CS 6313.001
Statistical Methods for Data Science

By
Prof. Min Chen

MINI PROJECT-5
DUO GROUP-41

1. Amit Kumar AXK210047
2. Vedant Paresh Shah VXS200021

Contribution of each group member: Both worked together and finished the questions as instructed. First went through all the details required, followed the class Note and textbook, practiced the R-concept, then wrote down the scripts. Both of us worked efficiently to complete the required project and finished it on time.

Question-1: Consider the data stored in bodytemp-heartrate.csv on eLearning, containing measurements of body temperature and heart rate for 65 male (gender = 1) and 65 female (gender = 2) subjects.

Initial Step:

Section: 1

Initial Step: We uploaded the csv file bodytemp-heartrate.csv on G-Drive and from there read the csv file. Once the data is read into bodyTempHeartRate, we separated the contents of male data set into maleDataSet and female data set into femaleDataSet to perform further testing and plotting.

Section: 2
R-Code:

```
> library(boot) # import the boot library
> id <- "1J1jZU-jH7NN5CHanq-w6GpdzLv3KICtU" # G-Drive file ID
> bodyTempHeartRate <- read.csv(
  +         sprintf("https://docs.google.com/uc?id=%s&export=download", id), header=TRUE)
> summary(bodyTempHeartRate)
```

Summary of the data from the csv file:

```
body_temperature       gender        heart_rate
Min.   : 96.30   Min.   :1.0   Min.   :57.00
1st Qu.: 97.80   1st Qu.:1.0   1st Qu.:69.00
Median : 98.30   Median :1.5   Median :74.00
Mean   : 98.25   Mean   :1.5   Mean   :73.76
3rd Qu.: 98.70   3rd Qu.:2.0   3rd Qu.:79.00
Max.   :100.80   Max.   :2.0   Max.   :89.00
```

(a) Do males and females differ in mean body temperature? Answer this question by performing an appropriate analysis of the data, including an exploratory analysis.

Section: 1

Using the maleDataset and femaleDataset from the Initial step, we segregated the male body temperature and female body temperature into maleBodyTemperature and femaleBodyTemperature respectively. Used these data to find the summary and drew boxplots using R (R-code and output is given in section 2 below). From the summary we found the mean of body temperature for Male is 98.1 and for female is 98.39 and from boxplot we can observe that the mean temperature of females is higher than of males as well. Also the Q1, Median and Q3 for female is higher than the male. It concludes that the Male and female differ in mean body temperature. To further strengthen our analysis, we drew QQ-plot and performed t-Distribution to find the confidence interval. From the Q-Q plots, it can be considered the distributions of the body temperature of males and females is approximately normal.

Let's assume "m" denote the body temperatures of males and "f" denote the body temperatures of females. So, the sample mean $\bar{m}$ estimates the population mean μ_m and the sample mean $\bar{f}$ estimates the population mean μ_f .

Null hypothesis Ho : $\bar{m} - \bar{f} = 0$
Alternate Hypothesis $H1$: $\bar{m} - \bar{f} \neq 0$

Here we will have to treat sample as independent with unequal variances from an approximately normal distribution (concluded from QQ-plot below in section: 2). We will have to use t-distribution with Satterthwaite's approximation to get the confidence interval.

R Syntax to get t-distribution for getting the confidence interval is: t.test(maleBodyTemperature, femaleBodyTemperature, alternative = 'two.sided', var.equal = F). Here, alternative parameter is a character string specifying the alternative hypothesis and var.equal a logical variable indicating whether to treat the two variances as being equal. 95% confidence interval returned from the t.test is (-0.53964856 -0.03881298) and the p-value we got is 0.02394. From the output of t.test, we observed that the p-value doesn't fall within the CI, as a result we rejected the null hypothesis and concluded that mean of the body temperature of male and female are not equal. The outcome of t-test also indicated that the mean of x i.e., male is 98.10462 and mean of y i.e. female is 98.39385. Using all these arguments and data we can conclude that males and females differ in mean body temperature.

Section: 2

R-Code:

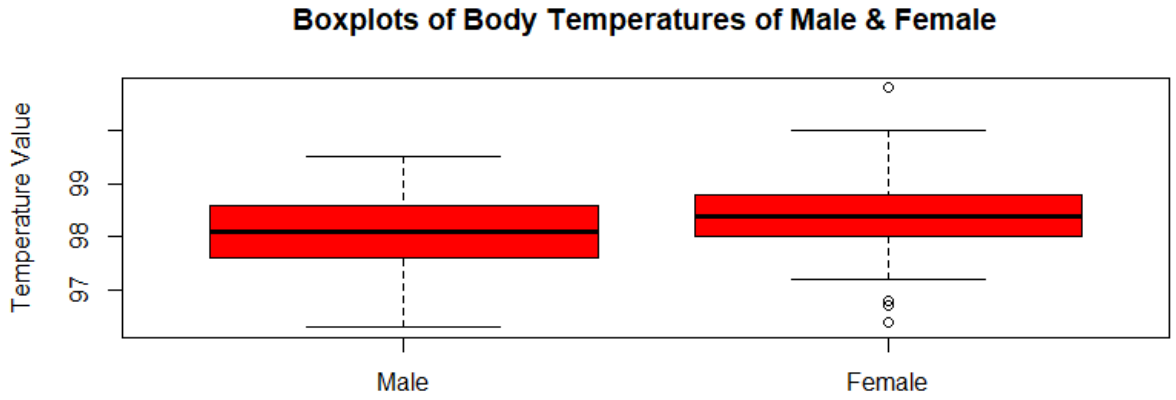
```
> maleDataSet = subset(bodyTempHeartRate, bodyTempHeartRate$gender == 1)

> femaleDataSet = subset(bodyTempHeartRate, bodyTempHeartRate$gender == 2)
> maleBodyTemperature <- as.numeric(maleDataSet$body_temperature)
> summary(maleBodyTemperature)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
96.3	97.6	98.1	98.1	98.6	99.5

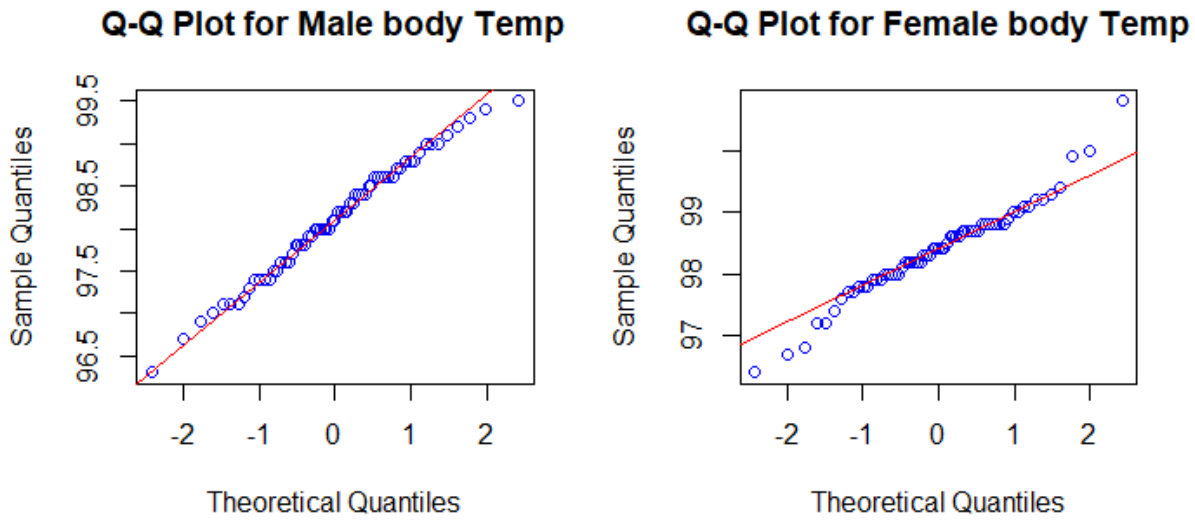
```
> femaleBodyTemperature <- as.numeric(femaleDataSet$body_temperature)
> summary(femaleBodyTemperature)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  96.40   98.00   98.40   98.39   98.80  100.80
```

```
> boxplot(maleBodyTemperature, femaleBodyTemperature,
+         main = "Boxplots of Body Temperatures of Male & Female",
+         col="red", names = c('Male', 'Female'), ylab = "Temperature Value")
```




```
# Drawing Q-Q plots for the body temperature values
> par(mfrow=c(1,2))
> qqnorm(maleBodyTemperature, col="blue", main = 'Q-Q Plot for Male body Temp')
> qqline(maleBodyTemperature, col="red")
> qqnorm(femaleBodyTemperature, col="blue", main = 'Q-Q Plot for Female body Temp')
> qqline(femaleBodyTemperature, col="red")
```




```
# Confidence interval using t,test function for the body temperature values
> t.test(maleBodyTemperature, femaleBodyTemperature, alternative =
+         'two.sided', var.equal = F)
```

```
        Welch Two Sample t-test

data:  maleBodyTemperature and femaleBodyTemperature
t = -2.2854, df = 127.51, p-value = 0.02394
alternative hypothesis: true difference in means is not equal to
 0
95 percent confidence interval:
 -0.53964856 -0.03881298
sample estimates:
mean of x mean of y
 98.10462  98.39385
```

(b) Do males and females differ in mean heart rate? Answer this question by performing an appropriate analysis of the data, including an exploratory analysis.

Section: 1

Using the maleDataset and femaleDataset from the Initial step, we segregated the male heart rate and female heart rate into maleHeartRate and femaleHeartRate respectively. Used these data to find the summary and drew boxplots using R (R-code and output is given in section 2 below). From the summary we found the mean of heart rate for Male is 73.37 and for female is 74.15 and from boxplot we can observe that the Q1 for females is less than Q1 for males, but this is not the case for mean, median and Q3 as those values are higher for females than the males. The values in females seem more stretched out so variability seems to be more. we also drew QQ-plot and performed t-Distribution to find the confidence interval. From the Q-Q plots, it can be considered the distributions of the heart rate of males and females is approximately normal.

Let's assume "m" denote the heart rate of males and "f" denote the heart rate of females. So, the sample mean $\bar{m}$ estimates the population mean μ_m and the sample mean $\bar{f}$ estimates the population mean μ_f .

Null hypothesis Ho : $\bar{m} - \bar{f} = 0$
Alternate Hypothesis H1: $\bar{m} - \bar{f} \neq 0$

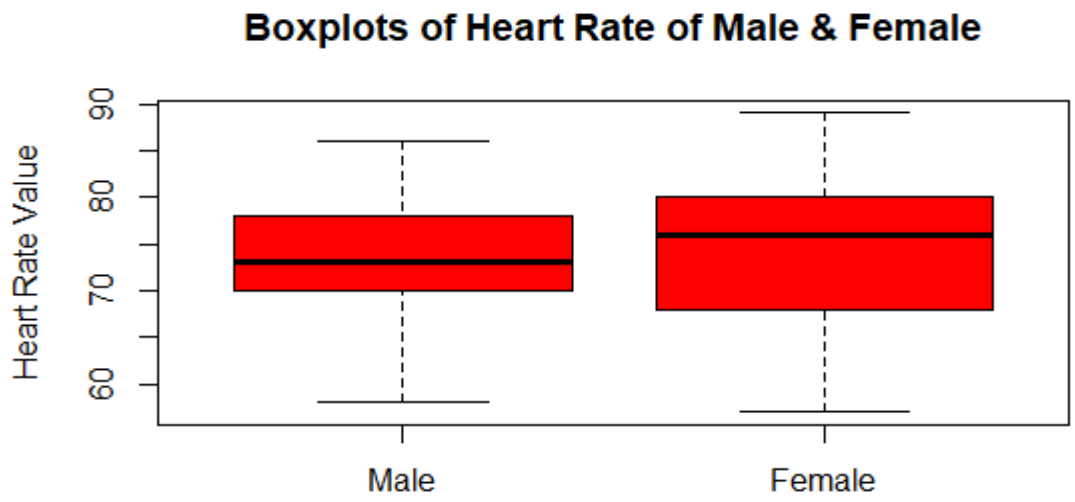
Here we will have to treat sample as independent with unequal variances from an approximately normal distribution (concluded from QQ-plot below in section: 2). We will have to use t-distribution with Satterthwaite's approximation to get the confidence interval.

R Syntax to get t-distribution for getting the confidence interval is: t.test (maleHeartRate, femaleHeartRate, alternative = 'two. sided', var. equal = F). Here, alternative parameter is a character string specifying the alternative hypothesis and var. equal a logical variable indicating whether to treat the two variances as being equal. 95% confidence interval returned from the t.test is (-3.243732 1.674501) and the p-value we got is 0.5287. From the output of t.test, we observed that the p-value fall within the CI, as a result we accepted the null hypothesis and concluded that the mean value of heart rate of females and males are equal.

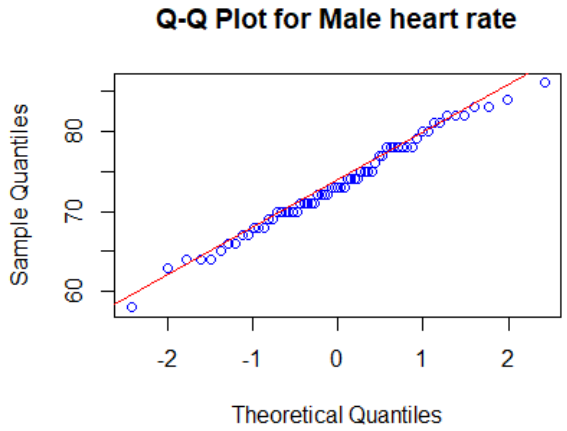
Section: 2

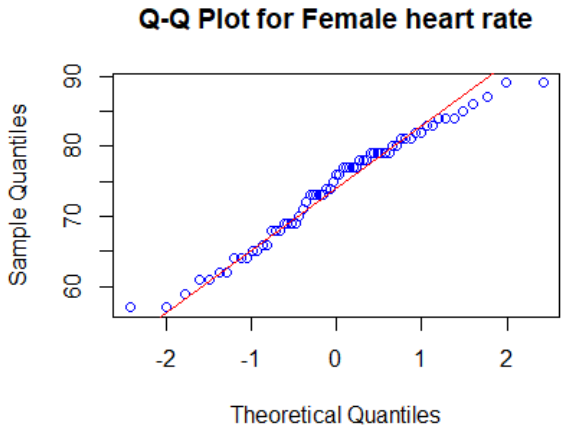
R Code:

```
> maleHeartRate <- as.numeric(maleDataSet$heart_rate)
> summary(maleHeartRate)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  58.00   70.00   73.00   73.37   78.00   86.00
> femaleHeartRate <- as.numeric(femaleDataSet$heart_rate)
> summary(femaleHeartRate)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  57.00   68.00   76.00   74.15   80.00   89.00
> boxplot(maleHeartRate, femaleHeartRate,
+     main = "Boxplots of Heart Rate of Male & Female", col="red", names = c('Male', 'Female'), ylab
+     = "Heart Rate Value")
```




```
# Drawing Q-Q plots for the heart rate values
> par(mfrow=c(1,2))
> qqnorm(maleHeartRate, col="blue", main = 'Q-Q Plot for Male heart rate')
> qqline(maleHeartRate, col="red")
> qqnorm(femaleHeartRate, col="blue", main = 'Q-Q Plot for Female heart rate')
> qqline(femaleHeartRate, col="red")
```





Confidence interval using t,test function for the heart rate values

> t.test(maleHeartRate, femaleHeartRate, alternative = 'two.sided', var.equal = F)

```
        Welch Two Sample t-test

data:  maleHeartRate and femaleHeartRate
t = -0.63191, df = 116.7, p-value = 0.5287
alternative hypothesis: true difference in means is not equal to
 0
95 percent confidence interval:
 -3.243732  1.674501
sample estimates:
mean of x mean of y
 73.36923  74.15385
```

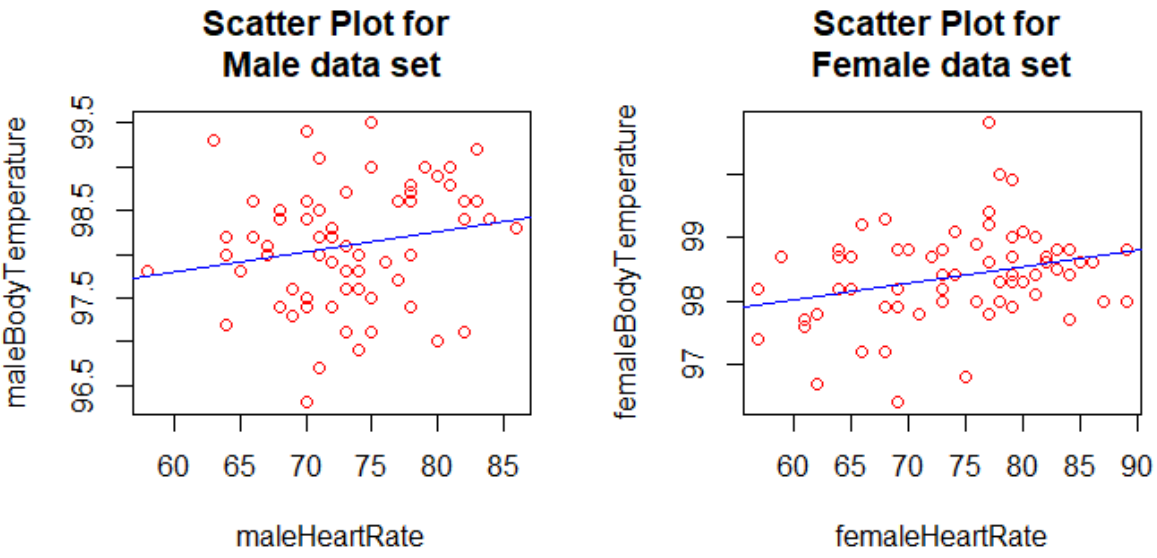
(c) Is there a linear relationship between body temperature and heart rate? Does this relationship depend on gender? Answer these questions by performing an appropriate analysis of the data, including an exploratory analysis.

Section: 1

To find the relationship between body temperature and heart rate, we will use scatter plot and on top of that a regression line will be drawn which will reflects the linear relationship between them. Also to find the correlation between Temperature and Heart rate for male and female dataset, we used cor() function. Which returned the value of correlation between body temperature and heart rate for male is 0.1955894 and for female is 0.2869312. Larger the value, stronger is the correlation. From the cor() function result we can conclude that the correlation of body temperature and heart rate for female is bit stronger than male. Also from the scatter plot(given below in section: 2) we can conclude that the relation between body temperature and heart rate is positive. As we observed the correlation coefficient for body temperature and heart rate for female is bit higher than that of male, hence it is dependent on gender.

Section: 2

```
> cor(maleBodyTemperature, maleHeartRate)
[1] 0.1955894

> cor(femaleBodyTemperature, femaleHeartRate)
[1] 0.2869312

# Drawing the scatter plots for the body temperature and heart rate values for males and females
> par(mfrow=c(1,2))
> plot(maleHeartRate, maleBodyTemperature, pch=1, main='Scatter Plot for
+         Male data set', col='red')
> abline(lm(maleBodyTemperature~maleHeartRate), col='blue')
> plot(femaleHeartRate, femaleBodyTemperature, pch=1, main='Scatter Plot for
+         Female data set', col='red')
> abline(lm(femaleBodyTemperature~femaleHeartRate), col='blue')
```



Question-2: The goal of this exercise to see how large n should be for the large-sample and the (parametric) bootstrap percentile method confidence intervals for the mean of an exponential population to be accurate. To be specific, let X1, . . . , Xn represent a random sample from an exponential (λ) distribution. Note that this distribution is skewed and its mean is $\mu = 1/\lambda$. We can construct two confidence intervals for μ — one the large-sample z-interval (interval 1) and the other a (parametric) bootstrap percentile method interval (interval 2). We would like to investigate their accuracy, i.e., how close their estimated coverage probabilities are to the assumed nominal level of confidence, for various combinations of (n, λ). This investigation will focus on $1 - \alpha = 0.95$, λ = 0.01, 0.1, 1, 10 and n = 5, 10, 30, 100. Thus, we have a total of $4 * 4 = 16$ combinations of (n, λ) to investigate.

Answer: Here we will first create functions to calculate the Confidence interval for two datasets first one is a large sample with size n and for the second dataset we create it by using the Parametric Bootstrap Method. Parametric Bootstrap Creates Dataset of n size using the Original Sample and by resampling it using Mean square method. Both datasets have an exponential distribution.

We have been given that Confidence Interval needs to be calculated for 95%. Also, Lambda Values we need to calculate have been given λ = 0.01, 0.1, 1, 10 and Nominal Level = 5, 10, 30, 100.

(a) For a given setting, compute Monte Carlo estimates of coverage probabilities of the two intervals by simulating appropriate data, using them to construct the two confidence intervals, and repeating the process 5000 times.

Section-1:

To Simulate the monte Carlo Estimates for Coverage Probability we have created 2 functions. To calculate the Coverage probability for size n large Samples, we used Z Interval to calculate the Confidence interval we calculate this in ZCI(n, lambda) function. And for Parametric Bootstrap we first need to resample from the size n large sample by using mean.star(n, lambda) function then we calculate its confidence interval in ParBootCI(n, lambda). The mean of the Resample's is taken 1000 times.

Here we will take the Confidence Interval for both the Functions 5000 times for each value of lambda and n.

Here we will take value of Lambda = 0.01 and n = 5.

For Z-Interval = 0.8008

For Parametric Bootstrap Interval = 0.903

Section-2:

(Z-Interval)

```
ZCI = function(n,lambda)
{
 alpha = 0.05
 Sample = rexp(n,lambda)
 LB = mean(Sample)+(-1)*qnorm(1-(alpha/2))*(sd(Sample)/sqrt(n))
 UB = mean(Sample)+(1)*qnorm(1-(alpha/2))*(sd(Sample)/sqrt(n))
 M = 1/lambda
 if(LB<M & UB>M)
 {
  return(1)
 }
 else
 {
  return(0)
 }
}
ZCoverProb = function(n,lambda,nsim)
{
 CIMatrix = replicate(nsim,ZCI(n,lambda))
 CovProbOrg = CIMatrix[which(CIMatrix==1)]
 return(length(CovProbOrg)/nsim)
}
ZMatrix = matrix(c(ZCoverProb(5,0.01,5000), ZCoverProb(5,0.1,5000), ZCoverProb(5,1,5000),
ZCoverProb(5,10,5000), ZCoverProb(10,0.01,5000), ZCoverProb(10,0.1,5000),
ZCoverProb(10,1,5000), ZCoverProb(10,10,5000), ZCoverPro30,0.01,5000),
ZCoverProbb(30,0.1,5000), ZCoverProb(30,1,5000), ZCoverProb(30,10,5000),
ZCoverProb(100,0.01,5000), ZCoverProb(100,0.1,5000), ZCoverProb(100,1,5000),
ZCoverProb(100,10,5000)), nrow=4, ncol=4)
```

```
print("Z CI Coverage probability")
ZMatrix[1,1]
ZMatrix[1,2]
ZMatrix[1,3]
ZMatrix[1,4]

ZMatrix[2,1]
ZMatrix[2,2]
ZMatrix[2,3]
ZMatrix[2,4]

ZMatrix[3,1]
ZMatrix[3,2]
ZMatrix[3,3]
ZMatrix[3,4]

ZMatrix[4,1]
ZMatrix[4,2]
ZMatrix[4,3]
ZMatrix[4,4]
```

Output:

[1] "Z CI Coverage probability"

[1] 0.8008

[1] 0.8736

[1] 0.9202

[1] 0.9376

[1] 0.8156

[1] 0.8746

[1] 0.9176

[1] 0.94

[1] 0.8114

[1] 0.8628

[1] 0.9188

[1] 0.9462

[1] 0.8112

[1] 0.8704

[1] 0.924

[1] 0.9408

(**Parametric Bootstrap Interval**)

```
mean.star = function(n,lambda)
{
  Resample = rexp(n,lambda)
  ResampleMean = mean(Resample)
  return(ResampleMean)
}
ParBootCI = function(n,lambda)
{
  x = rexp(n,lambda)
  nsamples = 1000
  M = 1/lambda
  lambda1 = 1/mean(x)
  BootDist = replicate(nsamples,mean.star(n,lambda1))
  Bound = sort(BootDist)[c(25,975)]
  if(Bound[1]<M & Bound[2]>M)
  {
    return(1)
  }
  else
  {
    return(0)
  }
}
BCoverProb = function(n,lambda,nsim)
{ CIMatrix = replicate(nsim,ParBootCI(n,lambda))
  CovProbBoot = CIMatrix[which(CIMatrix==1)]
  return(length(CovProbBoot)/nsim) }
```

```
BMatrix= matrix(c(BCoverProb(5,0.01,5000), BCoverProb(5,0.1,5000), BCoverProb(5,1,5000),
BCoverProb(5,10,5000), BCoverProb(10,0.01,5000), BCoverProb(10,0.1,5000),
BCoverProb(10,1,5000), BCoverProb(10,10,5000), BCoverProb(30,0.01,5000),
BCoverProb(30,0.1,5000), BCoverProb(30,1,5000), BCoverProb(30,10,5000),
BCoverProb(100,0.01,5000), BCoverProb(100,0.1,5000), BCoverProb(100,1,5000),
BCoverProb(100,10,5000)), nrow=4, ncol=4)

print("Parametric Bootstrap CI Coverage probability")

BMatrix[1,1]

BMatrix[1,2]

BMatrix[1,3]

BMatrix[1,4]

BMatrix[2,1]

BMatrix[2,2]

BMatrix[2,3]

BMatrix[2,4]

BMatrix[3,1]

BMatrix[3,2]

BMatrix[3,3]

BMatrix[3,4]

BMatrix[4,1]

BMatrix[4,2]

BMatrix[4,3]

BMatrix[4,4]
```

Output:

[1] "Parametric Bootstrap CI Coverage probability"

[1] 0.903

[1] 0.9156

[1] 0.9392

[1] 0.9404

[1] 0.8958

[1] 0.921

[1] 0.9426

[1] 0.9398

[1] 0.891

[1] 0.9254

[1] 0.9358

[1] 0.9478

[1] 0.8974

[1] 0.9192

[1] 0.9334

[1] 0.9512

(b) Repeat (a) for the remaining combinations of (n, λ). Present an appropriate summary of the results.

Section-1:

Z-Interval:

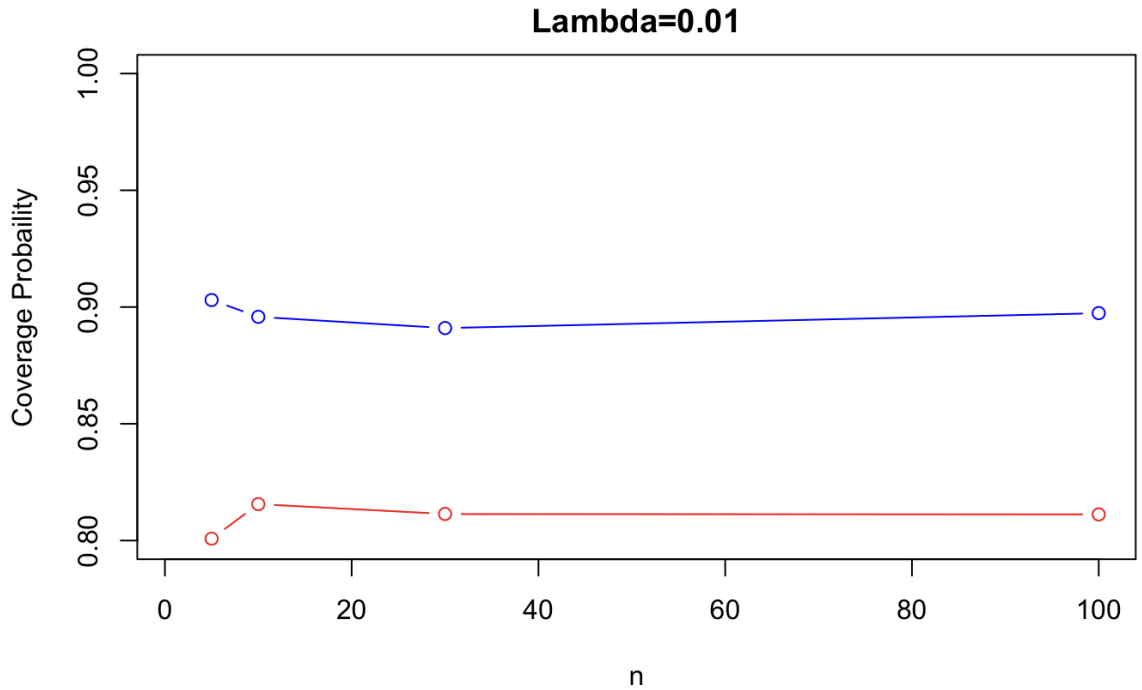
Z-Proportions	Lambda=0.01	Lambda=0.1	Lambda=1	Lambda=10
N=5	0.8008	0.8736	0.9202	0.9376
N=10	0.8156	0.8746	0.9176	0.94
N=30	0.8114	0.8628	0.9188	0.9462
N=100	0.8112	0.8704	0.924	0.9408

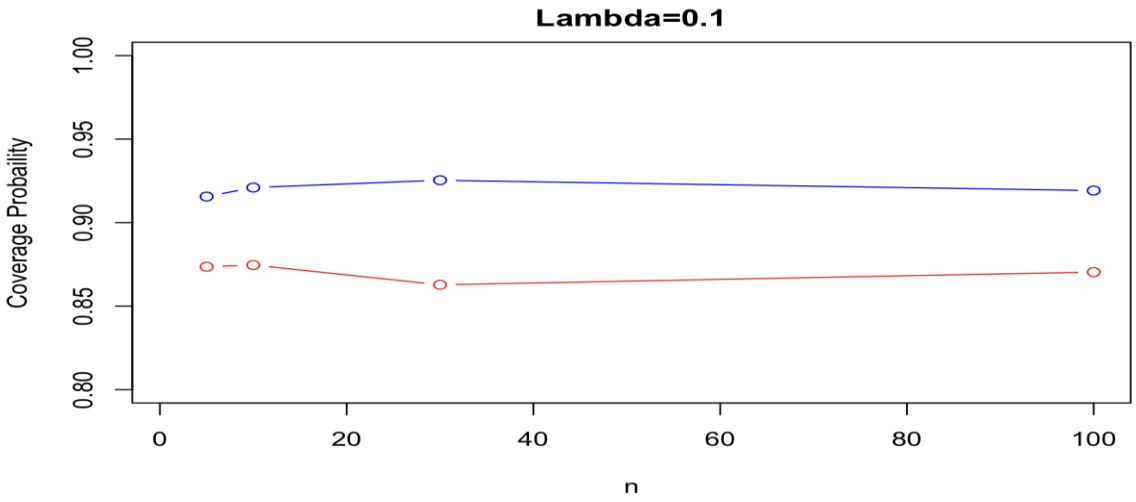
Parametric Bootstrap Interval:

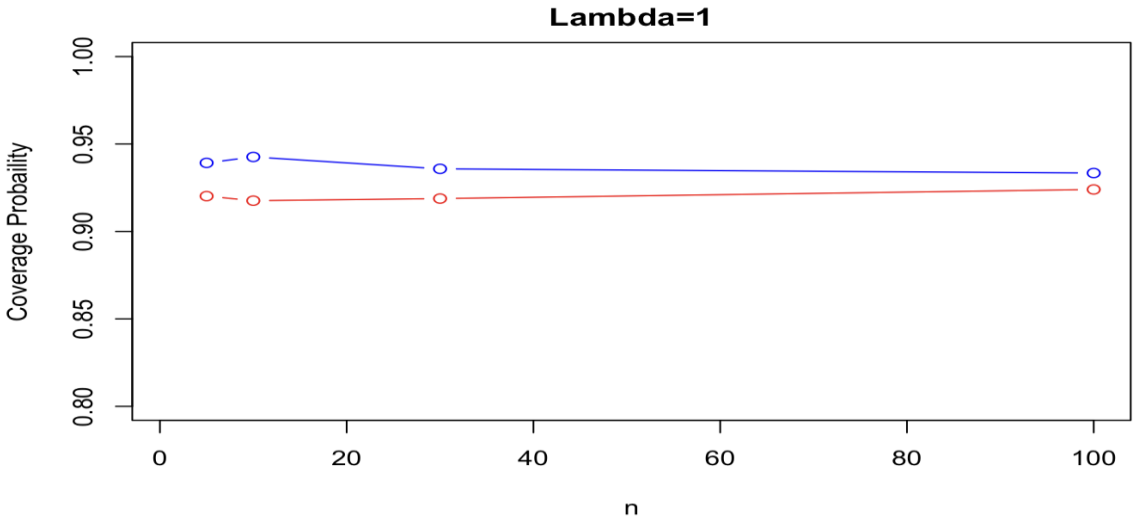
PB-Proportions	Lambda=0.01	Lambda=0.1	Lambda=1	Lambda=10
N=5	0.903	0.9156	0.9392	0.9404
N=10	0.8958	0.921	0.9426	0.9398
N=30	0.891	0.9254	0.9358	0.9478
N=100	0.8974	0.9192	0.9334	0.9512

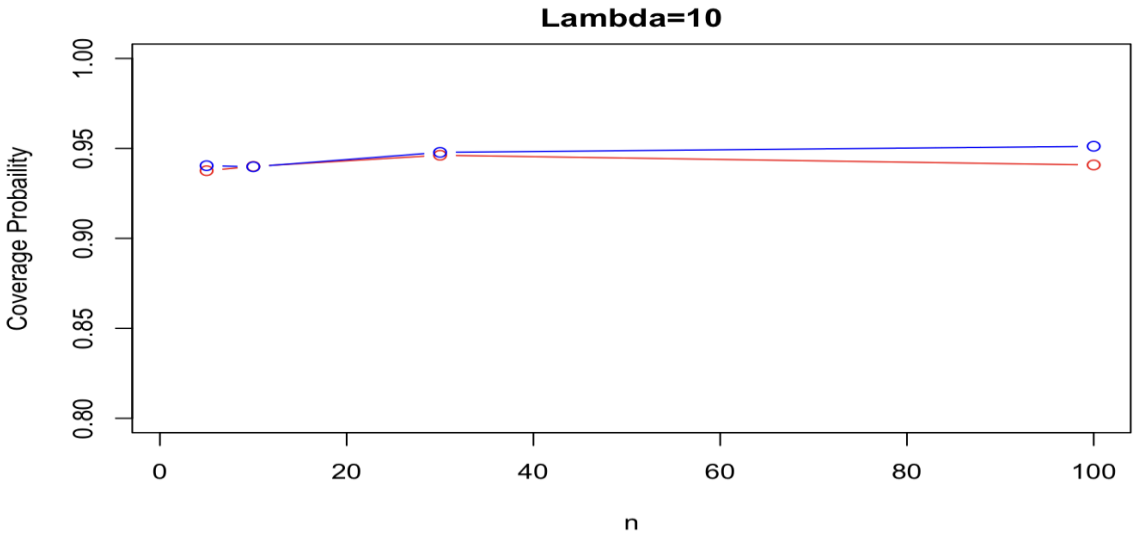
Graph For Lambda Values:

Red Lines - Z Intervals Blue Lines – Parametric Bootstrap Intervals

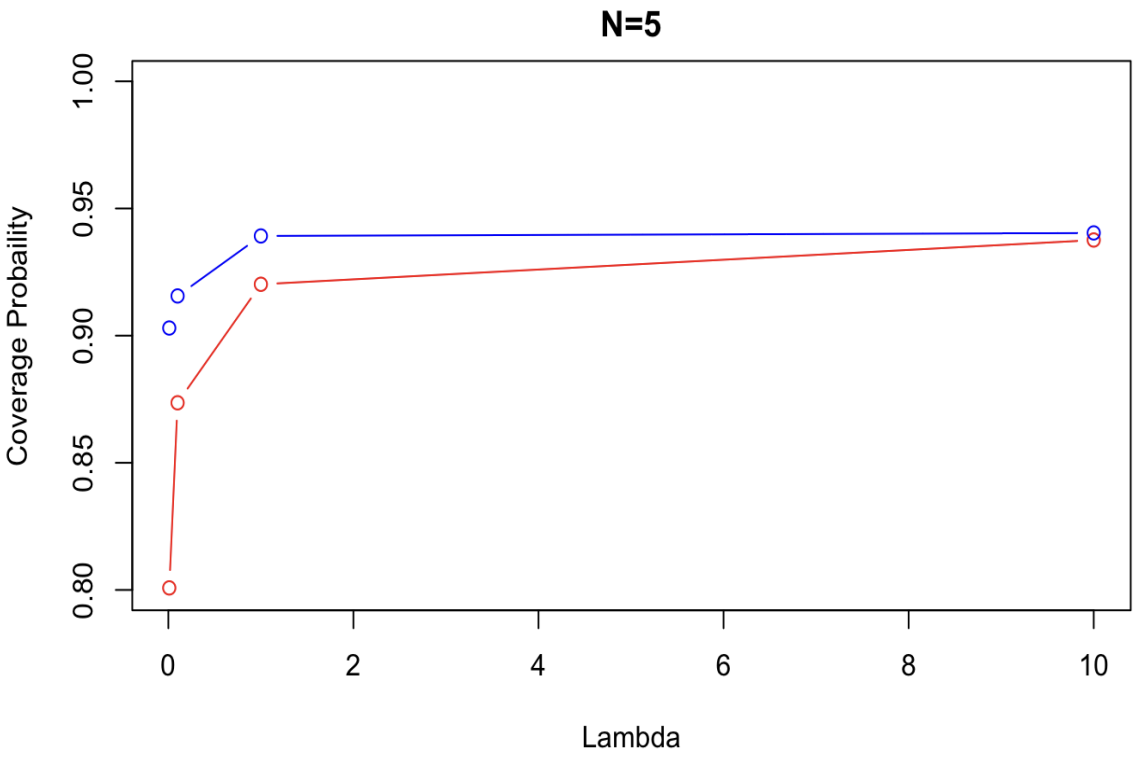


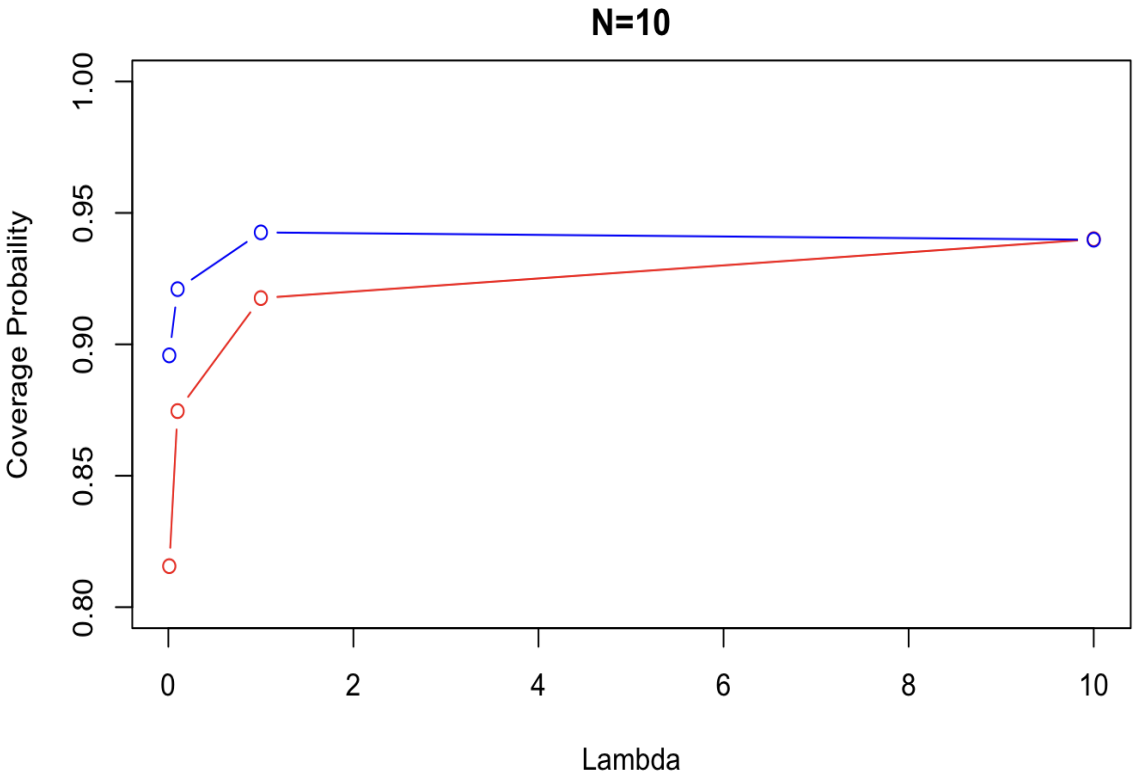

Lambda=0.1
Coverage Probaility
0.80
0.85
0.90
0.95
1.00
0
20
40
60
80
100
n

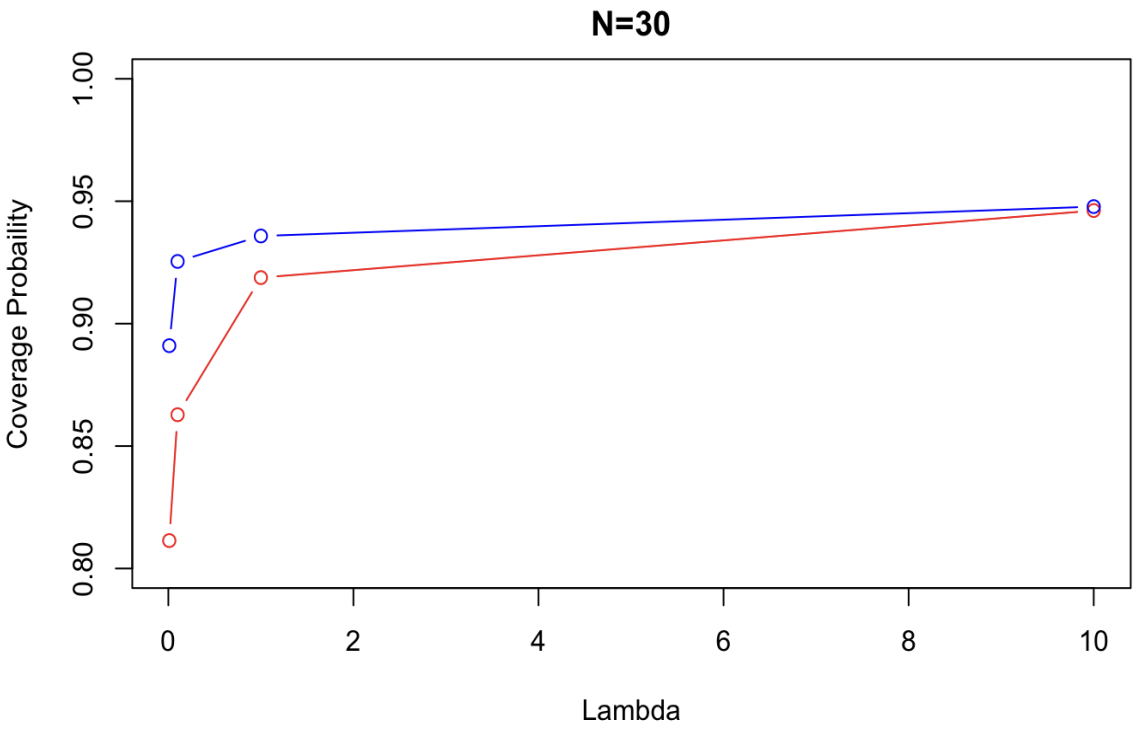

Lambda=1
Coverage Probaility
0.80
0.85
0.90
0.95
1.00
0
20
40
60
80
100
n

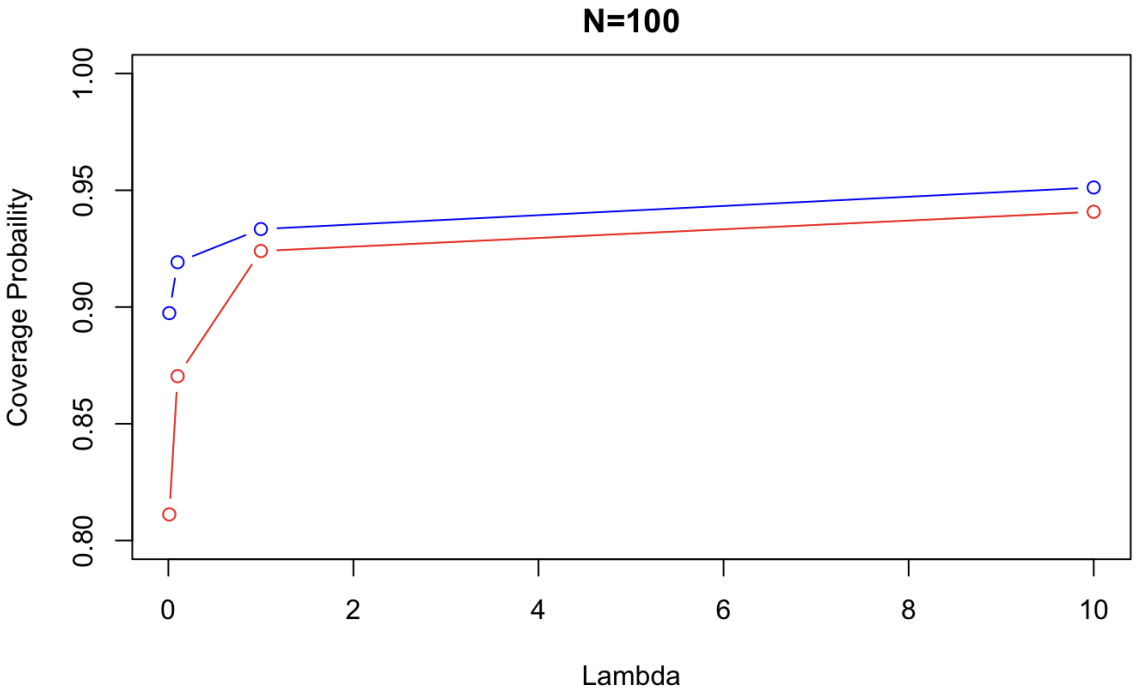

Lambda=10
Coverage Probaility
0.80
0.85
0.90
0.95
1.00
0
20
40
60
80
100
n

Graph For N Values:






N=30
Coverage Probaility
1.00
0.95
0.90
0.85
0.80
0
2
4
6
8
10
Lambda


N=100
Coverage Probaility
1.00
0.95
0.90
0.85
0.80
0
2
4
6
8
10
Lambda

Section-2:

```
plot(c(5,10,30,100),ZMatrix[,1],main="Lambda=0.01",xlab='n',ylab='Coverage Probaility',
col='red',type='b',xlim=c(1,100),ylim=c(0.8,1))

lines(c(5,10,30,100),BMatrix[,1],col='blue',type='b')

plot(c(5,10,30,100),ZMatrix[,2],main="Lambda=0.1",xlab='n',ylab='Coverage Probaility',
col='red',type='b',xlim=c(1,100),ylim=c(0.8,1))

lines(c(5,10,30,100),BMatrix[,2],col='blue',type='b')

plot(c(5,10,30,100),ZMatrix[,3],main="Lambda=1",xlab='n',ylab='Coverage Probaility',
col='red',type='b',xlim=c(1,100),ylim=c(0.8,1))

lines(c(5,10,30,100),BMatrix[,3],col='blue',type='b')

plot(c(5,10,30,100),ZMatrix[,4],main="Lambda=10",xlab='n',ylab='Coverage Probaility',
col='red',type='b',xlim=c(1,100),ylim=c(0.8,1))

lines(c(5,10,30,100),BMatrix[,4],col='blue',type='b')

plot(c(0.01,0.1,1,10),ZMatrix[1,],main="N=5",xlab='Lambda',ylab='Coverage Probaility',
col='red',type='b',xlim=c(0.01,10),ylim=c(0.8,1))

lines(c(0.01,0.1,1,10),BMatrix[1,],col='blue',type='b')

plot(c(0.01,0.1,1,10),ZMatrix[2,],main="N=10",xlab='Lambda',ylab='Coverage Probaility',
col='red',type='b',xlim=c(0.01,10),ylim=c(0.8,1))

lines(c(0.01,0.1,1,10),BMatrix[2,],col='blue',type='b')

plot(c(0.01,0.1,1,10),ZMatrix[3,],main="N=30",xlab='Lambda',ylab='Coverage Probaility',
col='red',type='b',xlim=c(0.01,10),ylim=c(0.8,1))

lines(c(0.01,0.1,1,10),BMatrix[3,],col='blue',type='b')

plot(c(0.01,0.1,1,10),ZMatrix[4,],main="N=100",xlab='Lambda',ylab='Coverage Probaility',
col='red',type='b',xlim=c(0.01,10),ylim=c(0.8,1))

lines(c(0.01,0.1,1,10),BMatrix[4,],col='blue',type='b')
```

(c) Interpret all the results. Be sure to answer the following questions: In case of the large-sample interval, how large n is needed for the interval to be accurate? Likewise, in case of the bootstrap interval, how large n is needed for the interval to be accurate? Do these answers depend on λ? Can we say that one method is more accurate than the other? Which interval would you recommend? Provide justification for all your conclusions.

Section-1:

For the given value of n the Coverage probability does not change that much respect regarding value of Lambda. The given value of Lambda, the coverage probability increases as n increases. The Coverage probability for n=100. Is closest to the 95% confidence interval irrespective of Lambda.

For the large sample interval, how large does n have to be for the interval to be accurate?

The large sample interval gives coverage probabilities close to our confidence level of 95% for n=100.

For the bootstrap interval, how large does n have to be for the interval to be accurate?

The bootstrap interval gives coverage probabilities close to 0.95 for for n=100.

Do these answers depend on lambda?

No, the answers to the above questions do not depend on lambda.

Can we say that one method is more accurate than the other? Which interval would you recommend?

The bootstrap interval appears to be working better for smaller n. But for large n both are equally good irrespective of Lambda.

(d) Do your conclusions in (c) depend on the specific values of λ that were fixed in advance? Explain.

No, the conclusions in (c) do not depend on lambda.

As seen in the tables above in (b). We can observe that the value of Lambda does not affect the Confidence Interval. But the value of n doe does affect the value Confidence Interval irrespective of the value of Lambda.