Project A

Raj Kumar Anand
rajanand@mit.edu

September 16, 2020

Exercise 1

For the purpose of this exercise, excel file uploaded on Canvas is used.

(a) In order to find the range of data, we first imported the excel(csv) file in R

```
### Loading SPX data from file
spx_data <- read.csv("spx_data.csv", header = TRUE)
## Skipping the first row containing NA
spx_data <- spx_data[2:nrow(spx_data),]
## Removing extra columns
spx_data <- spx_data[,c("Date","PX_OPEN","PX_HIGH","PX_LOW","PX_LAST")]
## Converting date column
spx_data$Date <- as.Date(spx_data$Date, format = "%m/%d/%Y")
## Converting other columns to float
spx_data[names(spx_data) != "Date"] <- lapply(spx_data[names(spx_data) != "Date
    "], function(x) as.numeric(as.character(x)))

## Finding date range
spx_data$Date[nrow(spx_data)] - spx_data$Date[1]

## Finding summary of data
summary(spx_data)
```

The file contains data for 33845 days, ranging from 3-Jan-1928 to 1-Sep-2020.

On finding the summary of data we notice that the data contains 281 NA's rows. So data is not entirely complete.

```
> summary(spx_data)
      Date               PX_OPEN          PX_HIGH          PX_LOW           PX_LAST
 Min.   :1928-01-03   Min.   :   4.40   Min.   :   4.40   Min.   :   4.40   Min.   :   4.40
 1st Qu.:1950-09-20   1st Qu.:  23.82   1st Qu.:  23.82   1st Qu.:  23.82   1st Qu.:  23.82
 Median :1973-12-26   Median :  99.40   Median :  99.40   Median :  99.40   Median :  99.40
 Mean   :1974-01-22   Mean   : 481.21   Mean   : 483.85   Mean   : 478.38   Mean   : 481.32
 3rd Qu.:1997-04-08   3rd Qu.: 819.19   3rd Qu.: 830.66   3rd Qu.: 811.22   3rd Qu.: 820.16
 Max.   :2020-09-01   Max.   :3509.73   Max.   :3528.03   Max.   :3494.60   Max.   :3526.65
                      NA's   :281       NA's   :281       NA's   :281       NA's   :281
```

(b) Following data integrity checks are applied on data.

Data Integrity Checks	Result on given Data	Can be solved without Alt. Data ?
Is $PX_{HIGH} \geq PX_{LOW}$?	Pass	No
Are $Prices > 0$?	Pass	No
Are $Prices = 0$?	Pass	No
Duplicated data?	Pass	Yes
If HIGH is highest?	Pass	No
If LOW is lowest?	Pass	No
Missing data?	Fail	Yes

```
#Checking if PX_HIGH is greater than PX_LOW
sum(which(spx_data$PX_HIGH < spx_data$PX_LOW)) # Pass

#Checking for negative prices
apply(spx_data[,names(spx_data) != "Date"], 2, function(x) sum(which(x < 0))) ##
    Pass


#Checking for zero prices
apply(spx_data[,names(spx_data) != "Date"], 2, function(x) sum(which(x == 0))) ##
    Pass

#Checking for duplications
print(paste0("Duplicated rows in spx data: ", dim(spx_data[duplicated(spx_data),])
    [1])) ## Pass

#Checking if high is the highest value and low is the lowest value
sum(spx_data$PX_HIGH != apply(spx_data[,c("PX_OPEN", "PX_LAST", "PX_LOW", "PX_HIGH
    ")],1, max),na.rm = TRUE) ## Pass
sum(spx_data$PX_LOW != apply(spx_data[,c("PX_OPEN", "PX_LAST", "PX_LOW", "PX_HIGH"
    )],1, min),na.rm = TRUE) ## Pass
```

(c) In order to calculate probabilities, we have assumed that our sample size (given in part (a)), which is equal to 33845 days (with 281 NA's) is sufficiently large. Therefore, probabilities are calculated empirically using the following formulas.

$$Prob_{event} = \frac{\text{Days when event happened}}{\text{Total number of days}}$$

Probabilities are summarised in table below.

Description	Probability
Market HIGH occurs at OPEN	63.14 %
Market HIGH occurs at CLOSE	61.06 %
Market LOW occurs at OPEN	64.42%
Market LOW occurs at CLOSE	59.22%

```
## Probability that market daily high occurs at open
sum(spx_data$PX_HIGH == spx_data$PX_OPEN, na.rm = TRUE)/sum(!is.na(spx_data$PX_
    HIGH))

## Probability that market daily high occurs at close
sum(spx_data$PX_HIGH == spx_data$PX_LAST, na.rm = TRUE)/sum(!is.na(spx_data$PX_
    HIGH))

## Probability that market daily low occurs at open
sum(spx_data$PX_LOW == spx_data$PX_OPEN, na.rm = TRUE)/sum(!is.na(spx_data$PX_LOW)
    )

## Probability that market daily low occurs at close
sum(spx_data$PX_LOW == spx_data$PX_LAST, na.rm = TRUE)/sum(!is.na(spx_data$PX_LOW)
    )
```

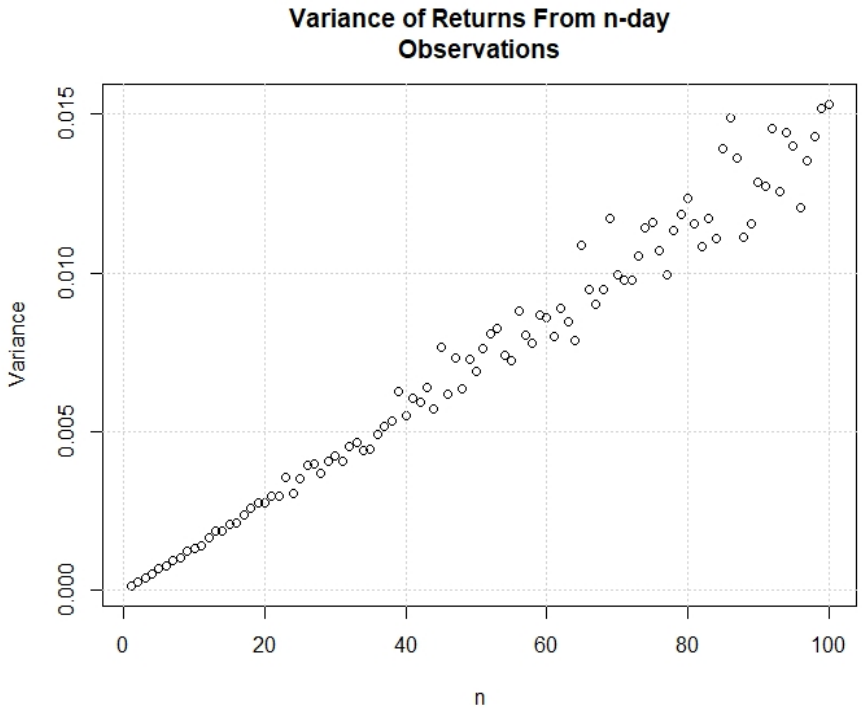
One of the methods for testing random walk is Variance Ratio test. Random Walk model predicts that variance has simple linear scaling with sampling interval. So, variance ratio test

states the following to check random walk, positive serial correlation and negative serial correlation.

$$\frac{Var(r_t^{(q)}}{qVar(r_t)} = 1 + \rho_1 \begin{cases} = 1 & \text{Random Walk} \\ > 1 & \text{positive serial correlation} \\ < 1 & \text{negative serial correlation} \end{cases}$$

The assumption that returns (from any of the prices) are random walk can easily be checked using the Variance Ratio Test.

```
## Variance Ratio Test
Variance <- var(diff(log(spx_data$PX_HIGH[!is.na(spx_data$PX_HIGH)])))
for (n in 2:100) {
    Variance[n] <- var(diff(log(spx_data$PX_HIGH[!is.na(spx_data$PX_HIGH)][seq(from
        =n, to=length(spx_data$PX_HIGH[!is.na(spx_data$PX_HIGH)]), by=n)]))
}
plot(Variance,xlab="n",main="Variance of Returns From n-day
```



(d) Intraday range is defined as given in formula.

$$H_t = \frac{p_{high,t} - p_{low,t}}{p_{low,t}}$$

Question asks for statistics from a period 1-Jan-1980 to 30-Aug-2011. So, we first need to subset the data from these dates. 1-Jan-1980 is not a Date in the data so we need to find the closest date from which we subset the data. This is done as shown below.

```
### Intraday range
closest_index_to_start_date <- which(abs(spx_data$Date - as.Date("1980-01-01"))
    == min(abs(spx_data$Date - as.Date("1980-01-01"))))
```

```
closest_index_to_end_date <- which(abs(spx_data$Date - as.Date("2011-08-30"))
    == min(abs(spx_data$Date - as.Date("2011-08-30"))))
intra_data <- spx_data[closest_index_to_start_date:closest_index_to_end_date,]

intraday_range <- data.frame("Date" = intra_data$Date, "Intraday_range" = (
    intra_data$PX_HIGH - intra_data$PX_LOW)/intra_data$PX_LOW)
top20_intraday <- intraday_range[order(intraday_range$Intraday_range,
    decreasing = TRUE)[1:20],]

## Finding number of dates in final 3 years
length(which(top20_intraday$Date >= as.Date("2008-09-01") & top20_intraday$Date
     <= as.Date("2011-08-30")))
```

Out of 20 dates mentioned in the table below, 15 dates belong to the last(final) 3 years.

	Date	Intraday_range
1975	1987-10-19	0.25739448
1976	1987-10-20	0.13471311
7290	2008-11-13	0.11520844
7266	2008-10-10	0.11497976
7278	2008-10-28	0.11267406
7265	2008-10-09	0.10565448
7267	2008-10-13	0.10318269
7269	2008-10-15	0.10023341
7295	2008-11-20	0.09727460
7660	2010-05-06	0.09550662
1977	1987-10-21	0.09470929
7270	2008-10-16	0.09456822
7257	2008-09-29	0.09280633
1980	1987-10-26	0.09222916
7301	2008-12-01	0.08939671
7262	2008-10-06	0.08888161
5700	2002-07-24	0.08849010
7274	2008-10-22	0.08661696
7296	2008-11-21	0.08121238
7297	2008-11-24	0.08037943

(e) Overnight return is calculated as given below.

$$R_{Overnight,t} = \frac{p_{open,t} - p_{close,t-1}}{p_{close,t-1}}$$

Question asks for statistics from period 1-Jan-1980 to 30-Aug-2011, which is same period as the previous question. So, we calculate the overnight return for over the same period using the code as shown below.

```
###   Overnight return
overnight <- data.frame("Date" = intra_data$Date, "overnight_return" = (intra_
    data$PX_OPEN - lag.xts(intra_data$PX_LAST))/lag.xts(intra_data$PX_LAST))
top20_overnight <- overnight[order(overnight$overnight_return, decreasing =
    TRUE)[1:20],]

bottom20_overnight <- overnight[order(overnight$overnight_return, decreasing =
    FALSE)[1:20],]
```

Using the overnight return calculated, we made two separate list, one for top 20 overnight return and one for lowest 20 overnight return. These lists are shown as below.

Date	overnight_return
1980-04-22	0.03637275
1982-01-28	0.03282960
1980-11-12	0.02536950
1980-03-28	0.02504582
1981-03-12	0.02493267
1981-09-28	0.02447459
1981-10-30	0.02376953
1980-01-08	0.02003558
1982-03-22	0.01952807
1981-10-02	0.01947386
1980-09-03	0.01931625
1980-06-04	0.01900281
1981-11-02	0.01895151
1980-04-09	0.01887352
1980-10-06	0.01855718
1980-07-14	0.01841480
1981-03-25	0.01811836
1980-03-18	0.01799335
1982-02-24	0.01757690
1980-12-17	0.01753446

Top 20 Overnight Returns

Date	overnight_return
1980-03-17	-0.03006734
1980-03-24	-0.02961587
1981-08-24	-0.02886327
1980-12-08	-0.02551668
1980-12-01	-0.02355537
1982-01-11	-0.02317022
1982-02-08	-0.02242879
1980-03-06	-0.02231621
1980-09-29	-0.02223981
1981-01-07	-0.02200985
1982-01-05	-0.02191625
1982-02-01	-0.02176080
1981-02-02	-0.02037823
1981-01-20	-0.02024261
1980-01-02	-0.02019641
1981-09-25	-0.01947657
1980-04-07	-0.01918747
1986-09-25	-0.01883359
1980-08-18	-0.01853325
1980-11-06	-0.01842686

Bottom 20 Overnight Returns

All the top 20 overnight returns belong to a three year period from 1-Jan-1980 to 31-Dec-1982. And 19 of the bottom 20 overnight returns belong to the same three year period of 1-Jan-1980 to 31-Dec-1982.

(f) One day jump is calculated using the following expression.

$$j_t = \frac{r_t}{\sigma_t *}$$

Where r_t is the log return given as $r_t = log(\frac{P_t}{P_{t-1}})$ and σ_t is standard deviation of returns using 63 trading days (prior to the start of day t).

In order to calculate log return for rolling volatility (standard deviation), first of all we need to take care of the missing rows of data. In order to deal with missing data we can either forward-fill it or back-fill it. Using filling function to remove blank rows will generate duplicate price data which will reduce the volatility. So in this exercise we will omit the missing rows.

Standard deviation can be calculated using the rollapply function in R. One important thing to note is to exclude the current row while calculating the rolling standard deviation. This is done by aligning it to the right index and specifying the width as a list of indices to be included in calculation. The code implementation is shown below.

Since the question does not specify any time period, all the historical data is being used for the analysis.

```
spx_data_clean <- na.omit(spx_data)
```

```
spx_data_clean$logReturn <- log(spx_data_clean$PX_LAST/lag.xts(spx_data_
    clean$PX_LAST))
spx_data_clean$RollingVol <- rollapply(spx_data_clean$logReturn, width =
    list(c(-63:-1)), function(x) sqrt(var(x)), fill= NA, align = "right")
spx_data_clean$impact <- spx_data_clean$logReturn/spx_data_clean$RollingVol
spx_data_clean <- na.omit(spx_data_clean)

top20_impact <- spx_data_clean[order(spx_data_clean$impact, decreasing =
    TRUE)[1:20],]

length(which(top20_impact$Date >= as.Date("2008-09-01") & top20_impact$Date
     <= as.Date("2011-08-30")))
```

Data for top 20 impact (absolute value) is given in the table below.

Date	PX_OPEN	PX_HIGH	PX_LOW	PX_LAST	logReturn	RollingVol	impact
1987-10-19	282.70	282.70	224.83	224.84	-0.22899723	0.012938518	17.698876
1962-05-28	55.50	55.50	55.50	55.50	-0.06908896	0.007435803	9.291393
1989-10-13	355.41	355.53	332.81	333.62	-0.06321316	0.006803581	9.291160
1955-09-26	42.61	42.61	42.61	42.61	-0.06847643	0.007684895	8.910523
1948-05-17	16.61	16.61	16.61	16.61	0.07628698	0.008589609	8.881310
1950-06-26	18.11	18.11	18.11	18.11	-0.05531611	0.006299885	8.780495
1946-09-03	15.53	15.53	15.53	15.53	-0.06963658	0.008723341	7.982788
1940-05-13	11.11	11.11	11.11	11.11	-0.05685834	0.007131548	7.972790
2018-02-05	2741.06	2763.39	2638.17	2648.94	-0.04184256	0.005313552	7.874688
2018-10-10	2873.90	2874.02	2784.86	2785.68	-0.03341633	0.004258461	7.847045
1940-05-14	10.28	10.28	10.28	10.28	-0.07764534	0.010035045	7.737419
2007-02-27	1449.25	1449.25	1389.42	1399.04	-0.03534959	0.004635348	7.626093
1939-09-05	12.64	12.64	12.64	12.64	0.11206366	0.014829834	7.556636
1929-10-28	22.74	22.74	22.74	22.74	-0.13857582	0.019275478	7.189228
1997-10-27	941.64	941.64	876.73	876.99	-0.07112745	0.010482923	6.785078
1933-03-15	6.81	6.81	6.81	6.81	0.15366132	0.022943985	6.697238
1955-07-06	43.18	43.18	43.18	43.18	0.03511613	0.005403470	6.498812
1953-09-18	23.95	23.95	23.95	23.95	0.03743525	0.005768049	6.490107
1963-11-26	72.38	72.38	72.38	72.38	0.03902178	0.006083434	6.414433
2011-08-08	1198.48	1198.48	1119.28	1119.46	-0.06895833	0.011246034	6.131791

As evident from the table, there is only 1 data points belonging to 3 year period ending in 2011.

Exercise 2

For this exercise, we are comparing data across three major data providers i.e Bloomberg, Yahoo finance and Wall Street Journal (WSJ). SPX 500 OHLC levels for 6-Oct-1982 are summarised in the table below.

Price Type	OPEN	HIGH	LOW	CLOSE
Bloomberg	122.00	126.97	122.00	126.97
Yahoo Finance	122.00	125.97	122.00	125.97
WSJ	125.97	126.30	121.82	125.97

On plotting different prices from different sources we get the following observations.


PX_HIGH
150
140
130
120
110
1-Sep-82
8-Sep-82
15-Sep-82
22-Sep-82
29-Sep-82
6-Oct-82
13-Oct-82
20-Oct-82
27-Oct-82
3-Nov-82
10-Nov-82
Yahoo
Bloomberg
WSJ


PX_LOW
150
140
130
120
110
1-Sep-82
8-Sep-82
15-Sep-82
22-Sep-82
29-Sep-82
6-Oct-82
13-Oct-82
20-Oct-82
27-Oct-82
3-Nov-82
10-Nov-82
Yahoo
Bloomberg
WSJ

From above plots, we can make following inferences.

- Plots for Yahoo and Bloomberg are almost overlapping which means their data are very consistent with each other.
- Data from WSJ has a lot of noise in it. On observing the plot for HIGH, we see sharp jumps in WSJ data. While it may happen that data from WSJ is correct while other two sources are not correct, but statistically (1 vs 2), WSJ data seems to have a lot of noise.
- Surprisingly, data for CLOSE is consistent across all the three sources with an exception on 6-Oct-1982 when Yahoo and Bloomberg shows some inconsistency. Consistency in

Close price can be attributed to the fact that most daily strategies/financial operations like End-of-Day hedging use Closing price so all vendors pay a lot of attention in having correct Closing price.

An incorrect price data can have a huge economic impact.

- With the rise of passive investment funds and ETFs, that usually have huge exposure to broad market index, a slight change in price can impact the performance of these funds.
- The index market is directly related to index futures market. An incorrect price in Index market can create arbitrage opportunities
- Additionally, S&P 500 indices are used as an underlying for option on these indices without huge notional tied to different levels of index. An incorrect price can potentially trigger exercise on these options.
- An incorrect data can also pose a trust crisis in overall financial institutions.

To resolve these discrepancies, following steps can be followed.

- Financial institutions can use multiple data provider which can be used to verify different data feeds. In the above example, seeing the consistency between Yahoo and Bloomberg, it can be safely concluded that data feeds from WSJ has a lot of problems.
- In case the discrepancies cannot be resolved by multiple data sources, these instances can be flagged and considered manually using business judgement.

Exercise 3

The constituents of an index can be found in Bloomberg by using "MEMB" function. Using the function, we first found the constituents of index on 24-Aug-2020 and 31-Aug-2020 to find the replacements in the Index announced on 24-Aug-2020.

Composition of Index on		24-Aug-20		
Ticker	**Name**	**Weight**	**Shares**	**Price**
AAPL UW Equity	Apple Inc	12.197503	1	503.43
AXP UN Equity	American Express Co	2.406646	1	99.33
BA UN Equity	Boeing Co/The	4.319268	1	178.27
CAT UN Equity	Caterpillar Inc	3.432736	1	141.68
CSCO UW Equity	Cisco Systems Inc	1.021971	1	42.18
CVX UN Equity	Chevron Corp	2.112751	1	87.2
DIS UN Equity	Walt Disney Co/The	3.166462	1	130.69
DOW UN Equity	Dow Inc	1.117432	1	46.12
GS UN Equity	Goldman Sachs Group Inc/The	5.023599	1	207.34
HD UN Equity	Home Depot Inc/The	6.947608	1	286.75
IBM UN Equity	International Business Machines Corp	3.045075	1	125.68
INTC UW Equity	Intel Corp	1.190603	1	49.14
JNJ UN Equity	Johnson & Johnson	3.686412	1	152.15
JPM UN Equity	JPMorgan Chase & Co	2.424333	1	100.06
KO UN Equity	Coca-Cola Co/The	1.162255	1	47.97
MCD UN Equity	McDonald's Corp	5.151527	1	212.62
MMM UN Equity	3M Co	3.962135	1	163.53
MRK UN Equity	Merck & Co Inc	2.069624	1	85.42
MSFT UW Equity	Microsoft Corp	5.177452	1	213.69
NKE UN Equity	NIKE Inc	2.709506	1	111.83
PFE UN Equity	Pfizer Inc	0.941046	1	38.84
PG UN Equity	Procter & Gamble Co/The	3.355931	1	138.51
RTX UN Equity	Raytheon Technologies Corp	1.499278	1	61.88
TRV UN Equity	Travelers Cos Inc/The	2.783889	1	114.9
UNH UN Equity	UnitedHealth Group Inc	7.482822	1	308.84
V UN Equity	Visa Inc	5.001066	1	206.41
VZ UN Equity	Verizon Communications Inc	1.443309	1	59.57
WBA UW Equity	Walgreens Boots Alliance Inc	0.962852	1	39.74
WMT UN Equity	Walmart Inc	3.181968	1	131.33
XOM UN Equity	Exxon Mobil Corp	1.02294	1	42.22

Composition on 31-Aug-2020				
Ticker	Name	Weight	Shares	Price
AAPL UW Equity	Apple Inc	2.986346	1	129.04
AMGN UW Equity	Amgen Inc	5.862532	1	253.32
AXP UN Equity	American Express Co	2.351076	1	101.59
BA UN Equity	Boeing Co/The	3.976394	1	171.82
CAT UN Equity	Caterpillar Inc	3.293451	1	142.31
CRM UN Equity	salesforce.com Inc	6.309882	1	272.65
CSCO UW Equity	Cisco Systems Inc	0.977089	1	42.22
CVX UN Equity	Chevron Corp	1.942374	1	83.93
DIS UN Equity	Walt Disney Co/The	3.05184	1	131.87
DOW UN Equity	Dow Inc	1.044203	1	45.12
GS UN Equity	Goldman Sachs Group Inc/The	4.741264	1	204.87
HD UN Equity	Home Depot Inc/The	6.596621	1	285.04
HON UN Equity	Honeywell International Inc	3.831289	1	165.55
IBM UN Equity	International Business Machines Corp	2.853738	1	123.31
INTC UW Equity	Intel Corp	1.179125	1	50.95
JNJ UN Equity	Johnson & Johnson	3.550336	1	153.41
JPM UN Equity	JPMorgan Chase & Co	2.318676	1	100.19
KO UN Equity	Coca-Cola Co/The	1.146262	1	49.53
MCD UN Equity	McDonald's Corp	4.941449	1	213.52
MMM UN Equity	3M Co	3.772738	1	163.02
MRK UN Equity	Merck & Co Inc	1.973386	1	85.27
MSFT UW Equity	Microsoft Corp	5.219394	1	225.53
NKE UN Equity	NIKE Inc	2.589447	1	111.89
PG UN Equity	Procter & Gamble Co/The	3.201342	1	138.33
TRV UN Equity	Travelers Cos Inc/The	2.685489	1	116.04
UNH UN Equity	UnitedHealth Group Inc	7.233279	1	312.55
V UN Equity	Visa Inc	4.90604	1	211.99
VZ UN Equity	Verizon Communications Inc	1.371673	1	59.27
WBA UW Equity	Walgreens Boots Alliance Inc	0.879889	1	38.02
WMT UN Equity	Walmart Inc	3.213377	1	138.85

Replacement done in INDU Index on 24-Aug-2020 are given in table below.

Previous Index Members	New Index Members
Pfizer Inc	Amergen Inc
Raytheon Technologies Corp	salesforce.com Inc
Exxon Mobil Corp	Honeywell International Inc

(a) Formula for calculating Index is given as follow.

$$I = \frac{\sum q_i p_i}{D} = \frac{\sum q_i' p_i'}{D'}$$

Where I is Index Level and q_i and p_i are quantities and prices before announcements and q_i' and p_i' are quantities and prices after announcement.
Divisor is changed so that Index level remains the same. Dow Jones index level at market close on 24-Aug-2020 is 28308.46. Replacing the three stocks with new members of index and taking their market closing price we get a table like below.

Present Earlier	Name	Price
AAPL UW Equity	Apple Inc	503.43
AXP UN Equity	American Express Co	99.33
BA UN Equity	Boeing Co/The	178.27
CAT UN Equity	Caterpillar Inc	141.68
CSCO UW Equity	Cisco Systems Inc	42.18
CVX UN Equity	Chevron Corp	87.2
DIS UN Equity	Walt Disney Co/The	130.69
DOW UN Equity	Dow Inc	46.12
GS UN Equity	Goldman Sachs Group Inc/The	207.34
HD UN Equity	Home Depot Inc/The	286.75
IBM UN Equity	International Business Machines Corp	125.68
INTC UW Equity	Intel Corp	49.14
JNJ UN Equity	Johnson & Johnson	152.15
JPM UN Equity	JPMorgan Chase & Co	100.06
KO UN Equity	Coca-Cola Co/The	47.97
MCD UN Equity	McDonald's Corp	212.62
MMM UN Equity	3M Co	163.53
MRK UN Equity	Merck & Co Inc	85.42
MSFT UW Equity	Microsoft Corp	213.69
NKE UN Equity	NIKE Inc	111.83
AMGN UW Equity	Amgen Inc	235.57
PG UN Equity	Procter & Gamble Co/The	138.51
CRM UN Equity	salesforce.com Inc	208.46
TRV UN Equity	Travelers Cos Inc/The	114.9
UNH UN Equity	UnitedHealth Group Inc	308.84
V UN Equity	Visa Inc	206.41
VZ UN Equity	Verizon Communications Inc	59.57
WBA UW Equity	Walgreens Boots Alliance Inc	39.74
WMT UN Equity	Walmart Inc	131.33
HON UN Equity	Honeywell International Inc	159.37

Using formula to find value of divisor.

$$I = \frac{\sum q_i' p_i'}{D'}$$
$$28308.46 = \frac{4587.78}{D'}$$
$$D' = 0.16206392$$

(b)

Departing Index Members	Weighting
Pfizer Inc	0.941046 %
Raytheon Technologies Corp	1.499278 %
Exxon Mobil Corp	1.02294 %
Sum	**3.463264** %

(c) After replacement, the remaining 27 companies have lesser total weight within the index. The change in the weight is shown in image below. Previous weights corresponds to weights on 24-Aug-2020 while new weights corresponds to weights on 31-Aug-2020.

27 Members common members			
Ticker	**Name**	**Previous Weight (%)**	**New Weight (%)**
AAPL UW Equity	Apple Inc	12.197503	2.986346
AXP UN Equity	American Express Co	2.406646	2.351076
BA UN Equity	Boeing Co/The	4.319268	3.976394
CAT UN Equity	Caterpillar Inc	3.432736	3.293451
CSCO UW Equity	Cisco Systems Inc	1.021971	0.977089
CVX UN Equity	Chevron Corp	2.112751	1.942374
DIS UN Equity	Walt Disney Co/The	3.166462	3.05184
DOW UN Equity	Dow Inc	1.117432	1.044203
GS UN Equity	Goldman Sachs Group Inc/The	5.023599	4.741264
HD UN Equity	Home Depot Inc/The	6.947608	6.596621
IBM UN Equity	International Business Machines Corp	3.045075	2.853738
INTC UW Equity	Intel Corp	1.190603	1.179125
JNJ UN Equity	Johnson & Johnson	3.686412	3.550336
JPM UN Equity	JPMorgan Chase & Co	2.424333	2.318676
KO UN Equity	Coca-Cola Co/The	1.162255	1.146262
MCD UN Equity	McDonald's Corp	5.151527	4.941449
MMM UN Equity	3M Co	3.962135	3.772738
MRK UN Equity	Merck & Co Inc	2.069624	1.973386
MSFT UW Equity	Microsoft Corp	5.177452	5.219394
NKE UN Equity	NIKE Inc	2.709506	2.589447
PG UN Equity	Procter & Gamble Co/The	3.355931	3.201342
TRV UN Equity	Travelers Cos Inc/The	2.783889	2.685489
UNH UN Equity	UnitedHealth Group Inc	7.482822	7.233279
V UN Equity	Visa Inc	5.001066	4.90604
VZ UN Equity	Verizon Communications Inc	1.443309	1.371673
WBA UW Equity	Walgreens Boots Alliance Inc	0.962852	0.879889
WMT UN Equity	Walmart Inc	3.181968	3.213377
		96.536735	**83.996298**

The sum of weight for remaining 27 companies decreased from 96.5367% to 83.9962 %. This was done because the incoming stocks had a greater weight as shown in below image.

Before Announcment		After Announcement	
Departing Member	**Weight**	**Incoming Member**	**Weight**
Pfizer Inc	0.941046	Amgen Inc	5.862532
Raytheon Technologies Corp	1.499278	salesforce.com Inc	6.309882
Exxon Mobil Corp	1.02294	Honeywell International Inc	3.831289
	3.463264		**16.003703**

(d) In case, one of the replacement member is AMZN (Amazon Inc.) instead of AMGN (Amgen Inc.), then we have the following composition on 24-Aug-2020.

With Amazon		
Present Earlier		**Price**
AAPL UW Equity	Apple Inc	503.43
AXP UN Equity	American Express Co	99.33
BA UN Equity	Boeing Co/The	178.27
CAT UN Equity	Caterpillar Inc	141.68
CSCO UW Equity	Cisco Systems Inc	42.18
CVX UN Equity	Chevron Corp	87.2
DIS UN Equity	Walt Disney Co/The	130.69
DOW UN Equity	Dow Inc	46.12
GS UN Equity	Goldman Sachs Group Inc/The	207.34
HD UN Equity	Home Depot Inc/The	286.75
IBM UN Equity	International Business Machines Corp	125.68
INTC UW Equity	Intel Corp	49.14
JNJ UN Equity	Johnson & Johnson	152.15
JPM UN Equity	JPMorgan Chase & Co	100.06
KO UN Equity	Coca-Cola Co/The	47.97
MCD UN Equity	McDonald's Corp	212.62
MMM UN Equity	3M Co	163.53
MRK UN Equity	Merck & Co Inc	85.42
MSFT UW Equity	Microsoft Corp	213.69
NKE UN Equity	NIKE Inc	111.83
AMZN US Equity	Amazon Inc	3307.5
PG UN Equity	Procter & Gamble Co/The	138.51
CRM UN Equity	salesforce.com Inc	208.46
TRV UN Equity	Travelers Cos Inc/The	114.9
UNH UN Equity	UnitedHealth Group Inc	308.84
V UN Equity	Visa Inc	206.41
VZ UN Equity	Verizon Communications Inc	59.57
WBA UW Equity	Walgreens Boots Alliance Inc	39.74
WMT UN Equity	Walmart Inc	131.33
HON UN Equity	Honeywell International Inc	159.37

To get new divisor, we use the following formula.

$$I = \frac{\sum q_i' p_i'}{D'}$$
$$28308.46 = \frac{7659.67}{D'}$$
$$D' = 0.27058$$

Similarly for replacing it Berkshire Class A share, we have the following composition on 24-Aug-2020.

With Bershire Class A		
Present Earlier		Price
AAPL UW Equity	AAPL UW Equity	503.43
AXP UN Equity	AXP UN Equity	99.33
BA UN Equity	BA UN Equity	178.27
CAT UN Equity	CAT UN Equity	141.68
CSCO UW Equity	CSCO UW Equity	42.18
CVX UN Equity	CVX UN Equity	87.2
DIS UN Equity	DIS UN Equity	130.69
DOW UN Equity	DOW UN Equity	46.12
GS UN Equity	GS UN Equity	207.34
HD UN Equity	HD UN Equity	286.75
IBM UN Equity	IBM UN Equity	125.68
INTC UW Equity	INTC UW Equity	49.14
JNJ UN Equity	JNJ UN Equity	152.15
JPM UN Equity	JPM UN Equity	100.06
KO UN Equity	KO UN Equity	47.97
MCD UN Equity	MCD UN Equity	212.62
MMM UN Equity	MMM UN Equity	163.53
MRK UN Equity	MRK UN Equity	85.42
MSFT UW Equity	MSFT UW Equity	213.69
NKE UN Equity	NKE UN Equity	111.83
BRK/A US Equity	Bershire Class A	318800
PG UN Equity	Procter & Gamble Co/The	138.51
CRM UN Equity	salesforce.com Inc	208.46
TRV UN Equity	Travelers Cos Inc/The	114.9
UNH UN Equity	UnitedHealth Group Inc	308.84
V UN Equity	Visa Inc	206.41
VZ UN Equity	Verizon Communications Inc	59.57
WBA UW Equity	Walgreens Boots Alliance Inc	39.74
WMT UN Equity	Walmart Inc	131.33
HON UN Equity	Honeywell International Inc	159.37

To get new divisor in this case, we use the following formula.

$$I = \frac{\sum q_i' p_i'}{D'}$$
$$28308.46 = \frac{3231522100}{D'}$$
$$D' = 11.41539349$$

As evident from the two divisors above, since the price of Bershire is very large compared to Amazon, the divisor value increases to a large value so that Index value on close of the day remains same.

(e) Ideally, stock-splits won't have any impact on the index level. By the definition of an index, it's level needs to be the same before and after the stock splits. So, just before the stock-splits is effective, the divisor will change to adjust for the change in the price.

Any change in level of index