between the features High, Low, Open, Close and Adjusted Close. This makes sense because all the features reflect a company's stock value but at different times of the day. The exception is for IBM, the Adjusted Close feature is less than perfect correlation between High, Low, Open and Close.

In the correlation matrix for the companies, the correlation between stock prices and volume for Apple is -0.41, Oracle is -0.55, Tesla is 0.17, IBM is – 0.39, Yelp is 0.27, and Microsoft is – 0.29. This means that for Apple, Oracle, IBM and Microsoft this means the more shares of stocks are traded, it lowered the company's stock value. Tesla and Yelp's stock prices did not drop with increase in volume.

Appendix A: Code

```
# DATS 6401: Lab 2 (Spring 22)
# Lydia Teinfalt
# 02/02/2022
import pandas_datareader as web
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

#Stock symbols for lab
stocks = ['AAPL', 'ORCL', 'TSLA', 'IBM', 'YELP', 'MSFT']

#Start Date
start_dt = '2000-01-01'

#End Date
end_dt = '2021-09-18'

#Source to read in stocks data
source = 'yahoo'

#Create empty dataframe
df1 = pd.DataFrame()

for i in range(len(stocks)):
    df = web.DataReader(stocks[i], data_source = source, start= start_dt, end
= end_dt)
    df['Symbol'] = stocks[i]
    df1 = df1.append(df)

columns = ['High','Low', 'Open', 'Close', 'Volume', 'Adj Close']
renamed_cols = {'High': 'High($)', 'Low': 'Low($)', 'Open': 'Open($)',
'Close': 'Close($)', 'Adj Close' : 'Adj Close($)'}
def max_min(dataframe, filename):
    max = []
    min= []

    for i in columns:
        max.append(dataframe[i].max())
        min.append(dataframe[i].min())

    max_series = pd.Series(max, index=dataframe.columns)
    min_series = pd.Series(min, index=dataframe.columns)
    dataframe = dataframe.append(max_series, ignore_index=True)
    dataframe = dataframe.append(min_series, ignore_index=True)
    dataframe.rename(columns=renamed_cols, inplace=True)
    dataframe = dataframe.round(decimals=2)
    pd.set_option('display.float_format', lambda x: '%.2f' % x)
    dataframe.insert(loc=0, column='Name', value=['AAPL', 'ORCL', 'TSLA',
'IBM', 'YELP', 'MSFT', 'Maximum Value', 'Minimum Value'])
```

```
    dataframe.to_csv(filename)
    print(dataframe)

def max_min_company(df, fn):
    df3 = df.copy()
    df3 = df3.round(decimals=2)
    pd.set_option('display.float_format', lambda x: '%.2f' % x)
    max_vals = df.idxmax(axis=0)
    min_vals = df.idxmin(axis=0)
    df3 = df3.append(max_vals, ignore_index=True)
    df3 = df3.append(min_vals, ignore_index=True)
    df3.insert(loc=0, column='Name',value=['AAPL', 'ORCL', 'TSLA', 'IBM',
'YELP', 'MSFT', 'Company Max Value', 'Company Min Value'])
    df3.rename(columns=renamed_cols, inplace=True)
    df3.to_csv(fn)
    print(df3)

print("*"*64)
# Create Mean Value Comparison Table
print("Mean Value Comparison")
df_mean = df1.groupby('Symbol').mean()
df_mean= pd.DataFrame(df_mean, index=['AAPL', 'ORCL', 'TSLA', 'IBM', 'YELP',
'MSFT'])



max_min(df_mean, "Stocks_Mean.csv")
max_min_company(df_mean, "Company_Mean.csv")

print("*"*64)

# Create Mean Value Comparison Table
print("Standard Deviation Value Comparison")
df_var = df1.groupby('Symbol').var()
df_var= pd.DataFrame(df_var, index=['AAPL', 'ORCL', 'TSLA', 'IBM', 'YELP',
'MSFT'])

max_min(df_var, "Stocks_Cov.csv")
max_min_company(df_var, "Company_variance.csv")
print("*"*64)

#Standard Deviation Value Comparison Table
print("Variance Value Comparison")
df_var = df1.groupby('Symbol').std()
df_var= pd.DataFrame(df_var, index=['AAPL', 'ORCL', 'TSLA', 'IBM', 'YELP',
'MSFT'])

max_min(df_var, "Stocks_STD.csv")
max_min_company(df_var, "Company_STD.csv")
print("*"*64)
#Standard Deviation Value Comparison Table
print("Variance Value Comparison")
df_med = df1.groupby('Symbol').median()
df_med= pd.DataFrame(df_med, index=['AAPL', 'ORCL', 'TSLA', 'IBM', 'YELP',
'MSFT'])

max_min(df_med, "Stocks_Median.csv")
```

```
max_min_company(df_med, "Company_Median.csv")
print("*"*64)

Stock_Co = {"AAPL": "Apple", "ORCL": "Oracle", "TSLA": "Tesla", "IBM": "IBM",
"YELP": "Yelp", "MSFT": "Microsoft"}

for j in stocks:
    df_corr = df1[df1['Symbol'] == j]
    print("Correlation Matrix for " + Stock_Co[j])
    #dc = df_corr.corr().round(decimals=2)
    dc = df_corr.corr()
    pd.set_option('display.float_format', lambda x: '%.2f' % x)
    dc.rename(columns=renamed_cols, inplace=True)
    dc.to_csv(j + ".csv")
    print(dc)
```

References

Sample Mean: Symbol (X Bar), Definition, Standard Error. (n.d.). Statistics How To. https://www.statisticshowto.com/probability-and-statistics/statistics-definitions/sample-mean/

Wikipedia Contributors. (2019, January 8). Variance. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Variance

Wikipedia Contributors. (2022). Median. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Median

Wikipedia Contributors. (2022). Standard_deviation. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Standard_deviation