**Mini Project 5**

**Name :**

**Utkarsh Gandhi ( usg170030 )**

**Dhwani Kaneria( drk170130 )**

**Contribution of team members:**

Utkarsh:

* Wrote R code for question 1 & 2
* Done analysis of the plots and wrote output

Dhwani:

* Wrote R code for question 1 & 2
* Done analysis of the plots and derived results

**Answer 1:**

1. **Comparison of mean body temperature of males and females**

**#R-code:-**

1. bodyData=read.csv("bodytemp-heartrate.csv") # reading csv file into a dataframe

# Extracting male and female data into separate frames

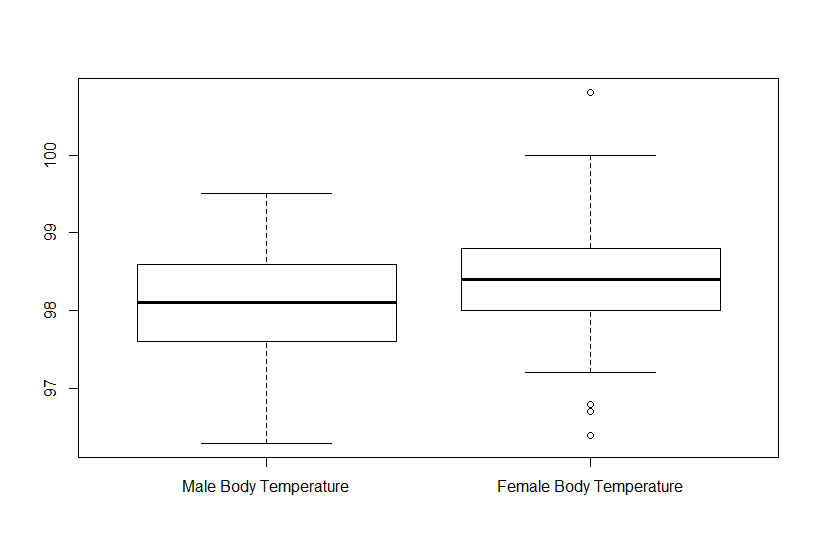
1. male.bodyData = subset(bodyData, gender == 1, select = c(body\_temperature,heart\_rate))
2. female.bodyData = subset(bodyData, gender == 2, select = c(body\_temperature, heart\_rate))

# Extracting male and female body temperatures into separate frames

1. male.Temp = c(male.bodyData[,1])
2. female.Temp = c(female.bodyData[,1])
3. n1 = length(male.Temp)
4. n2 = length(female.Temp)

# Boxplot to compare variances of the data

1. boxplot(male.Temp, female.Temp, names=c("Male Body Temperature","Female Body Temperature"))

****

* **Observed:** Interquartile range of Female Body Temperature is less than that of males. Thus we are going ahead with Satterthwaite approximation.(Not assuming that variances are equal)

**# Performing Hypothesis testing:**

**# H0: means are equal**

**# H1: means are not equal**

**# Using 95% Confidence Interval to test the hypothesis**

**# Two independent samples, Satterthwaite approximation case.**

**# Extracting sample means and sample variances for both samples**

**# Also calculating sample mean difference**

1. mean1 = mean(male.Temp)
2. mean2 = mean(female.Temp)
3. sample.mean.difference = mean1 - mean2
4. var1 = var(male.Temp)
5. var2 = var(female.Temp)

**#making function**

1. calc.satterthwaite=function(var1,var2,n1,n2,sample.mean.difference)
2. {

1. res = c()

**#Standard Error of Sample Mean difference**

1. sample.difference.se = sqrt((var1/n) + (var2/n))

**#Calculating the nu(degrees of freedom for t-distribution) for Satterthwaite Approximation**

1. nu=((sample.difference.se)^4)/(((var1^2)/(n1\*n1\* (n1-1))) + ((var2 ^ 2)/(n2 \* n2 \* (n2-1))))

**# 95% Confidence interval for mean difference**

1. ci.satterthwaite=sample.mean.difference+(c(-1,1)\*qt(0.975 ,df=nu)\*sample.difference.se)
2. return (ci.satterthwaite)
3. }

**# Obtaining 95% CI for difference in means**

1. ci=calc.satterthwaite(var1,var2,n1,n2,sample.mean.difference)

**# Checking if 0 lies in CI**

1. print(0 >= min(ci) && 0 <= max(ci))

**Output:**

[1] TRUE

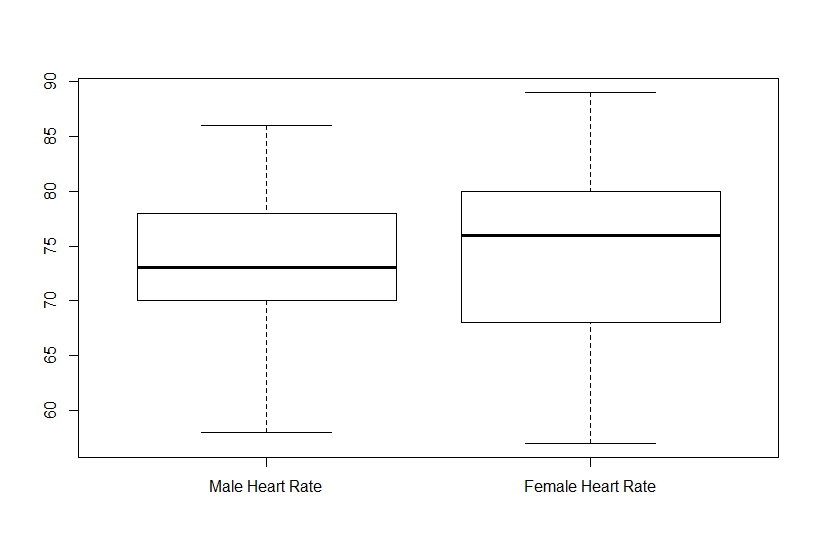
CI = [-1.2891527 , 0.7106912]

* **Observed:** 0 lies in the interval, so there is not sufficient evidence to reject Null hypothesis, H0. Thus we accept H0 and conclude that **mean body temperature of males is the same as mean body temperature of females**.

**(b) Comparing heart rates of males and females**

1. male.heartRate = c(male.bodyData[,2])
2. female.heartRate = c(female.bodyData[,2])
3. boxplot(male.heartRate, female.heartRate, names=c("Male Heart Rate","Female Heart Rate"))

**# plotting box plot**



**Observed**: Interquartile range of Female Heart Rate is much higher than that of males. Thus we are going ahead with Satterthwaite approximation.(Not assuming that variances are equal)

**# Hypothesis testing:**

**# H0: means are equal**

**# H1: means are not equal**

**# Using 95% Confidence Interval to test the hypothesis**

**# Two independent samples, Satterthwaite approximation case.**

**# Extracting sample means and sample variances for both samples**

**# Also calculating sample mean difference**

1. mean1 = mean(male.heartRate)
2. mean2 = mean(female.heartRate)
3. sample.mean.difference = mean1 - mean2
4. var1 = var(male.heartRate)
5. var2 = var(female.heartRate)

**# Obtaining the 95% confidence interval for difference in means**

ci=calc.satterthwaite(var1,var2,n1,n2,sample.mean.difference)

**# Checking if 0 lies in CI**

print(0 >= min(ci) && 0 <= max(ci))

**Output:**

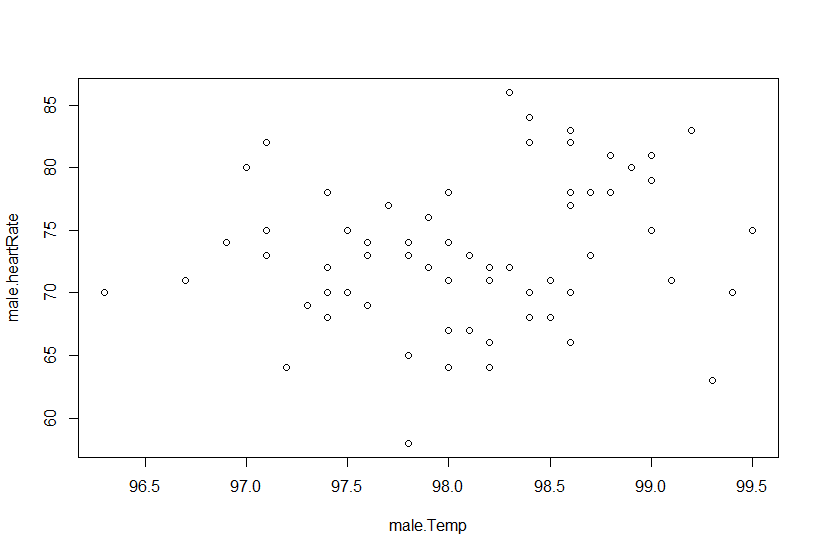
[1] TRUE

CI = [ -10.595227 , 9.025997]

* **Observed:** 0 lies in the interval, so there is not sufficient evidence to reject Null hypothesis, H0.Therefore, **mean heart rate of males is same as the mean heart rate of females**.

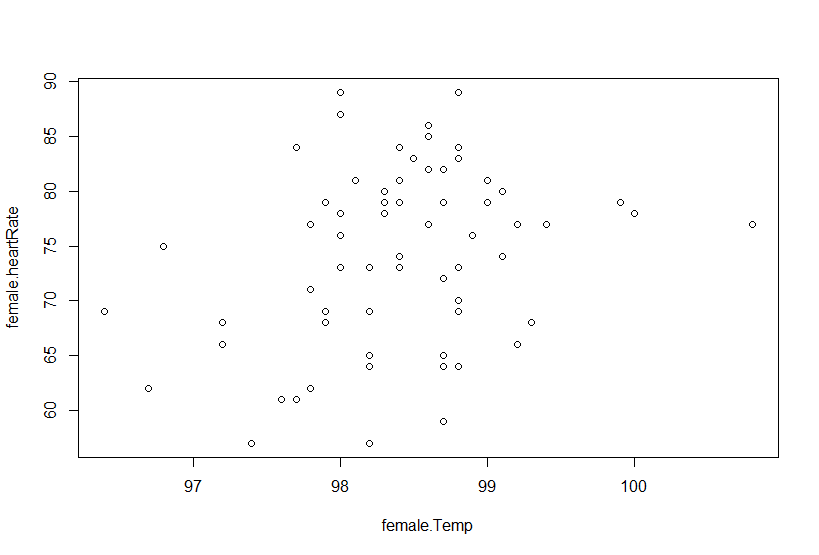
**(c) Determining if there is a linear relationship between body temperature and heart rate for both males and females.**

1. plot(male.Temp,male.heartRate)

**Figure 1 - scatter plot between body temperature and heart rate for male**  - 

2. plot(female.Temp, female.heartRate)

**Figure 2 - scatter plot between body temperature and heart rate for female**



**# Finding correlation coefficients**

corr.coeff.males=(cov(male.Temp,male.heartRate))/(sqrt(var(male.Temp))\*sqrt(var(male.heartRate)))

corr.coeff.females=(cov(female.Temp,female.heartRate))/(sqrt(var(female.Temp))\* sqrt(var(female.heartRate)))

**Output:**

corr.coeff.males = 0.1955894

corr.coeff.females = 0.2869312

* **Observed:** We can see that the Correlation coefficient between body temperature and heart rates for both males and females is very low, and not close to 1, indicating that it is **not a linear relationship**.
* As we can see from both the scatter plots i.e Figure 1 and Figure 2 that there is no linear relationship between temperature and heart rate. So this **relationship does not depends on gender**.

**Answer 2:**

**(a)**

**Computing coverage probabilities**

**#R code:**

1. **# method to find confidence interval for large sample z interval**
2. conf.int<-function(n,lambda,alpha){
3. x<-rexp(n,lambda)
4. len<-length(x)
5. ci<-mean(x)+c(-1,1)\*(qnorm(1-(0.05/2))\*(sd(x)/sqrt(len)))
6. return(ci)
7. }
8. **# method to find confidence interval for bootstrap interval**
9. library(boot)
10. mean.fun <- function(dat, idx) {
11. mean(dat[idx], na.rm = TRUE)
12. }
13. **# generating 5000 z intervals**
14. ci.mat<-replicate(5000,conf.int(5,0.01,0.05))
15. **# calculating coverage probability of large sample z interval**
16. Cov\_Prob1<-mean((1/0.01>=ci.mat[1,])\*(1/0.01<=ci.mat[2,]))\*100
17. **# generating 5000 bootstrap intervals**
18. ci.mat1<-replicate(5000,sort(boot(rexp(5,0.01), mean.fun, R=999)$t)[c(25,975)])
19. **# calculating coverage probability of bootstrap intervals**
20. Cov\_Prob2<-mean((1/0.01>=ci.mat1[1,])\*(1/0.01<=ci.mat1[2,]))\*100
21. cat('Coverage Probability 1 :',Cov\_Prob1,'Coverage Probability 2 :',Cov\_Prob2,' N:',5,'\n')

**OutPut :**

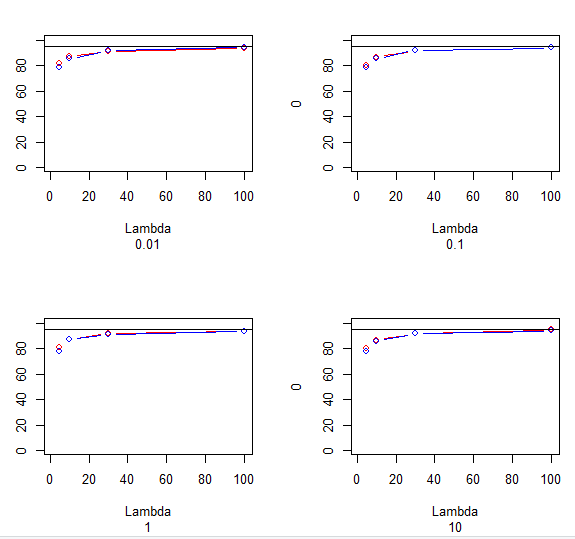
**Coverage Probability 1 : 81.34 Coverage Probability 2 : 78.48 N: 5**

**Narration of code:**

* For the specified values of n =5 and lambda =0.01, to get Monte Carlo simulation we used rexp method to find values of x and after getting the values we found mean and standard deviation and then put in the equations for large-sample z-interval and the other a bootstrap percentile method interval.
* After repeating this process for 5000 times we will be able to get expected coverage probabilities of intervals. By observing output, we can say that for small value of N=5 and Lambda=0.01, coverage probability is not near to 95%,i.e. estimated coverage probability is not close to nominal level of confidence. Even we can say that bootstrap method is not accurate compared to large-sample z-interval in N=5 and Lambda=0.01

**(B)**

1. par(mfrow=c(2,2))
2. **# method to find confidence interval for large sample z interval**
3. conf.int<-function(n,lambda,alpha){
4. x<-rexp(n,lambda)
5. len<-length(x)
6. ci<-mean(x)+c(-1,1)\*(qnorm(1-(0.05/2))\*(sd(x)/sqrt(len)))
7. return(ci)
8. }
9. **# method to find confidence interval for bootstrap interval**
10. library(boot)
11. mean.fun <- function(dat, idx) {
12. mean(dat[idx], na.rm = TRUE)
13. }
14. n<-c(5,10,30,100)  **# taking all values of n**
15. lambda<-c(0.01,0.1,1,10) **# taking all values of lambda**
16. for (j in lambda)
17. {
18. cat('value of Lambda',j,'\n')
19. **# Coverage Probability for specific Lambada with different N value**
20. cov1<-c()
21. cov2<-c()
22. for (i in n)
23. {
24. ci.mat<-replicate(5000,conf.int(i,j,0.05))
25. Cov\_Prob1<-mean((1/j>=ci.mat[1,])\*(1/j<=ci.mat[2,]))\*100
26. ci.mat1<-replicate(5000,sort(boot(rexp(i,j), mean.fun, R=999)$t)[c(25,975)])
28. Cov\_Prob2<-mean((1/j>=ci.mat1[1,])\*(1/j<=ci.mat1[2,]))\*100
29. cat('Coverage Probability 1 :',Cov\_Prob1,'Coverage Probability 2 :',Cov\_Prob2,' N:',i,'\n')
30. cov1<-c(cov1,Cov\_Prob1)
31. cov2<-c(cov2,Cov\_Prob2)
32. }
33. **# Plotting N value Vs coverage probability for Fixed Lambda**
34. plot(0,0,xlim =c(1,100),ylim = c(1,100),type = "n", xlab =c('Lambda',j))
35. lines(n,cov1,col = "red",type = 'b')
36. lines(n,cov2,type = 'b',col = "blue")
37. **# fixed line for 95 % coverage probability interval**
38. abline(h=95)
39. }



**value of Lambda 0.01**

Coverage Probability 1 : 80.54 Coverage Probability 2 : 78.66 N: 5

Coverage Probability 1 : 86.84 Coverage Probability 2 : 85.46 N: 10

Coverage Probability 1 : 92.14 Coverage Probability 2 : 92.28 N: 30

Coverage Probability 1 : 94.22 Coverage Probability 2 : 94.2 N: 100

**value of Lambda 0.1**

Coverage Probability 1 : 81.88 Coverage Probability 2 : 79.14 N: 5

Coverage Probability 1 : 86.42 Coverage Probability 2 : 86.18 N: 10

Coverage Probability 1 : 91.98 Coverage Probability 2 : 91.78 N: 30

Coverage Probability 1 : 93.7 Coverage Probability 2 : 93.94 N: 100

**value of Lambda 1**

Coverage Probability 1 : 81.36 Coverage Probability 2 : 78.7 N: 5

Coverage Probability 1 : 86.28 Coverage Probability 2 : 86.84 N: 10

Coverage Probability 1 : 91.76 Coverage Probability 2 : 91.58 N: 30

Coverage Probability 1 : 94.34 Coverage Probability 2 : 93.54 N: 100

**value of Lambda 10**

Coverage Probability 1 : 80.64 Coverage Probability 2 : 78.48 N: 5

Coverage Probability 1 : 86.56 Coverage Probability 2 : 85.9 N: 10

Coverage Probability 1 : 91.8 Coverage Probability 2 : 91.92 N: 30

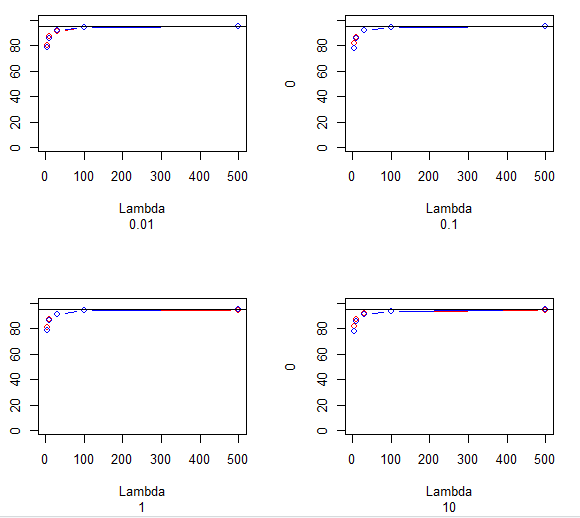
Coverage Probability 1 : 93.6 Coverage Probability 2 : 93.86 N: 100

**Narration of code:**

For different values, same code as 2a is run. By assigning values to different vectors and running them through for loop.

**(C)**

* When N is small Both interval don’t give correct coverage accuracy of interval. As n increasing, interval becomes more accurate for both intervals.
* For Large sample interval, n is needed to be more than 100 to get exact accuracy as 95%.
* Also for Bootstrap interval, n is needed to be more than 100 to get accuracy as 95%.
* This answer is not depends on value of lambda value, for every value of lambda, large n size remains same.
* By taking, n<-c(5,10,30,100,500) , we got below result. By observing plot, we can see that for n value more than 100, coverage probability is almost on 95% line for every lambda. this value does not even depends on the values of Lambda.
* Comparing both the intervals, Large sample seems to be more accurate than bootstrap interval for small value of n (i.e. n=5). For large value of N, Both intervals ouputs almost same result. So we can choose either of them, for large value of N.



**(D)**

* No, our conclusion does not depend on fixed value of lambda, because we have consider every possible values of lambda.
* Observing above Plot, We can say that regardless of value of Lambda, both intervals estimate coverage probability close to nominal level of confidence as 95% for value of N more than 100. But for N=5, Large sample interval gives more accurate coverage probability than bootstrap interval.