Part 1) Probability:

P(A1) = Probability of people in age group 18-34 = 4250/10000 = 0.425
P(A2) = Probability of people in age group 35-49 = 2850/10000=0.2850
P(A3) = Probability of people in age group 50-64= 1640/10000 =0.1640
P(A4) = Probability of people in age group >=65 = 1260/10000=0.1260

P(B/A1) = Probability of people who had BMI above 30 in age group 18-34 =1062/4250=0.24988
P(B/A2) = Probability of people who had BMI above 30 in age group 35-49 =1710/2850=0.6000
P(B/A3) = Probability of people who had BMI above 30 in age group 50-64 =656/1640=0.4000
P(B/A4) = Probability of people who had BMI above 30 in age group >=65 =189/1260=0.1500

....

a) Probability of a randomly selected person will have a BMI of above 30

P(B) = P(B|A1) *P(A1) + P(B|A2) *P(A2) + P(B|A3) *P(A3) + P(B/A4) *P(A4)

=0.24988*0.425+0.6000*0.2850+0.4000*0.1640+0.1500*0.1260

=0.3617

#Alternatively:

Total number of people who had BMI above 30 = 1062+1710+656+189 = 3617
P(B) = 3617/10000 = 0.3617

…………………………………………………………………………………………………..

b) Probability of A randomly selected person will have a BMI above 30 and, in the age, group 18-34 = P(A1/B)

= P(A1). P(B/A1)/(P(A1). P(B/A1) + P(A2). P(B/A2) + P(A3). P(B/A3) + P(A4). P(B/A4))

=0.2936

……………………………………………………………………………………………………..

c) Probability of A randomly selected person will have a BMI above 30 and, in the age, group 35-49= P(A2/B)

= P(A2). P(B/A2)/(P(A1). P(B/A1) + P(A2). P(B/A2) + P(A3). P(B/A3) + P(A4). P(B/A4))

=0.4727

……………………………………………………………………………………………………..

d) Probability of A randomly selected person will have a BMI above 30 and, in the age, group 50-64= P(A3/B)

= P(A3). P(B/A3)/(P(A1). P(B/A1) + P(A2). P(B/A2) + P(A3). P(B/A3) + P(A4). P(B/A4))

=0.1813

………

e) Probability of A randomly selected person will have a BMI above 30 and, in the age, group >=65 = P(A4/B)

= P(A4). P(B/A4)/(P(A1). P(B/A1) + P(A2). P(B/A2) + P(A3). P(B/A3) + P(A4). P(B/A4))

=0.0522

………

 # using function in R

Code:

```
bayes <- function(prior,liklihood){

  numerators <- prior*liklihood

  return(numerators/sum(numerators))

}

prior<- c(4250/10000,2850/10000,1640/10000,1260/10000)

prior

likelihood<-c(1062/4250,1710/2850,656/1640,189/1260)

likelihood

bayes(prior,likelihood)
```

console:

```
> bayes <- function(prior,liklihood){

+   numerators <- prior*liklihood

+   return(numerators/sum(numerators))
```

```
+ }

> prior<- c(4250/10000,2850/10000,1640/10000,1260/10000)

> prior

[1] 0.425 0.285 0.164 0.126

> likelihood<-c(1062/4250,1710/2850,656/1640,189/1260)

> likelihood

[1] 0.2498824 0.6000000 0.4000000 0.1500000

> bayes(prior,likelihood)

[1] 0.29361349 0.47276749 0.18136577 0.05225325
```

Part 2 Random variables:

```
# Part 2 Random Variables
```

#a) script:

```
s <- rolldie(3,makespace = TRUE)

#checking first 3 samples ouput

head(s,n=3)

#sum of the rollls is greater than 3 but less than 8

a <- subset(s,X1+X2+X3 >3 & X1+X2+X3 < 8)

#first theree sample output

head(s,n=3)

prob(a)
```

console answer(a):

```
# Part 2 Random Variables

> #a)
```

```
> s <- rolldie(3,makespace = TRUE)

> #checking first 3 samples ouput

> head(s,n=3)

  X1 X2 X3      probs

1  1  1  1 0.00462963

2  2  1  1 0.00462963

3  3  1  1 0.00462963

> #sum of the rollls is greater than 3 but less than 8

> a <- subset(s,X1+X2+X3 >3 & X1+X2+X3 < 8)

> #first theree sample output

> head(s,n=3)

  X1 X2 X3      probs

1  1  1  1 0.00462963

2  2  1  1 0.00462963

3  3  1  1 0.00462963

> prob(a)

[1] 0.1574074
```

…………………………………………………………………………………………

#b) script:

```
#all rolls are identical

b <- subset(s,X1==X2 & X2==X3)

#first three samples output

head(b,n=3)

prob(b)
```

console answer(b):

#b)

> #all rolls are identical

> b <- subset(s,X1==X2 & X2==X3)

> #first three samples output

> head(b,n=3)

```
   X1 X2 X3      probs
1   1  1  1 0.00462963
44  2  2  2 0.00462963
87  3  3  3 0.00462963
```

> prob(b)

[1] 0.02777778

…………………………………………………………………….

#c) script:

#only two of the three rolls are identical.

c <- subset(s,X1==X2 & X1!=X3 | X2==X3 & X2!=X1| X1==X3 & X2!=X3)

#first three sample output

head(c,n=3)

prob(c).

console answer(c):

#c)

> #only two of the three rolls are identical.

> c <- subset(s,X1==X2 & X1!=X3 | X2==X3 & X2!=X1| X1==X3 & X2!=X3)

> #first three sample output

```
> head(c,n=3)
  X1 X2 X3      probs
2  2  1  1 0.00462963
3  3  1  1 0.00462963
4  4  1  1 0.00462963
> prob(c)
[1] 0.4166667
```

………

#d) script

```
# None of the three rolls are identical
d <- subset(s,X1!=X2 & X2!=X3 & X1!=X3)
#first three sample output
head(d,n=3)
prob(d)
```

> #d) console

```
> # None of the three rolls are identical
> d <- subset(s,X1!=X2 & X2!=X3 & X1!=X3)
> #first three sample output
> head(d,n=3)
   X1 X2 X3      probs
9   3  2  1 0.00462963
10  4  2  1 0.00462963
11  5  2  1 0.00462963
> prob(d)
```

[1] 0.5555556

…………………………………………………………………………………………………..

#e) script

probablity that only two of three rolls are identical

#given sum of the rolls are greater than 3 and less than 8

#using conditional R construct

prob(c,given = a)

#Alternatively

#we can use P(C/A) = P(C intersection A)/P(A)

prob(intersect(c,a))/prob(a)

> #e) console:

> # probablity that only two of three rolls are identical

> #given sum of the rolls are greater than 3 and less than 8

> #using conditional R construct

> prob(c,given = a)

[1] 0.6176471

>

> #Alternatively

> #we can use P(C/A) = P(C intersection A)/P(A)

> prob(intersect(c,a))/prob(a)

[1] 0.6176471

>

……………………………………………………………………………

Part 3 functions:

Scripts:

```
#Part 3 Functions

sum_of_first_N_even_squares <- function(n){

  m<-0 # counter for counting the number of even numbers starts from 0

  sum1<-0 # to add squared number starts from 0

  num1<-0 # current even number and starts from value 0


  while (m<n) {

    num1<-num1+2  # in every loop, current even number is equall to 2 plus previous number

    sum1<-sum1+num1**2

    m<-m+1


  }

  return(sum1)

}


sum_of_first_N_even_squares(0)

sum_of_first_N_even_squares(2)

sum_of_first_N_even_squares(5)

sum_of_first_N_even_squares(10)
```

Console:

```
> #Part 3 Functions
> sum_of_first_N_even_squares <- function(n){
+   m<-0 # counter for counting the number of even numbers starts from 0
+   sum1<-0 # to add squared number starts from 0
+   num1<-0 # current even number and starts from value 0
+
+   while (m<n) {
+     num1<-num1+2  # in every loop, current even number is equall to 2 plus previous number
+     sum1<-sum1+num1**2
+     m<-m+1
+
+   }
+   return(sum1)
+ }
>
> sum_of_first_N_even_squares(0)
[1] 0
> sum_of_first_N_even_squares(2)
[1] 20
> sum_of_first_N_even_squares(5)
[1] 220
> sum_of_first_N_even_squares(10)
[1] 1540
```

 # Part 4 R

```
tesla <- read.csv("https://people.bu.edu/kalathur/datasets/TSLA2022.csv")

#to compute the probablity space for given data

tsla <- probspace(tesla)
```

……………………………………………………………………………………….

```
#a)


sm<-summary(tsla$Close)

#changing names of the variables

names(sm)<- c("Min","Q1","Q2","Mean","Q3","Max")

sm
```

Console(a):

```
> # Part 4 R

> tesla <- read.csv("https://people.bu.edu/kalathur/datasets/TSLA2022.csv")

> #to compute the probablity space for given data

> tsla <- probspace(tesla)

> #a)

> sm<-summary(tsla$Close)

> #changing names of the variables

> names(sm)<- c("Min","Q1","Q2","Mean","Q3","Max")

> sm

  Min    Q1    Q2  Mean    Q3   Max

109.0 225.0 272.0 263.1 302.5 400.0
```

…………………………………………………………………………………………

#b)

min_close<- subset(tsla,tsla$Close==min(tsla$Close))

min_close

rownames(min_close)

min_close$Date

min_close$Close

paste("The minimum Tesla value of",min_close$Close, "is at row",rownames(min_close), "on", min_close$Date)

console(b):

> #b)

> min_close<- subset(tsla,tsla$Close==min(tsla$Close))

> min_close

```
      Date Open High Low Close    Volume       probs
248 12/27/22  118  120 109   109 208643400 0.003984064
```

> rownames(min_close)

[1] "248"

> min_close$Date

[1] "12/27/22"

> min_close$Close

[1] 109

> paste("The minimum Tesla value of",min_close$Close, "is at row",rownames(min_close), "on", min_close$Date)

[1] "The minimum Tesla value of 109 is at row 248 on 12/27/22"

…….

#c) scripts:

```
max_close <- subset(tsla,tsla$Close==max(tsla$Close))

max_close

rownames(max_close)

max_close$Date

max_close$Close

paste("The maximum Tesla value of",max_close$Close, "is at row",rownames(max_close),"on",max_close$Date)
```

console(c):

```
> #c)

> max_close <- subset(tsla,tsla$Close==max(tsla$Close))

> max_close

   Date Open High Low Close    Volume       probs

1 1/3/22  383  400 379   400 103931400 0.003984064

> rownames(max_close)

[1] "1"

> max_close$Date

[1] "1/3/22"

> max_close$Close

[1] 400

> paste("The maximum Tesla value of",max_close$Close, "is at row",rownames(max_close),"on",max_close$Date)

[1] "The maximum Tesla value of 400 is at row 1 on 1/3/22"
```

…………………………………………………………………………………………………….

#d) scripts:

```
high_close_low_open <- subset(tsla,tsla$Close>tsla$Open)

#probability of tesla being its closing price greater than its opening price

# total number of rows that has higher closing price than opening price

#divide by total number of days stock trade happens


probability_high_close_low_open<- prob(high_close_low_open)

probability_high_close_low_open


#Alternatively:

#probablity can be calculated using R

nrow(high_close_low_open)/nrow(tsla)
```

console:

```
> #d)

> high_close_low_open <- subset(tsla,tsla$Close>tsla$Open)

> #probability of tesla being its closing price greater than its opening price

> # total number of rows that has higher closing price than opening price

> #divide by total number of days stock trade happens

> probability_high_close_low_open<- prob(high_close_low_open)

> probability_high_close_low_open

[1] 0.4501992

> #Alternatively:

> #probablity can be calculated using R

> nrow(high_close_low_open)/nrow(tsla)
```

[1] 0.4501992

………

#e)

high_vol_trade <- subset(tsla,tsla$Volume>100000000)

#probablity that on any given day, the tesla traidn volume

#woube be greater than 100 millin shares is

probablility_high_vol_trade <- prob(high_vol_trade)

probablility_high_vol_trade

#alternatively

#probablity can be calculated by using R command

nrow(high_vol_trade)/nrow(tsla)

console:

> #e)

> high_vol_trade <- subset(tsla,tsla$Volume>100000000)

> #probablity that on any given day, the tesla traidn volume

> #woube be greater than 100 millin shares is

> probablility_high_vol_trade <- prob(high_vol_trade)

> probablility_high_vol_trade

[1] 0.2231076

> #alternatively

> #probablity can be calculated by using R command

> nrow(high_vol_trade)/nrow(tsla)

[1] 0.2231076

……

#f)

#for conditional probablity

#probablity that on any given day,tesla closing price is greater than opening price

#given tesla trade volume is greater than 100 mil

#using R command

prob(high_close_low_open, given = high_vol_trade)

#Alternatively,

#P(A/B) = P(A intersect B)/P(B)

conditional_probablity <-
prob(intersect(high_close_low_open,high_vol_trade))/prob(high_vol_trade)

conditional_probablity

console:

> #f)

> #for conditional probablity

> #probablity that on any given day,tesla closing price is greater than opening price

> #given tesla trade volume is greater than 100 mil

> #using R command

> prob(high_close_low_open, given = high_vol_trade)

[1] 0.4642857

> #Alternatively,

> #P(A/B) = P(A intersect B)/P(B)

> conditional_probablity <-
prob(intersect(high_close_low_open,high_vol_trade))/prob(high_vol_trade)

> conditional_probablity

```
[1] 0.4642857
```

………

```
#g)

# there are 251 days trading happenned = nrow(tsla)

Total_number_of_Shares <-  nrow(tsla)

# total money spent buying all 251 shares in its low price of respective day

buy_price<- sum(tesla$Low)

buy_price

# sell price would be closing  price of last day 251 st day

g <- subset(tesla,rownames(tesla)==nrow(tesla))

sell_price<- nrow(tesla)* g$Close

sell_price

#finding loss or gain selling all shares

loss.gain<- sell_price - buy_price

loss.gain

# as selling 251 shares get $33516 while $64873 was spent buying those shares.

#that is why there will be loss in this tradinng

#loss_amount 33516

paste("there will be ",loss.gain,"gain after selling all the shares ")
```

console:

```
#g)

> # there are 251 days trading happenned = nrow(tsla)

> Total_number_of_Shares <-  nrow(tsla)

> # total money spent buying all 251 shares in its low price of respective day
```

```
> buy_price<- sum(tesla$Low)

> buy_price

[1] 64389

> # sell price would be closing  price of last day 251 st day

> last_day_trade <- subset(tesla,rownames(tesla)==nrow(tesla))

> sell_price<- nrow(tesla)* last_day_trade$Close

> sell_price

[1] 30873

> #finding loss or gain selling all shares

> loss_gain<- sell_price - buy_price

> loss_gain

[1] -33516

> # as selling 251 shares get $33516 while $64873 was spent buying those shares.

> #that is why there will be loss in this trading

> #loss_amount 33516

> paste("there will be ",loss.gain,"gain after selling all the shares ")

[1] "there will be  -33516 gain after selling all the shares "
```

……

The End
