CS544 Module 2 Assignment

© 2022, Suresh Kalathur, Boston University. All Rights Reserved. The following document should not be disseminated outside the purview of its intended purpose

**Hao Wu**

**02/21/2022**

**Part1) Probability - 25 points**

Use the Bayes theorem to calculate the following probabilities. Show the individual steps of the Bayes theorem. You can use R for the calculations. Use the Word document for this part.

Suppose that in a particular state, among 10000 people surveyed, 4250 people are in the age group 18-34 years, 2850 people are in the age group 35-49 years, 1640 people are in the age group 50-64 years, and the remaining are 65 years & over.

Out of those in the age group 18-34 years, 1062 people had a BMI of above 30. Of those in the age group 35-49 years, 1710 people had a BMI of above 30. Among those in the 50-64 years range, 656 people had a BMI of above 30. In the last age group, 189 people had a BMI of above 30.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Age | BMI | BMI Percent | Total | Percent |
| 18-34 | 1062 | 1062/4250 | 4250 | 0.425 |
| 35-49 | 1710 | 1710/2850 | 2850 | 0.285 |
| 50-64 | 656 | 1640/656 | 1640 | 0.164 |
| Above64 | 189 | 189/1260 | 1260 | 0.126 |

1. What is the probability that a randomly selected person in this survey will have a BMI of above 30?

= (1062+1710+656+189)/10000

= 0.3617

The probability that a randomly selected person in this survey will have a BMI of above 30 is 36.17%

1. If a randomly selected person had a BMI of above 30, what is the probability of that person being in the age group 18-34 years?

=0.1062/0.3617

=0.2936

The probability that a randomly selected person had a BMI of above 30 in this survey that person being in the age group 18-34 years is 29.36%

1. If a randomly selected person had a BMI of above 30, what is the probability of that person being in the age group 35-49 years?

=0.1710/0.3617

=0.4728

The probability that a randomly selected person had a BMI of above 30 in this survey that person being in the age group 35-49 years is 47.28%

1. If a randomly selected person had a BMI of above 30, what is the probability of that person being in the age group 50-64 years?

=0.6560/0.3617

=0.1814

The probability that a randomly selected person had a BMI of above 30 in this survey that person being in the age group 35-49 years is 18.14%

e) If a randomly selected person had a BMI of above 30, what is the probability of that person being in the 65 years & over?

=0.0189/0.3617

=0.0523

The probability that a randomly selected person had a BMI of above 30 in this survey that person being in the age group above 65 years is 5.23%

##Part 2

library(prob)

dice <- rolldie(3,6)

dice

##Question a

dicetotal <- rowSums(dice[c("X1","X2","X3")])

dicetotal

##Calculate each roll probability

dice$probs <- round(1/nrow(dice),digits = 5)

dice6to10 <- dice[dicetotal > 6 & dicetotal <10,]

dice6to10

Q1 <- colSums(dice6to10[4])

Q1

##Question b

## Get the index of all the three rolls are identical

threerolls\_index <- dice$X1 == dice$X2 & dice$X2 == dice$X3

## Get the rows of all the three rolls are identical

threerolls <- dice[threerolls\_index == "TRUE",]

threerolls

## Sum the probability

Q2 <- colSums(threerolls[4])

Q2

##Question c

##Get the index of only two rolls are identical and the answer need except the three rolls identical

tworolls\_index <- (dice$X1 == dice$X2 | dice$X1 == dice$X3 |dice$X2 == dice$X3) &(!threerolls\_index)

tworolls <- dice[tworolls\_index == "TRUE",]

tworolls

Q3 <- colSums(tworolls[4])

Q3

##Question d

norolls\_index <- dice$X1 != dice$X2 & dice$X2 != dice$X3 & dice$X1 != dice$X3

norolls <- dice[norolls\_index == "TRUE",]

norolls

Q4 <- colSums(norolls[4])

Q4

##Question e

dice2total <- rowSums(tworolls[c("X1","X2","X3")])

dice2total

dice2 <- tworolls[dice2total > 6 & dice2total < 10,]

dice2

Q5 <- colSums(dice2[4])

Q5

##Part 3

##Question a

sum\_of\_first\_N\_even\_squares <- function(n) {

sum = 0 ## set a value sum as a initial value

for (i in seq(2, by = 2, len = n)){

sum = sum + i\*i ## take each of number in the sequence to square, then accumulative it to sum int.

}

return(sum) ## return our result

}

sum\_of\_first\_N\_even\_squares(2)

sum\_of\_first\_N\_even\_squares(5)

sum\_of\_first\_N\_even\_squares(10)

##Question b

sum\_of\_first\_N\_even\_squares\_V2 <- function(n) {

vec = seq(2, by = 2, len = n) ##Get the sequence

sum = sum(vec^2) ## Get the sum of square of the sequence

return(sum) ## Return the result

}

sum\_of\_first\_N\_even\_squares\_V2(2)

sum\_of\_first\_N\_even\_squares\_V2(5)

sum\_of\_first\_N\_even\_squares\_V2(10)

##Part 4

tsla <- read.csv("https://people.bu.edu/kalathur/datasets/TSLA.csv")

head(tsla)

##Question a

sm <- summary(tsla[5])

sm

which(tsla[5] == min(tsla[5]))

##Question b

n <- which(tsla[5] == min(tsla[5]))

n ## n is the index of which row is minimum

for( i in n){ ## for loop to print out each sentence

cat("The minimum Tesla value of",min(tsla[5]), "is at row", i,"on", tsla[i,1],"\n")

}

##Question c

d <-which(tsla[5] == max(tsla[5]))

for( i in d){ ## for loop to print out each sentence

cat("The maximum Tesla value of",max(tsla[5]), "is at row", i,"on", tsla[i,1],"\n")

}

##Question d

closehigh <- tsla[tsla[2]<tsla[5],] ##get the row which close price greater than open price

probd <- nrow(closehigh)/nrow(tsla) ## get the probability

probd

##Question e

vol <- tsla[tsla[6]>20000000,] ## Take the row which volumn is greater than 20 millions

probv <- nrow(vol)/nrow(tsla) ## get the probability

probv

##Question f

f <- closehigh[closehigh[6]>20000000,]## From the perious data frame which we create for the close price greater than open price

probf <- nrow(f)/nrow(tsla)

probf

##Question g

totalhold <- colSums(tsla["Open"]) ## Accumulate the price which on the open column

totalhold

netgain <- (tsla[nrow(tsla),5]\*nrow(tsla)) - totalhold ##Use the last day close price multiple the day then minus the total price

netgain

##Question h

totallow <- colSums(tsla["Low"])

totallow

netgainlow <-(tsla[nrow(tsla),5]\*nrow(tsla)) - totallow

netgainlow

##Question i

totalhigh <- colSums(tsla["High"])

totalhigh

netgainhigh <- (tsla[nrow(tsla),5]\*nrow(tsla)) - totalhigh

netgainhigh

> ##Part 2

> library(prob)

> dice <- rolldie(3,6)

> dice

X1 X2 X3

1 1 1 1

2 2 1 1

3 3 1 1

4 4 1 1

5 5 1 1

6 6 1 1

7 1 2 1

8 2 2 1

9 3 2 1

10 4 2 1

11 5 2 1

12 6 2 1

13 1 3 1

14 2 3 1

15 3 3 1

16 4 3 1

17 5 3 1

18 6 3 1

19 1 4 1

20 2 4 1

21 3 4 1

22 4 4 1

23 5 4 1

24 6 4 1

25 1 5 1

26 2 5 1

27 3 5 1

28 4 5 1

29 5 5 1

30 6 5 1

31 1 6 1

32 2 6 1

33 3 6 1

34 4 6 1

35 5 6 1

36 6 6 1

37 1 1 2

38 2 1 2

39 3 1 2

40 4 1 2

41 5 1 2

42 6 1 2

43 1 2 2

44 2 2 2

45 3 2 2

46 4 2 2

47 5 2 2

48 6 2 2

49 1 3 2

50 2 3 2

51 3 3 2

52 4 3 2

53 5 3 2

54 6 3 2

55 1 4 2

56 2 4 2

57 3 4 2

58 4 4 2

59 5 4 2

60 6 4 2

61 1 5 2

62 2 5 2

63 3 5 2

64 4 5 2

65 5 5 2

66 6 5 2

67 1 6 2

68 2 6 2

69 3 6 2

70 4 6 2

71 5 6 2

72 6 6 2

73 1 1 3

74 2 1 3

75 3 1 3

76 4 1 3

77 5 1 3

78 6 1 3

79 1 2 3

80 2 2 3

81 3 2 3

82 4 2 3

83 5 2 3

84 6 2 3

85 1 3 3

86 2 3 3

87 3 3 3

88 4 3 3

89 5 3 3

90 6 3 3

91 1 4 3

92 2 4 3

93 3 4 3

94 4 4 3

95 5 4 3

96 6 4 3

97 1 5 3

98 2 5 3

99 3 5 3

100 4 5 3

101 5 5 3

102 6 5 3

103 1 6 3

104 2 6 3

105 3 6 3

106 4 6 3

107 5 6 3

108 6 6 3

109 1 1 4

110 2 1 4

111 3 1 4

112 4 1 4

113 5 1 4

114 6 1 4

115 1 2 4

116 2 2 4

117 3 2 4

118 4 2 4

119 5 2 4

120 6 2 4

121 1 3 4

122 2 3 4

123 3 3 4

124 4 3 4

125 5 3 4

126 6 3 4

127 1 4 4

128 2 4 4

129 3 4 4

130 4 4 4

131 5 4 4

132 6 4 4

133 1 5 4

134 2 5 4

135 3 5 4

136 4 5 4

137 5 5 4

138 6 5 4

139 1 6 4

140 2 6 4

141 3 6 4

142 4 6 4

143 5 6 4

144 6 6 4

145 1 1 5

146 2 1 5

147 3 1 5

148 4 1 5

149 5 1 5

150 6 1 5

151 1 2 5

152 2 2 5

153 3 2 5

154 4 2 5

155 5 2 5

156 6 2 5

157 1 3 5

158 2 3 5

159 3 3 5

160 4 3 5

161 5 3 5

162 6 3 5

163 1 4 5

164 2 4 5

165 3 4 5

166 4 4 5

167 5 4 5

168 6 4 5

169 1 5 5

170 2 5 5

171 3 5 5

172 4 5 5

173 5 5 5

174 6 5 5

175 1 6 5

176 2 6 5

177 3 6 5

178 4 6 5

179 5 6 5

180 6 6 5

181 1 1 6

182 2 1 6

183 3 1 6

184 4 1 6

185 5 1 6

186 6 1 6

187 1 2 6

188 2 2 6

189 3 2 6

190 4 2 6

191 5 2 6

192 6 2 6

193 1 3 6

194 2 3 6

195 3 3 6

196 4 3 6

197 5 3 6

198 6 3 6

199 1 4 6

200 2 4 6

201 3 4 6

202 4 4 6

203 5 4 6

204 6 4 6

205 1 5 6

206 2 5 6

207 3 5 6

208 4 5 6

209 5 5 6

210 6 5 6

211 1 6 6

212 2 6 6

213 3 6 6

214 4 6 6

215 5 6 6

216 6 6 6

> ##Question a

> dicetotal <- rowSums(dice[c("X1","X2","X3")])

> dicetotal

[1] 3 4 5 6 7 8 4 5 6 7 8 9 5 6 7 8 9 10 6 7 8 9 10 11 7 8 9 10 11 12 8 9 10 11 12 13 4 5 6 7 8 9

[43] 5 6 7 8 9 10 6 7 8 9 10 11 7 8 9 10 11 12 8 9 10 11 12 13 9 10 11 12 13 14 5 6 7 8 9 10 6 7 8 9 10 11

[85] 7 8 9 10 11 12 8 9 10 11 12 13 9 10 11 12 13 14 10 11 12 13 14 15 6 7 8 9 10 11 7 8 9 10 11 12 8 9 10 11 12 13

[127] 9 10 11 12 13 14 10 11 12 13 14 15 11 12 13 14 15 16 7 8 9 10 11 12 8 9 10 11 12 13 9 10 11 12 13 14 10 11 12 13 14 15

[169] 11 12 13 14 15 16 12 13 14 15 16 17 8 9 10 11 12 13 9 10 11 12 13 14 10 11 12 13 14 15 11 12 13 14 15 16 12 13 14 15 16 17

[211] 13 14 15 16 17 18

> ##Calculate each roll probability

> dice$probs <- round(1/nrow(dice),digits = 5)

>

> dice6to10 <- dice[dicetotal > 6 & dicetotal <10,]

> dice6to10

X1 X2 X3 probs

5 5 1 1 0.00463

6 6 1 1 0.00463

10 4 2 1 0.00463

11 5 2 1 0.00463

12 6 2 1 0.00463

15 3 3 1 0.00463

16 4 3 1 0.00463

17 5 3 1 0.00463

20 2 4 1 0.00463

21 3 4 1 0.00463

22 4 4 1 0.00463

25 1 5 1 0.00463

26 2 5 1 0.00463

27 3 5 1 0.00463

31 1 6 1 0.00463

32 2 6 1 0.00463

40 4 1 2 0.00463

41 5 1 2 0.00463

42 6 1 2 0.00463

45 3 2 2 0.00463

46 4 2 2 0.00463

47 5 2 2 0.00463

50 2 3 2 0.00463

51 3 3 2 0.00463

52 4 3 2 0.00463

55 1 4 2 0.00463

56 2 4 2 0.00463

57 3 4 2 0.00463

61 1 5 2 0.00463

62 2 5 2 0.00463

67 1 6 2 0.00463

75 3 1 3 0.00463

76 4 1 3 0.00463

77 5 1 3 0.00463

80 2 2 3 0.00463

81 3 2 3 0.00463

82 4 2 3 0.00463

85 1 3 3 0.00463

86 2 3 3 0.00463

87 3 3 3 0.00463

91 1 4 3 0.00463

92 2 4 3 0.00463

97 1 5 3 0.00463

110 2 1 4 0.00463

111 3 1 4 0.00463

112 4 1 4 0.00463

115 1 2 4 0.00463

116 2 2 4 0.00463

117 3 2 4 0.00463

121 1 3 4 0.00463

122 2 3 4 0.00463

127 1 4 4 0.00463

145 1 1 5 0.00463

146 2 1 5 0.00463

147 3 1 5 0.00463

151 1 2 5 0.00463

152 2 2 5 0.00463

157 1 3 5 0.00463

181 1 1 6 0.00463

182 2 1 6 0.00463

187 1 2 6 0.00463

> Q1 <- colSums(dice6to10[4])

> Q1

probs

0.28243

>

> ##Question b

>

>

> ## Get the index of all the three rolls are identical

> threerolls\_index <- dice$X1 == dice$X2 & dice$X2 == dice$X3

>

> ## Get the rows of all the three rolls are identical

> threerolls <- dice[threerolls\_index == "TRUE",]

> threerolls

X1 X2 X3 probs

1 1 1 1 0.00463

44 2 2 2 0.00463

87 3 3 3 0.00463

130 4 4 4 0.00463

173 5 5 5 0.00463

216 6 6 6 0.00463

>

> ## Sum the probability

> Q2 <- colSums(threerolls[4])

> Q2

probs

0.02778

>

>

>

> ##Question c

> ##Get the index of only two rolls are identical and the answer need except the three rolls identical

> tworolls\_index <- (dice$X1 == dice$X2 | dice$X1 == dice$X3 |dice$X2 == dice$X3) &(!threerolls\_index)

> tworolls <- dice[tworolls\_index == "TRUE",]

> tworolls

X1 X2 X3 probs

2 2 1 1 0.00463

3 3 1 1 0.00463

4 4 1 1 0.00463

5 5 1 1 0.00463

6 6 1 1 0.00463

7 1 2 1 0.00463

8 2 2 1 0.00463

13 1 3 1 0.00463

15 3 3 1 0.00463

19 1 4 1 0.00463

22 4 4 1 0.00463

25 1 5 1 0.00463

29 5 5 1 0.00463

31 1 6 1 0.00463

36 6 6 1 0.00463

37 1 1 2 0.00463

38 2 1 2 0.00463

43 1 2 2 0.00463

45 3 2 2 0.00463

46 4 2 2 0.00463

47 5 2 2 0.00463

48 6 2 2 0.00463

50 2 3 2 0.00463

51 3 3 2 0.00463

56 2 4 2 0.00463

58 4 4 2 0.00463

62 2 5 2 0.00463

65 5 5 2 0.00463

68 2 6 2 0.00463

72 6 6 2 0.00463

73 1 1 3 0.00463

75 3 1 3 0.00463

80 2 2 3 0.00463

81 3 2 3 0.00463

85 1 3 3 0.00463

86 2 3 3 0.00463

88 4 3 3 0.00463

89 5 3 3 0.00463

90 6 3 3 0.00463

93 3 4 3 0.00463

94 4 4 3 0.00463

99 3 5 3 0.00463

101 5 5 3 0.00463

105 3 6 3 0.00463

108 6 6 3 0.00463

109 1 1 4 0.00463

112 4 1 4 0.00463

116 2 2 4 0.00463

118 4 2 4 0.00463

123 3 3 4 0.00463

124 4 3 4 0.00463

127 1 4 4 0.00463

128 2 4 4 0.00463

129 3 4 4 0.00463

131 5 4 4 0.00463

132 6 4 4 0.00463

136 4 5 4 0.00463

137 5 5 4 0.00463

142 4 6 4 0.00463

144 6 6 4 0.00463

145 1 1 5 0.00463

149 5 1 5 0.00463

152 2 2 5 0.00463

155 5 2 5 0.00463

159 3 3 5 0.00463

161 5 3 5 0.00463

166 4 4 5 0.00463

167 5 4 5 0.00463

169 1 5 5 0.00463

170 2 5 5 0.00463

171 3 5 5 0.00463

172 4 5 5 0.00463

174 6 5 5 0.00463

179 5 6 5 0.00463

180 6 6 5 0.00463

181 1 1 6 0.00463

186 6 1 6 0.00463

188 2 2 6 0.00463

192 6 2 6 0.00463

195 3 3 6 0.00463

198 6 3 6 0.00463

202 4 4 6 0.00463

204 6 4 6 0.00463

209 5 5 6 0.00463

210 6 5 6 0.00463

211 1 6 6 0.00463

212 2 6 6 0.00463

213 3 6 6 0.00463

214 4 6 6 0.00463

215 5 6 6 0.00463

> Q3 <- colSums(tworolls[4])

> Q3

probs

0.4167

>

> ##Question d

> norolls\_index <- dice$X1 != dice$X2 & dice$X2 != dice$X3 & dice$X1 != dice$X3

> norolls <- dice[norolls\_index == "TRUE",]

> norolls

X1 X2 X3 probs

9 3 2 1 0.00463

10 4 2 1 0.00463

11 5 2 1 0.00463

12 6 2 1 0.00463

14 2 3 1 0.00463

16 4 3 1 0.00463

17 5 3 1 0.00463

18 6 3 1 0.00463

20 2 4 1 0.00463

21 3 4 1 0.00463

23 5 4 1 0.00463

24 6 4 1 0.00463

26 2 5 1 0.00463

27 3 5 1 0.00463

28 4 5 1 0.00463

30 6 5 1 0.00463

32 2 6 1 0.00463

33 3 6 1 0.00463

34 4 6 1 0.00463

35 5 6 1 0.00463

39 3 1 2 0.00463

40 4 1 2 0.00463

41 5 1 2 0.00463

42 6 1 2 0.00463

49 1 3 2 0.00463

52 4 3 2 0.00463

53 5 3 2 0.00463

54 6 3 2 0.00463

55 1 4 2 0.00463

57 3 4 2 0.00463

59 5 4 2 0.00463

60 6 4 2 0.00463

61 1 5 2 0.00463

63 3 5 2 0.00463

64 4 5 2 0.00463

66 6 5 2 0.00463

67 1 6 2 0.00463

69 3 6 2 0.00463

70 4 6 2 0.00463

71 5 6 2 0.00463

74 2 1 3 0.00463

76 4 1 3 0.00463

77 5 1 3 0.00463

78 6 1 3 0.00463

79 1 2 3 0.00463

82 4 2 3 0.00463

83 5 2 3 0.00463

84 6 2 3 0.00463

91 1 4 3 0.00463

92 2 4 3 0.00463

95 5 4 3 0.00463

96 6 4 3 0.00463

97 1 5 3 0.00463

98 2 5 3 0.00463

100 4 5 3 0.00463

102 6 5 3 0.00463

103 1 6 3 0.00463

104 2 6 3 0.00463

106 4 6 3 0.00463

107 5 6 3 0.00463

110 2 1 4 0.00463

111 3 1 4 0.00463

113 5 1 4 0.00463

114 6 1 4 0.00463

115 1 2 4 0.00463

117 3 2 4 0.00463

119 5 2 4 0.00463

120 6 2 4 0.00463

121 1 3 4 0.00463

122 2 3 4 0.00463

125 5 3 4 0.00463

126 6 3 4 0.00463

133 1 5 4 0.00463

134 2 5 4 0.00463

135 3 5 4 0.00463

138 6 5 4 0.00463

139 1 6 4 0.00463

140 2 6 4 0.00463

141 3 6 4 0.00463

143 5 6 4 0.00463

146 2 1 5 0.00463

147 3 1 5 0.00463

148 4 1 5 0.00463

150 6 1 5 0.00463

151 1 2 5 0.00463

153 3 2 5 0.00463

154 4 2 5 0.00463

156 6 2 5 0.00463

157 1 3 5 0.00463

158 2 3 5 0.00463

160 4 3 5 0.00463

162 6 3 5 0.00463

163 1 4 5 0.00463

164 2 4 5 0.00463

165 3 4 5 0.00463

168 6 4 5 0.00463

175 1 6 5 0.00463

176 2 6 5 0.00463

177 3 6 5 0.00463

178 4 6 5 0.00463

182 2 1 6 0.00463

183 3 1 6 0.00463

184 4 1 6 0.00463

185 5 1 6 0.00463

187 1 2 6 0.00463

189 3 2 6 0.00463

190 4 2 6 0.00463

191 5 2 6 0.00463

193 1 3 6 0.00463

194 2 3 6 0.00463

196 4 3 6 0.00463

197 5 3 6 0.00463

199 1 4 6 0.00463

200 2 4 6 0.00463

201 3 4 6 0.00463

203 5 4 6 0.00463

205 1 5 6 0.00463

206 2 5 6 0.00463

207 3 5 6 0.00463

208 4 5 6 0.00463

> Q4 <- colSums(norolls[4])

> Q4

probs

0.5556

>

> ##Question e

> dice2total <- rowSums(tworolls[c("X1","X2","X3")])

> dice2total

2 3 4 5 6 7 8 13 15 19 22 25 29 31 36 37 38 43 45 46 47 48 50 51 56 58 62 65 68 72 73 75 80

4 5 6 7 8 4 5 5 7 6 9 7 11 8 13 4 5 5 7 8 9 10 7 8 8 10 9 12 10 14 5 7 7

81 85 86 88 89 90 93 94 99 101 105 108 109 112 116 118 123 124 127 128 129 131 132 136 137 142 144 145 149 152 155 159 161

8 7 8 10 11 12 10 11 11 13 12 15 6 9 8 10 10 11 9 10 11 13 14 13 14 14 16 7 11 9 12 11 13

166 167 169 170 171 172 174 179 180 181 186 188 192 195 198 202 204 209 210 211 212 213 214 215

13 14 11 12 13 14 16 16 17 8 13 10 14 12 15 14 16 16 17 13 14 15 16 17

> dice2 <- tworolls[dice2total > 6 & dice2total < 10,]

> dice2

X1 X2 X3 probs

5 5 1 1 0.00463

6 6 1 1 0.00463

15 3 3 1 0.00463

22 4 4 1 0.00463

25 1 5 1 0.00463

31 1 6 1 0.00463

45 3 2 2 0.00463

46 4 2 2 0.00463

47 5 2 2 0.00463

50 2 3 2 0.00463

51 3 3 2 0.00463

56 2 4 2 0.00463

62 2 5 2 0.00463

75 3 1 3 0.00463

80 2 2 3 0.00463

81 3 2 3 0.00463

85 1 3 3 0.00463

86 2 3 3 0.00463

112 4 1 4 0.00463

116 2 2 4 0.00463

127 1 4 4 0.00463

145 1 1 5 0.00463

152 2 2 5 0.00463

181 1 1 6 0.00463

> Q5 <- colSums(dice2[4])

> Q5

probs

0.11112

>

>

> ##Part 3

> ##Question a

> sum\_of\_first\_N\_even\_squares <- function(n) {

+ sum = 0 ## set a value sum as a initial value

+ for (i in seq(2, by = 2, len = n)){

+ sum = sum + i\*i ## take each of number in the sequence to square, then accumulative it to sum int.

+

+ }

+ return(sum) ## return our result

+ }

> 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

>

>

> ##Question b

> sum\_of\_first\_N\_even\_squares\_V2 <- function(n) {

+ vec = seq(2, by = 2, len = n) ##Get the sequence

+ sum = sum(vec^2) ## Get the sum of square of the sequence

+ return(sum) ## Return the result

+ }

> sum\_of\_first\_N\_even\_squares\_V2(2)

[1] 20

> sum\_of\_first\_N\_even\_squares\_V2(5)

[1] 220

> sum\_of\_first\_N\_even\_squares\_V2(10)

[1] 1540

>

> ##Part 4

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

> head(tsla)

Date Open High Low Close Volume

1 2021-01-04 719 744 717 730 48638200

2 2021-01-05 724 741 719 735 32245200

3 2021-01-06 758 774 749 756 44700000

4 2021-01-07 778 817 775 816 51498900

5 2021-01-08 856 884 838 880 75055500

6 2021-01-11 849 854 804 811 59301600

> ##Question a

>

> sm <- summary(tsla[5])

> sm

Close

Min. : 563

1st Qu.: 668

Median : 730

Mean : 780

3rd Qu.: 850

Max. :1230

> which(tsla[5] == min(tsla[5]))

[1] 44 95

> ##Question b

> n <- which(tsla[5] == min(tsla[5]))

> n ## n is the index of which row is minimum

[1] 44 95

> for( i in n){ ## for loop to print out each sentence

+ cat("The minimum Tesla value of",min(tsla[5]), "is at row", i,"on", tsla[i,1],"\n")

+ }

The minimum Tesla value of 563 is at row 44 on 2021-03-08

The minimum Tesla value of 563 is at row 95 on 2021-05-19

>

> ##Question c

> d <-which(tsla[5] == max(tsla[5]))

> for( i in d){ ## for loop to print out each sentence

+ cat("The maximum Tesla value of",max(tsla[5]), "is at row", i,"on", tsla[i,1],"\n")

+ }

The maximum Tesla value of 1230 is at row 213 on 2021-11-04

>

> ##Question d

> closehigh <- tsla[tsla[2]<tsla[5],] ##get the row which close price greater than open price

> probd <- nrow(closehigh)/nrow(tsla) ## get the probability

> probd

[1] 0.5

>

> ##Question e

> vol <- tsla[tsla[6]>20000000,] ## Take the row which volumn is greater than 20 millions

> probv <- nrow(vol)/nrow(tsla) ## get the probability

> probv

[1] 0.7142857

>

> ##Question f

> f <- closehigh[closehigh[6]>20000000,]## From the perious data frame which we create for the close price greater than open price

> probf <- nrow(f)/nrow(tsla)

> probf

[1] 0.3849206

>

> ##Question g

> totalhold <- colSums(tsla["Open"]) ## Accumulate the price which on the open column

> totalhold

Open

196525

> netgain <- (tsla[nrow(tsla),5]\*nrow(tsla)) - totalhold ##Use the last day close price multiple the day then minus the total price

> netgain

Open

69839

>

> ##Question h

> totallow <- colSums(tsla["Low"])

> totallow

Low

192202

> netgainlow <-(tsla[nrow(tsla),5]\*nrow(tsla)) - totallow

> netgainlow

Low

74162

>

> ##Question i

> totalhigh <- colSums(tsla["High"])

> totalhigh

High

200523

> netgainhigh <- (tsla[nrow(tsla),5]\*nrow(tsla)) - totalhigh

> netgainhigh

High

65841