CS 6313 Statistical Methods in Data Science
Mini Project 5

Gaurav Joshi

April 23, 2022

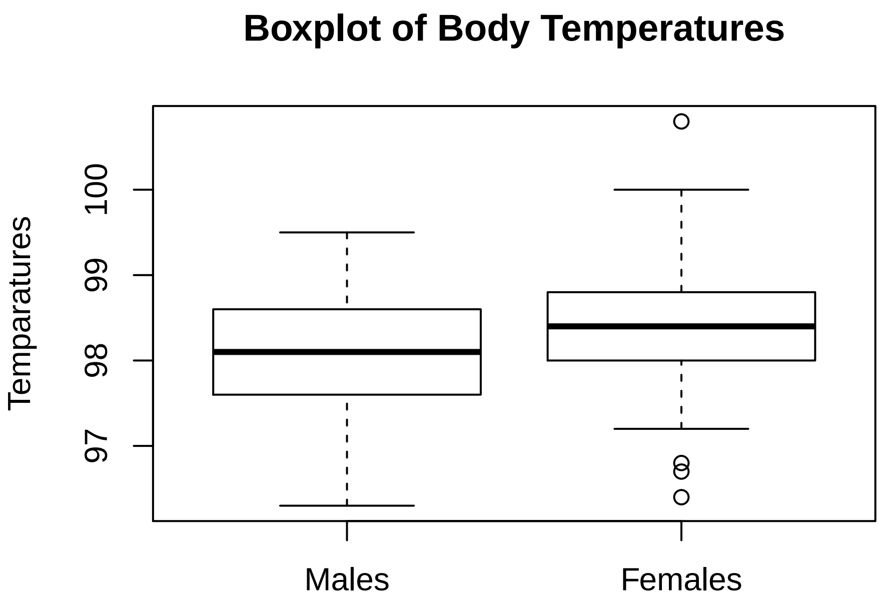
Question 1

(a)

To check the difference in mean body temperatures of males and females, we start by drawing a box plot.

The R Code to draw box-plot is given below.

```r
# Part 1: Draw scatter plot with the line

# read temperature from data set
body_temp_rate <- read.csv("bodytemp-heartrate.csv")

# Separate male and female data from the data sets
males <- subset(body_temp_rate, body_temp_rate$gender == 1)
females <- subset(body_temp_rate, body_temp_rate$gender == 2)


# draw a scatter plot of gpa wrt to act
boxplot(males$body_temperature, females$body_temperature,
        names = c('Males', 'Females'), main = "Boxplot of Body Temperatures",
        ylab = "Temparatures")
```

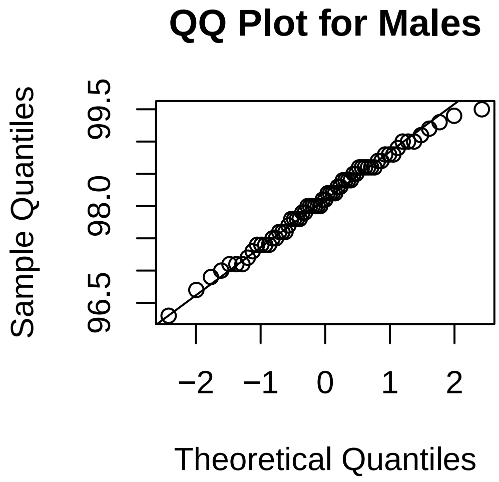


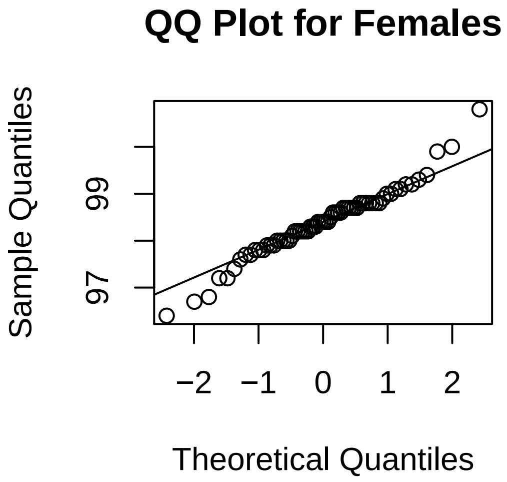
We can make the following observations using the box plot.

- Females have a higher mean body temperature than males.
- Female body temperature has more outliers than their male counterparts.
- Female body temperature box plot also has greater Q1, Q3 and median values than male body temperature.

Thus, we cannot assume equal variance. We use QQ-plot for this purpose.

```
# Draw qqnorm with qqline for male temperatures
qqnorm(males$body_temperature, main = 'QQ Plot for Males')
qqline(males$body_temperature)

# Draw qqnorm with qqline for female temparatures
qqnorm(females$body_temperature, main = 'QQ Plot for Females')
qqline(females$body_temperature)
```





From the QQ plot, it is clear that male and female body temperature approximately have a normal distribution.

We want to check if there is any difference between mean body temperatures of males and females. Thus, the null hypothesis H_0 will be difference in mean equal to 0 whereas the alternative hypothesis H_1 will be difference in mean not equal to 0

Thus, $H_0 = \bar{m} - \bar{f} = 0$ and $H_1 = \bar{m} - \bar{f} \neq 0$ where m stands for sample mean of male body temperatures and f stands for sample mean of female body temperatures.

Since, the two plots are normally distributed with unequal variance, we use t-distribution with Satterthwaite's approximation to get the confidence interval.

```
# Store the body temperatures as a separate variables
male_temperature <- males$body_temperature
female_temperature <- females$body_temperature

# Perform t.test on the male and female temperatures
t.test(male_temperature, female_temperature, alternative = 'two.sided')

##
##  Welch Two Sample t-test
##
## data:  male_temperature and female_temperature
## 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
```

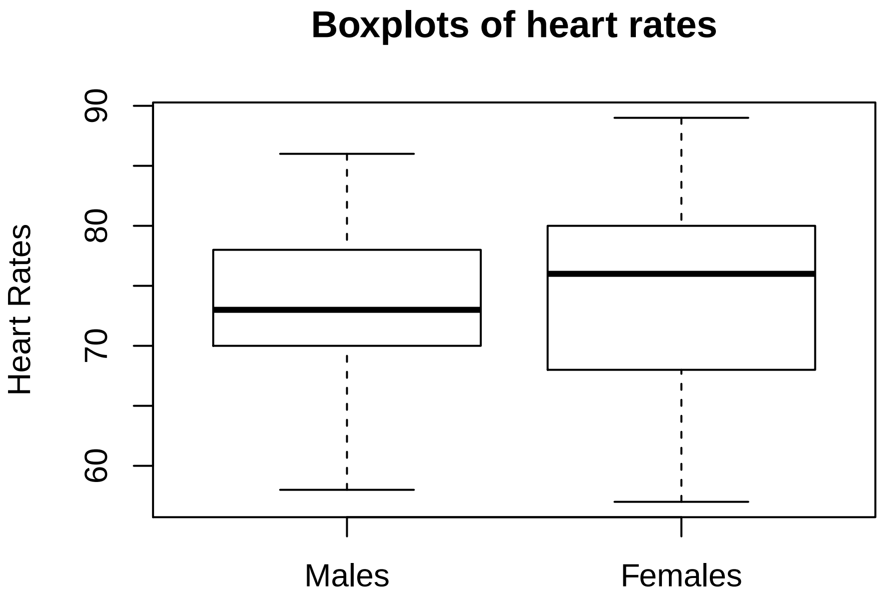
Thus, we obtain `p-value = 0.02394` and confidence interval = `(-0.53964856, -0.03881298)`

Since the p-value is less than 0.05 and 0 does not lie in the confidence interval, the null hypothesis is incorrect. We can reject the null hypothesis, meaning the mean body temperature of males and females is different.

(b)

To check the difference in mean heart rates of males and females, we start by drawing a box plot.

```r
boxplot(males$heart_rate, females$heart_rate, main = "Boxplots of heart rates",
        names = c('Males', 'Females'), ylab = 'Heart Rates')
```



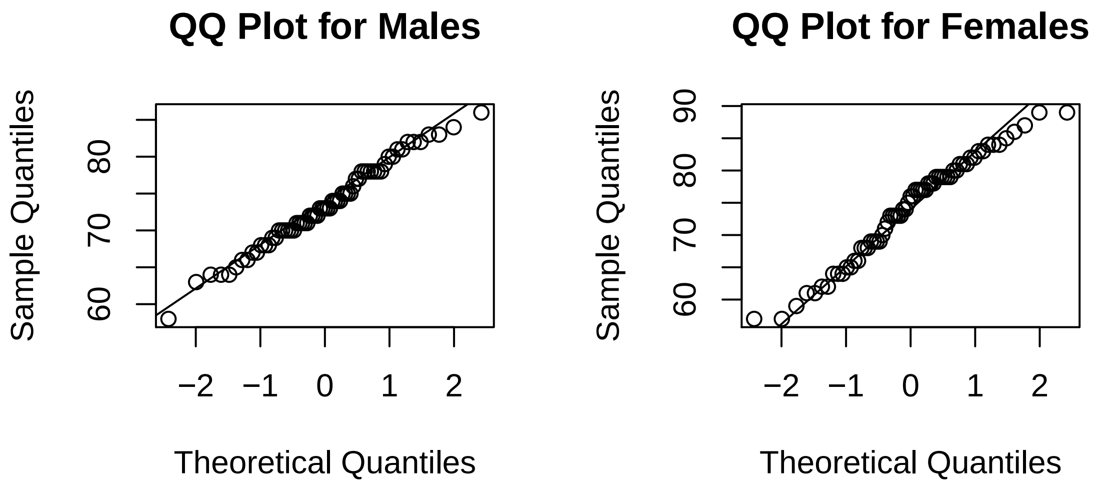
We can make the following observations using the box plot.

- Male heart rates have a higher Q1 than female heart rates
- Female heart rates have a higher Q3 and median than male heart rate
- Female heart rate box plot shows more variability than male heart rate box plot.

Thus, we cannot assume equal variance. We use QQ plot for this purpose.

```r
# Draw qqnorm with qqline for male heart rates
qqnorm(males$heart_rate, main = "QQ Plot for Males")
qqline(males$heart_rate)
```

```r
qqnorm(females$heart_rate, main = "QQ Plot for Females")
qqline(females$heart_rate)
```



From the QQ plot, it is clear that male and female heart rates approximately have a normal distribution.

We want to check if there is any difference between mean heart rates of males and females. Thus, the null hypothesis H_0 will be difference in mean equal to 0 whereas the alternative hypothesis H_1 will be difference in mean not equal to 0

Thus, $H_0 = \bar{m} - \bar{f} = 0$ and $H_1 = \bar{m} - \bar{f} \neq 0$ where m stands for sample mean of male heart rates and f stands for sample mean of female heart rates.

Since, the two plots are normally distributed with unequal variance, we use t-distribution with Satterthwaite's approximation to get the confidence interval.

```r
# Perform t.test on the male and female heart rates
t.test(males$heart_rate, females$heart_rate, alternative = 'two.sided')
```

```
##
##  Welch Two Sample t-test
##
## data:  males$heart_rate and females$heart_rate
## 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
```

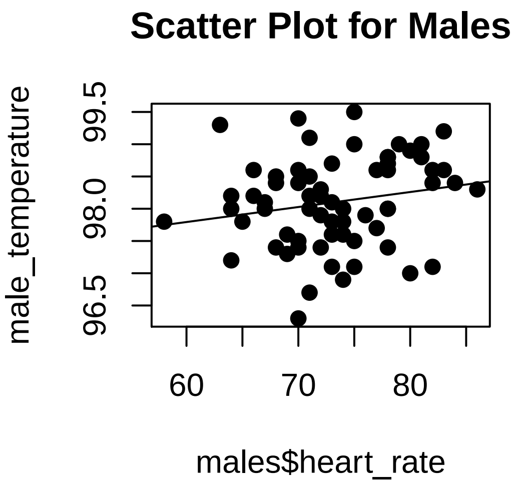
Thus, we obtain `p-value = 0.5287` and confidence interval = `(-3.243732, 1.674501)`

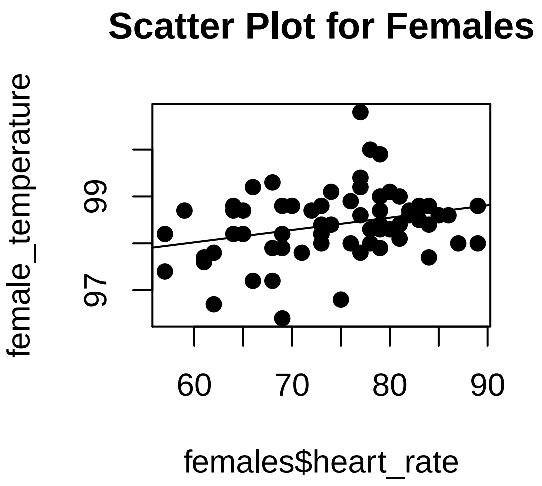
Since the p-value is greater than 0.05 and 0 does lie in the confidence interval, the null hypothesis is correct. We can accept the null hypothesis, meaning the mean heart rates of males and females is equal.

(c)

To check if there is a linear relationship between body temperature and heart rates, we can draw a scatter plot with a regression line that examines linear relationship between the two entities for males and females.

```
# Draw qqnorm with qqline for male heart rates
plot(males$heart_rate, male_temperature, pch = 19,
     main = "Scatter Plot for Males")
abline(lm(males$body_temperature~males$heart_rate))

plot(females$heart_rate, female_temperature, pch = 19,
     main = "Scatter Plot for Females")
abline(lm(females$body_temperature~females$heart_rate))
```





Since the `abline` function plots a line with positive slope for both the graphs, it indicates positive correlation between body temperature and heart rates for males and females.

We can get the correlation between the two variables using the `cor` function.

```
# Find correlation values between heart rate and body temperature
cor(males$body_temperature, males$heart_rate)
```

```
## [1] 0.1955894

cor(females$body_temperature, females$heart_rate)

## [1] 0.2869312
```

We can see that the females have a greater correlation value of `0.2869312` compared to their male counterparts `0.1955894`.

Thus, females have a stronger correlation between body temperature and heart rate than males.

Question 2

(a)

The Monte Carlo estimations of coverage probabilities can be constructed using confidence intervals. We use the following methods to fix to construct the intervals

- `z_confidence`: This method takes n and λ values as input and returns if the confidence intervals contains the true mean.
- `z_proportion`: This method runs the `z_confidence` method 5000 times and calculates coverage probability and returns the mean.
- `b_confidence`: This method takes n and λ values as input and returns if the confidence intervals contains the true mean.
- `b_proportion`: This method runs the `z_confidence` method 5000 times and calculates coverage probability and returns the mean.

```
z_confidence <- function(n, lambda) {
  U <- rexp(n, lambda)
  lower_bound <- mean(U) - qnorm(0.975) * sd(U)/sqrt(n)
  upper_bound <- mean(U) + qnorm(0.975) * sd(U)/sqrt(n)

  # return true if lies within the interval
  if (upper_bound > 1/lambda & lower_bound < 1/lambda) return(1)

  return(0)

}

z_proportion <- function(n, lambda) {
  val <- replicate(5000, z_confidence(n, lambda))

  return(length(val[which(val == 1)])/5000)
}

z_interval <- z_proportion(5, 0.01)
z_interval

## [1] 0.8256
```

Similarly, we can compute b_interval (`bootstrap interval`) using R.

```
m_star <- function(n, lambda) {
  U <- rexp(n, lambda)
  return (mean(U))
}
```

```r
b_confidence <- function(n, lambda) {
  U <- rexp(n, lambda)
  inverse_lambda <- 1/lambda
  inverse_mean <- 1/mean(U)

  V <- replicate(1000, m_star(n, inverse_mean))
  bound <- sort(V)[c(25, 975)]

  if (bound[2] > inverse_lambda & bound[1] < inverse_lambda) return(1)

  return (0)
}

b_proportion <- function(n, lambda) {
  val <- replicate(5000, b_confidence(n, lambda))
  one <- val[which(val == 1)]
  return (length(one)/5000)
}

b_interval <- b_proportion(5, 0.01)
b_interval

## [1] 0.8906
```

Thus, we get the `Z-interval`: 0.8256 and `bootstrap interval`: 0.8906

(b)

We can repeat the above process for the remaining combinations. We get the following table for Z-proportions:

`Z-Proportions`	`L = 0.01`	`L = 0.1`	`L = 1`	`L = 5`	`L = 10`
`N = 5`	0.8256	0.8076	0.8096	0.824	0.8064
`N = 10`	0.8654	0.8724	0.8628	0.8716	0.8762
`N = 30`	0.9248	0.9174	0.9202	0.9114	0.9198
`N = 100`	0.9386	0.9456	0.9356	0.9464	0.9414

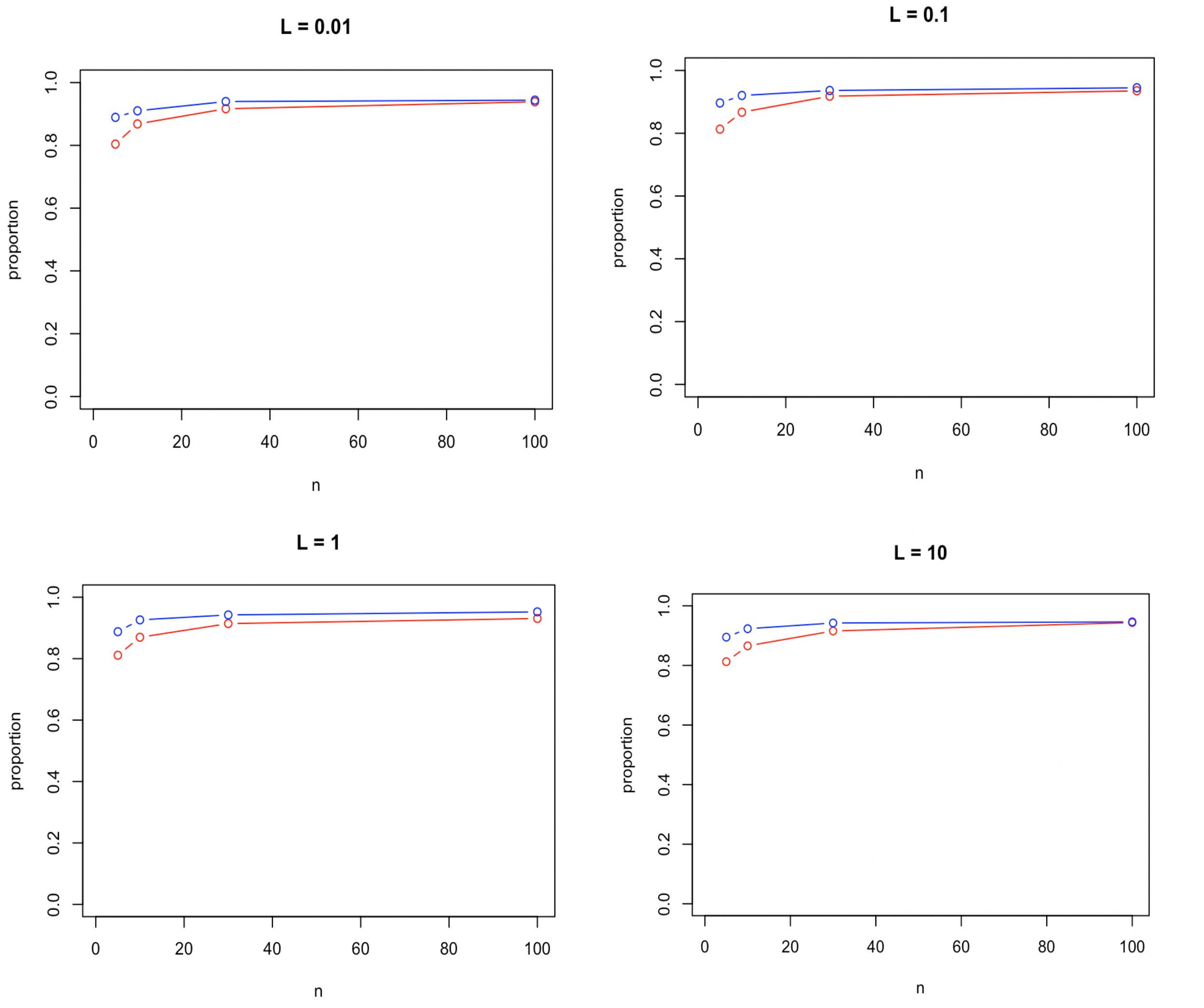
Similarly, we can also get the B-proportion table.

`B-Proportions`	`L = 0.01`	`L = 0.1`	`L = 1`	`L = 5`	`L = 10`
`N = 5`	0.8906	0.8922	0.8972	0.8974	0.8966
`N = 10`	0.924	0.9238	0.9256	0.9234	0.9178
`N = 30`	0.9326	0.9346	0.938	0.9386	0.941
`N = 100`	0.9488	0.9494	0.9456	0.9466	0.9448

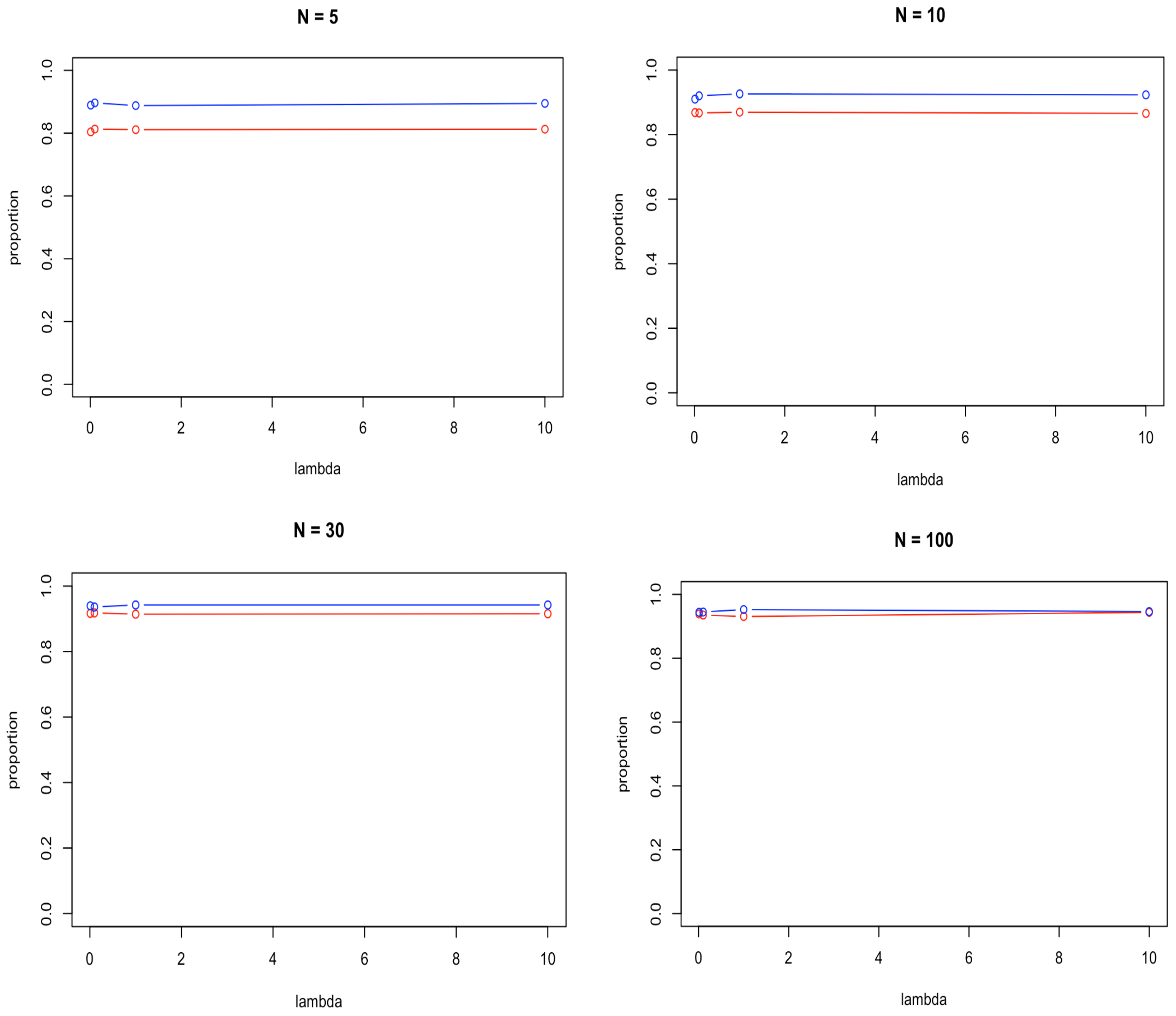
R Code to Generate Graphs:

```r
# Create a matrix of z_prop values with sorted by col of L values
zci <- matrix(c(z_proportion(5, 0.01), z_proportion(10, 0.01),
                z_proportion(30, 0.01), z_proportion(100, 0.01),
                z_proportion(5, 0.1),  z_proportion(10, 0.1),
                z_proportion(30, 0.1), z_proportion(100, 0.1),
                z_proportion(5, 1), z_proportion(10, 1),
                z_proportion(30, 1), z_proportion(100, 1),
                z_proportion(5, 10), z_proportion(10, 10),
                z_proportion(30, 10), z_proportion(100, 10)),
                nrow = 4, ncol = 4)

bci <- matrix(c(b_proportion(5, 0.01), b_proportion(10, 0.01),
                b_proportion(30, 0.01), b_proportion(100, 0.01),
                b_proportion(5, 0.1),  b_proportion(10, 0.1),
                b_proportion(30, 0.1), b_proportion(100, 0.1),
                b_proportion(5, 1), b_proportion(10, 1),
                b_proportion(30, 1), b_proportion(100, 1),
                b_proportion(5, 10), b_proportion(10, 10),
                b_proportion(30, 10), b_proportion(100, 10)),
                nrow = 4, ncol = 4)

# Plot graphs for L = 0.01, 0.1, 1, 10 using the logic given below
# The only change would be zci[, 2] and bci[, 2] to
# zci[, 4] and bci[, 4] to generate remaining plots
plot(c(5, 10, 30, 100), zci[, 1], xlab = 'n', ylab = 'proportion',
     main = "L = 0.01", xlim = c(1, 100), ylim = c(0, 1),
     type = 'b', col = 'red')
lines(c(5, 10, 30, 100), bci[, 1], type = 'b', col = 'blue')

# Plot graphs for N = 5, 10, 30 and 100 using the logic given below
# The only change would be zci[2, ] and bci[2, ] to
# zci[4, ] and bci[4, ] to generate remaining plots
plot(c(0.01, 0.1, 1, 10), zci[1,], xlab = 'lambda', ylab = 'proportion',
     main = "N = 5", xlim = c(0.01, 10), ylim = c(0, 1),
     type = 'b', col = 'red')
lines(c(0.01, 0.1, 1, 10), bci[1,], type = 'b', col = 'blue')
```

In the graphs below, we fix λ for N. The red line indicates z-proportion and blue line indicates bootstrap proportion.



In the graphs below, we fix N for λ. The red line indicates z-proportion and blue line indicates bootstrap proportion.



(c)

From the first set of graphs where λ is fixed for N, the graphs are very similar to each other. This can be interpreted as λ having no impact on coverage probabilities. From the second set of graphs where N is fixed for λ, we can see that the graphs are still very similar to each other. We can conclude that coverage probability does not depend on N.

It is clear that z-proportion and bootstrap proportion were similar for large samples especially from $N = 30$. Bootstrap coverage probability is far more accurate for smaller values of N and shows very little variation. Thus, bootstrap method should be recommended based on the observed results.

(d)

From the graphs in 2b it is clear that the λ values produced consistent result for coverage probability so λ has no impact on the results observed in 2c.