coursework

nikoya

2023-02-18

library(knitr)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.0 ✔ purrr 1.0.1   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.5.0   
## ✔ readr 2.1.3 ✔ forcats 0.5.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

#Question 1)

##Part1

##Question 1a)  
gpas <- c(2.0, 2.1, 3.2, 3.1, 3.6, 2.8, 1.9, 1.8, 2.8, 3.0, 1.9, 1.8, 2.0, 2.1, 2.0, 2.2, 2.2, 3.4, 3.5, 3.0, 3.1, 2.0, 2.3, 2.6, 3.1, 3.3, 3.7, 2.9, 2.3, 2.0)  
mean(gpas)

## [1] 2.59

##Part2

median(gpas)

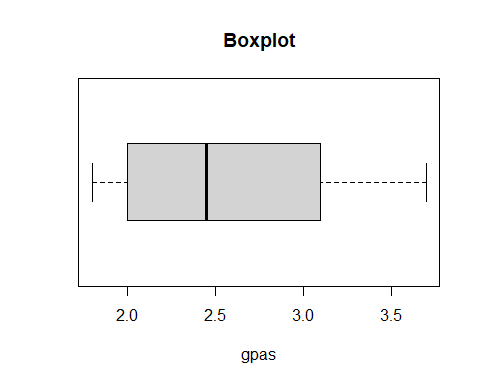
## [1] 2.45

##Part3

summary(gpas)

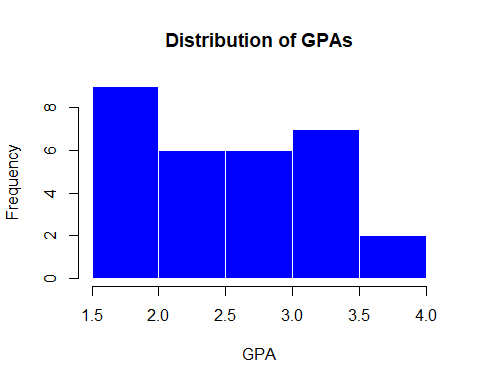
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.80 2.00 2.45 2.59 3.10 3.70

##Create boxplot  
boxplot(gpas, main = "Boxplot",horizontal = T,xlab="gpas")



##Part4

hist(gpas,main = "Distribution of GPAs", xlab = "GPA", ylab = "Frequency",col="blue",border="white")



##Part 4 cont..

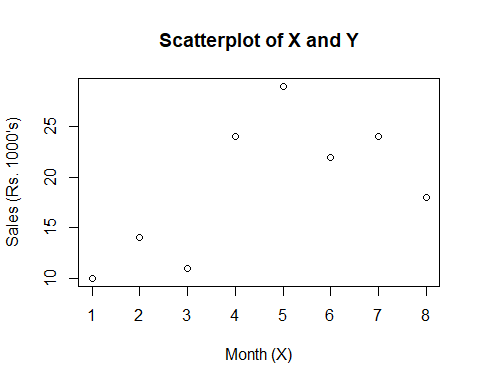
According to the histogram of GPAs for this class, the majority of students have GPAs ranging from 2.0 to 3.0. This implies that class performance may be average or slightly below average, depending on the grading scale and course expectations.

Furthermore, the histogram has a peak around 2.0, indicating that several students received low GPAs. However, there are a few students with GPAs higher than 3.5, which may indicate that some students performed exceptionally well.

#Question 2

##Part 1

X <- c(1, 2, 3, 4, 5, 6, 7, 8)  
Y <- c(10, 14, 11, 24, 29, 22, 24, 18)  
plot(X, Y, xlab = "Month (X)", ylab = "Sales (Rs. 1000's)", main = "Scatterplot of X and Y")



##Part 2

cor(X, Y)

## [1] 0.6115766

##Part 3

cor(X[1:5], Y[1:5])

## [1] 0.8988071

##Part 4

The results of (b) and (c) demonstrate a substantial positive correlation between month and overall sales, and an even stronger correlation for the first five months. This shows that sales tend to rise over time, and that this pattern is most pronounced in the first few months of the year.

#Question3

#Part 1

# Define the probabilities of each outcome  
probs <- c(2/6,4/6)  
# Define the amount won by A for each outcome  
wins <- c(300,-75)  
# Calculate the probability mass function of X  
pmf <- data.frame(X = wins, P\_X = probs)  
print(pmf)

## X P\_X  
## 1 300 0.3333333  
## 2 -75 0.6666667

##Part 2

expected\_win <- sum(wins \* probs)#multiplies the X and Y values   
print(expected\_win)

## [1] 50

##Part 3

variance <- sum(probs \* (wins - expected\_win)^2)#variance  
sd\_X <- sqrt(variance) #standard deviation  
print(sd\_X)

## [1] 176.7767

#Question 4

##Part 1

The binomial distribution with parameters n = 12 and p = 0.03 is the best fit for X. This is due to the fact that X represents the number of successes (infected people) in a fixed number of trials (sample size) with a fixed probability of success (infection rate).

##Part 2

n <- 12  
p <- 0.03  
  
less\_than\_4 <- pbinom(3, size = n, prob = p)  
print(less\_than\_4)

## [1] 0.9996696

##Part 3

mean<-c(n \* p)  
variance <- n \* p \* (1 - p)  
print(mean)

## [1] 0.36

print(variance)

## [1] 0.3492

##Part 4

less\_than\_equal\_4<-pbinom(4, size = n, prob = p)  
greater\_than\_4 <- 1 - less\_than\_equal\_4  
print(greater\_than\_4)

## [1] 1.612637e-05

##Question 5

##Part 1

mean<-47  
sd<-7  
p90<-qnorm(0.9, mean ,sd)  
p90

## [1] 55.97086

##Part 2

p65<-qnorm(0.65, mean ,sd)  
p65

## [1] 49.69724

##Part 3

median<-qnorm(0.5,mean,sd)  
median

## [1] 47

##Part 4

p50<- pnorm(50,mean,sd)  
p\_below\_50<- p50 \* 100  
p\_below\_50

## [1] 66.58824

#Question 6

value <- c(1, 2, 3, 5, 5, 8, 7, 6, 8, 9)  
bootstrap\_samples <- 2000 #number of bootstrap samples to take  
boot <- numeric(bootstrap\_samples)   
for (i in 1:bootstrap\_samples) { #A for loop is then used to iterate through each bootstrap sample  
 boot\_sample <- sample(value, size = length(value), replace = TRUE)  
 boot[i] <- mean(boot\_sample) #The mean of the bootstrap sample is then calculated and stored in the 'boot' vector  
}  
bias <- mean(boot) - mean(value)#subtracting the mean of the original dataset from the mean of the bootstrap means.  
bias

## [1] 0.02655

mean(value)

## [1] 5.4