Mini Project 5

CS6313.001

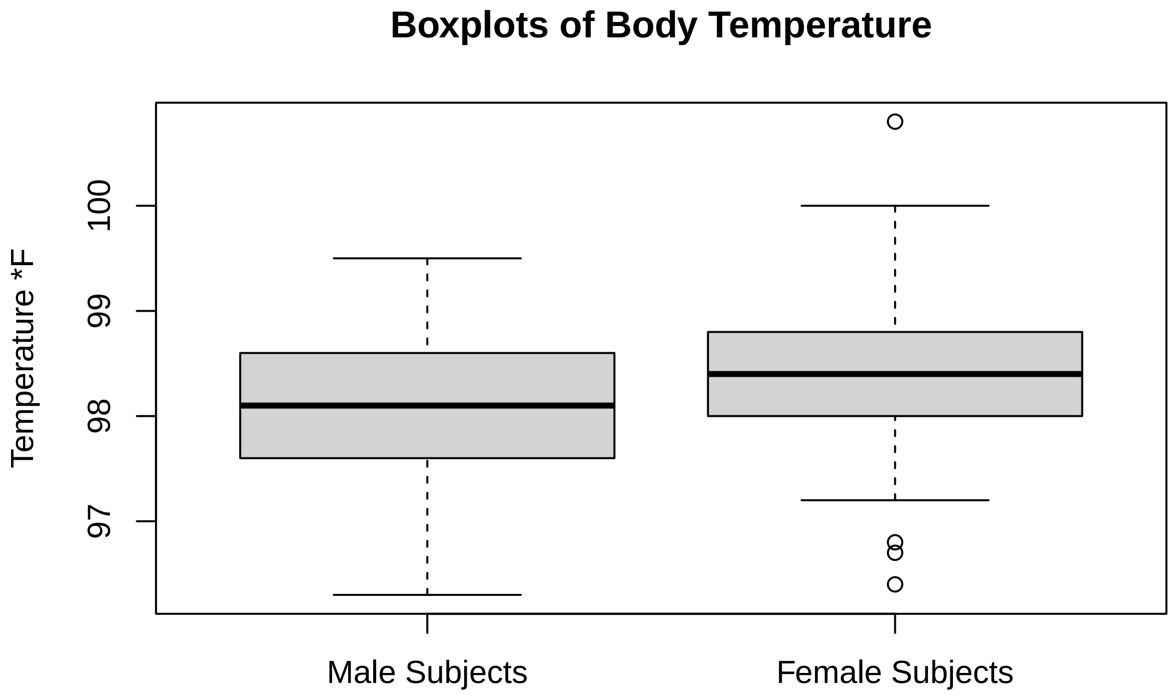
Rutvij Shah (rds190000)

19 November 2021

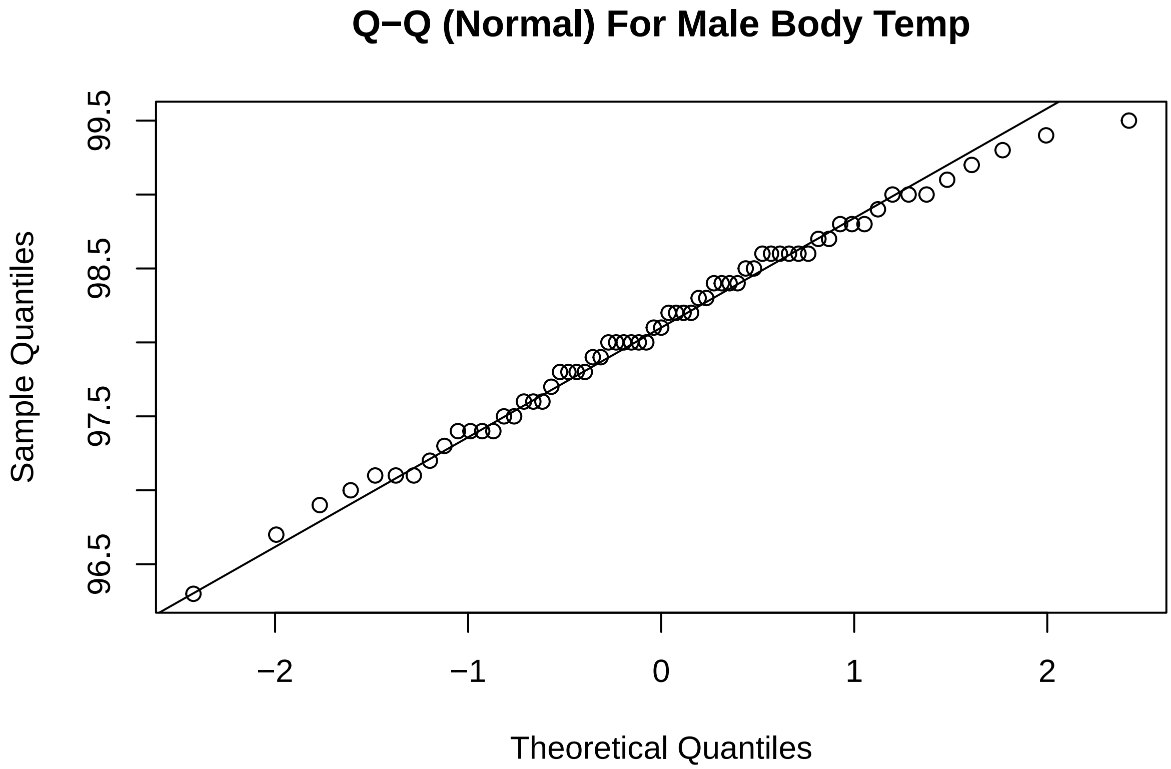
Question 1

Part (a)

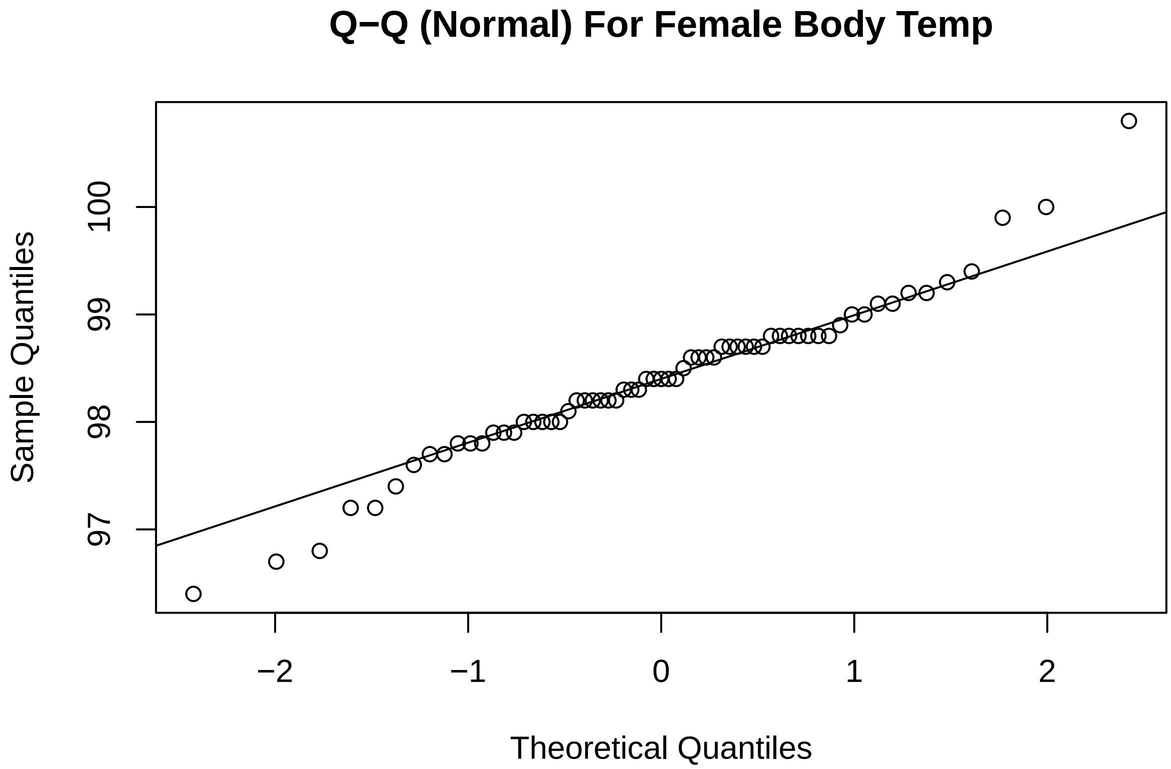
```r
bt_hr = read.csv("~/Downloads/bodytemp-heartrate.csv")

male_subs = bt_hr[bt_hr$gender == 1,][,c("body_temperature","heart_rate")]
female_subs = bt_hr[bt_hr$gender == 2,][,c("body_temperature","heart_rate")]

male_temp = male_subs$body_temperature; female_temp = female_subs$body_temperature

boxplot(
  male_temp, female_temp,
  main = "Boxplots of Body Temperature",
  names = c('Male Subjects', 'Female Subjects'),
  ylab = "Temperature *F" )
```



```r
qqnorm(male_temp, main="Q-Q (Normal) For Male Body Temp"); qqline(male_temp)
```



```r
qqnorm(female_temp, main="Q-Q (Normal) For Female Body Temp"); qqline(female_temp)
```



```
# Five Point Summary + Mean for Body Temperature by gender
summary(male_temp)
```

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

```
summary(female_temp)
```

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

```
cat("Variance male body temp:", round(var(male_temp), 2))
```

```
## Variance male body temp: 0.49
```

```
cat("Variance female body temp:", round(var(female_temp), 2))
```

```
## Variance female body temp: 0.55
```

```
# Since the variance are not equal
t.test(male_temp, female_temp, var.equal = F)
```

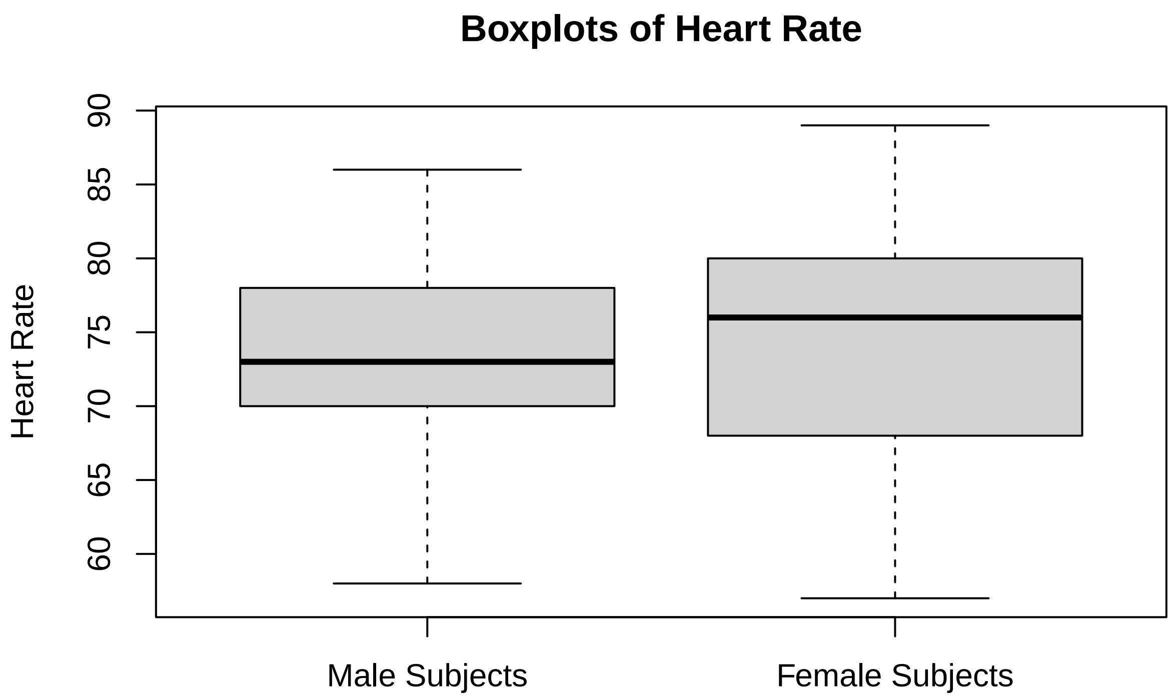
```
##
##  Welch Two Sample t-test
##
## data:  male_temp and female_temp
## t = -2.2854, df = 127.51, p-value = 0.02394
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.53964856 -0.03881298
## sample estimates:
## mean of x mean of y
##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

Conclusion: Both the box-plot EDA & the T-Test for the Null Hypothesis that both have same mean body temperature show that the Null Hypotheses is false.

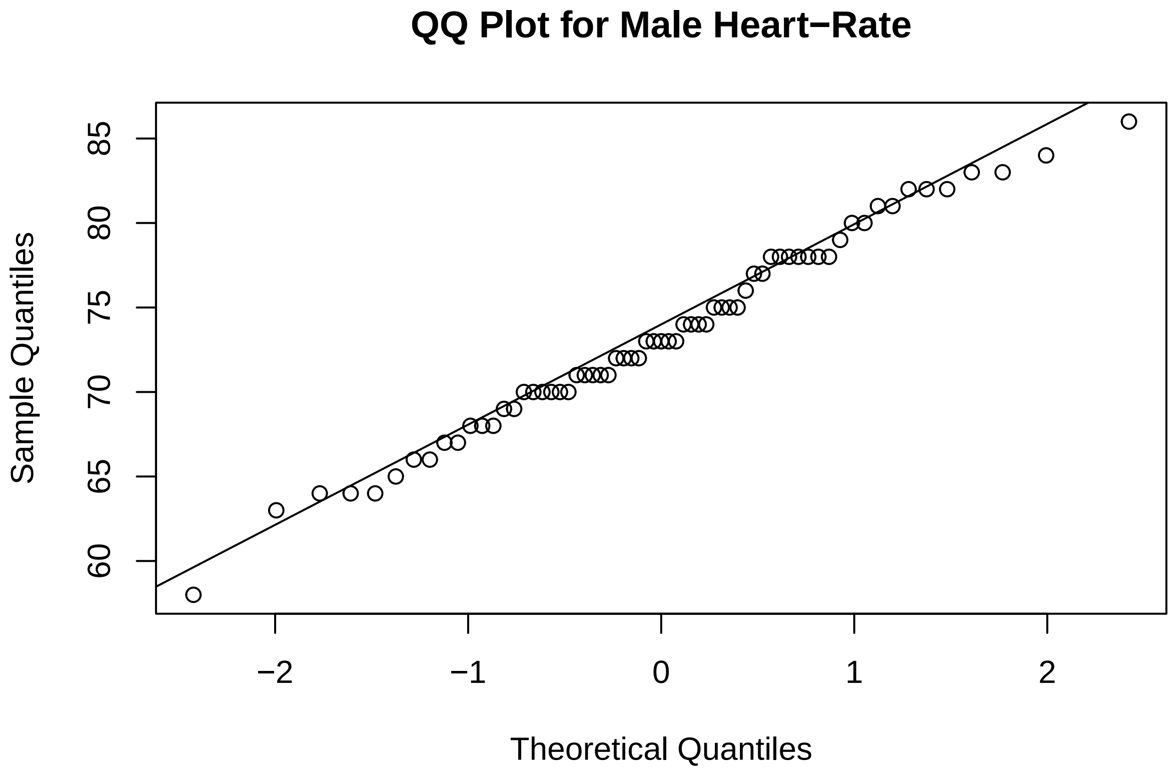
Since the value of $p = 0.024$ is less than $\alpha = 0.05$, and also since 0 does not lie in the CI, we reject the Null Hypotheses and **accept the alternative that there is a difference in means of male and female body temperature**

Part (b)

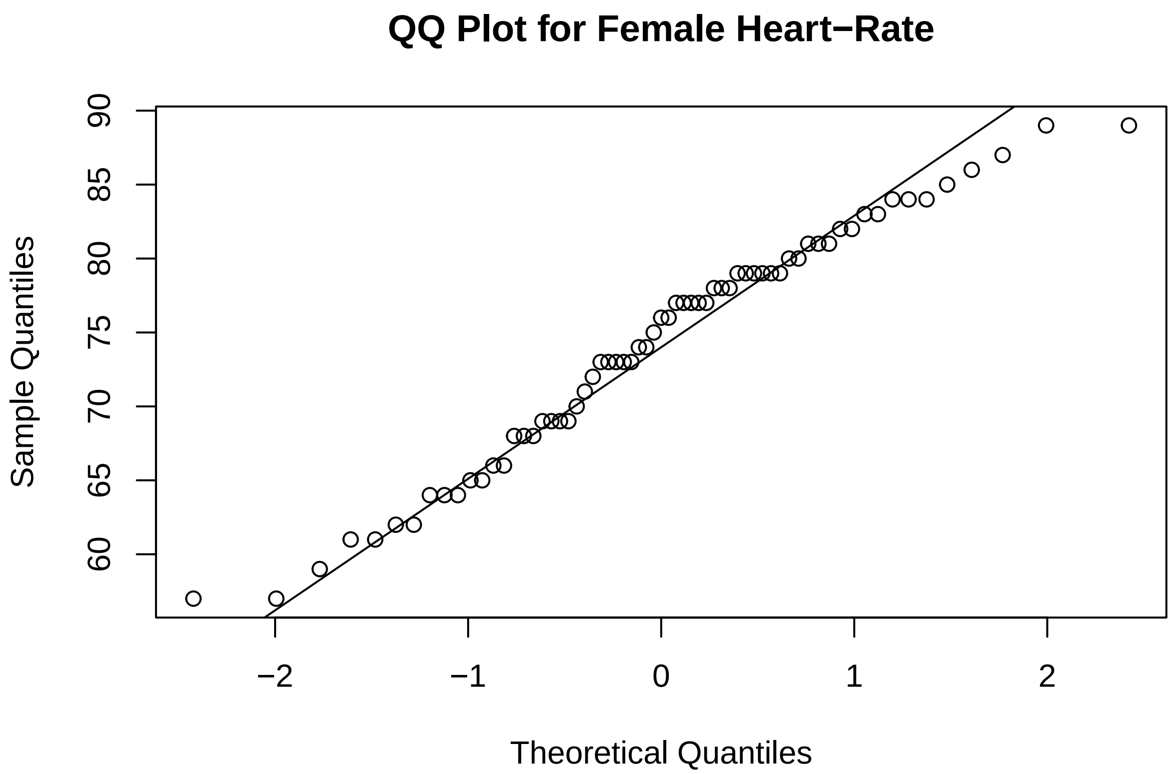
```
male_hr = male_subs$heart_rate;
female_hr = female_subs$heart_rate

boxplot(
  male_hr, female_hr,
  main = "Boxplots of Heart Rate",
  names = c('Male Subjects', 'Female Subjects'),
  ylab = "Heart Rate" )
```



```
qqnorm(male_hr, main = "QQ Plot for Male Heart-Rate"); qqline(male_hr)
```



```
qqnorm(female_hr, main = "QQ Plot for Female Heart-Rate"); qqline(female_hr)
```



```
cat("Variance male heart rate:", round(var(male_temp), 2))
```

```
## Variance male heart rate: 0.49
```

```
cat("Variance female heart rate:", round(var(female_temp), 2))
```

```
## Variance female heart rate: 0.55
```

```
t.test(male_hr, female_hr, var.equal = F)
```

```
##
##  Welch Two Sample t-test
##
## data:  male_hr and female_hr
## 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
```

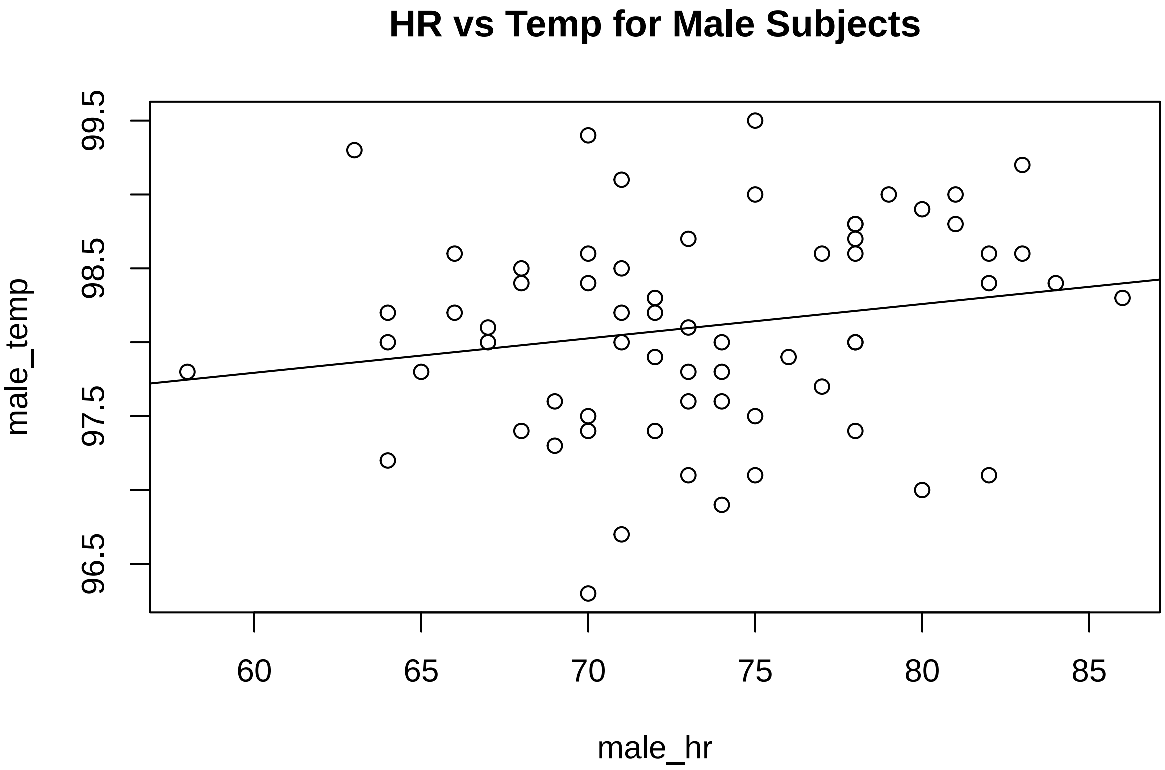
Based on both the box plot & the t-test for our hypotheses that male subjects and female subjects have similar mean heart rate, we can conclude that is indeed the case.

The null hypotheses is that both male & female heart rate has the same mean, and since $p = 0.529$ is greater than $\alpha = 0.05$, we accept the null hypotheses. Also, the difference of means, 0, lies within our CI.

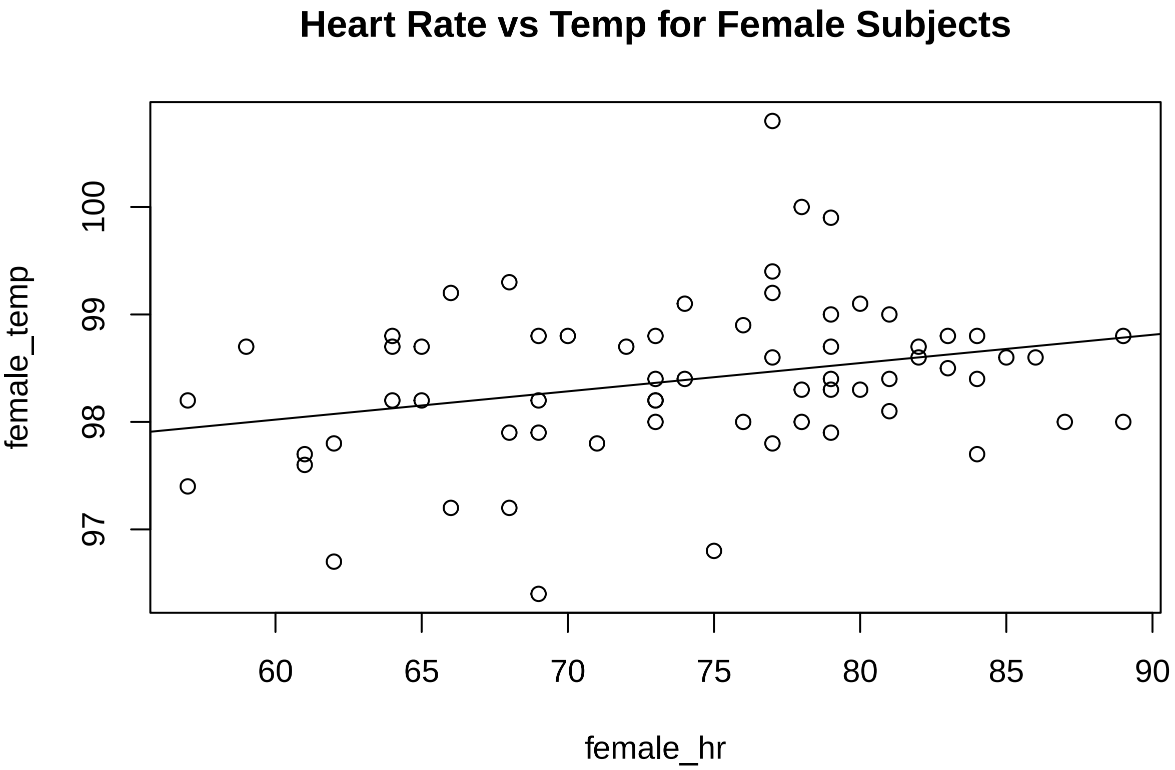
Part (c)

```
s1 = "HR vs Temp for Male Subjects"
s2 = "Heart Rate vs Temp for Female Subjects"
```

```
plot(male_hr, male_temp, main = s1); abline(lm(male_temp~male_hr))
```



```
plot(female_hr, female_temp, main = s2); abline(lm(female_temp~female_hr))
```



```
cor(male_hr, male_temp)
```

```
## [1] 0.1955894
```

```
cor(female_hr, female_temp)
```

```
## [1] 0.2869312
```

Observations:

There is weak linear correlation between heart rate and body temperature for both the genders. $\rho = 0.196$ for males and $\rho = 0.287$ for females, this also shows that the strength of the linear correlation is higher for females than for males but whether it is statistically significant is a question that can't be answered due to the small sample size of 65 each.

Question 2

Part (a)

```
library(dplyr)
library(data.table)
library(parallel)

alpha = 1 - 0.95
alpha.by2 = alpha / 2

z.ci <- function(n, lambda) {

  random_sample = rexp(n, lambda)
  true_mean = 1/lambda

  Z_CI = mean(random_sample) + c(-1, +1) * qnorm(1-alpha.by2) * sd(random_sample)/sqrt(n)

  return(as.integer(between(true_mean, Z_CI[1], Z_CI[2])))

}

mean.resample_rexp <- function(n, lambda.bar) {

  nstar <- rexp(n, lambda.bar)
  return(mean(nstar))

}

b.ci <- function(n, lambda) {

  n.boot = 1000
  random_sample = rexp(n, lambda)
  true_mean = 1/lambda

  sample_mean = mean(random_sample)
  lambda.hat = 1/sample_mean

  resamples = rexp(n*n.boot, lambda.hat)
  bootstrapsamples = matrix(resamples, nrow=n, ncol=n.boot)
  mean.stars = colMeans(bootstrapsamples)

  percentiles = c(alpha.by2, 1 - alpha.by2)
  B_CI = sort(mean.stars)[percentiles*n.boot]

  return(as.integer(between(true_mean, B_CI[1], B_CI[2])))

}

calculate_coverage_probabilities <- function(nsims, ci_estimator_func, n, lambda) {
  values <- replicate(nsims, ci_estimator_func(n, lambda))
  num_ones <- sum(values)
  return(num_ones/nsims)
}


# random sample from exponential distribution with size n & some value of lambda
lambda = 0.01
n = 5

z_cover = calculate_coverage_probabilities(5000, z.ci, n, lambda)

b_cover = calculate_coverage_probabilities(5000, b.ci, n, lambda)

cat(paste("Coverage Probability of Z interval for n: ", n, " and lambda: ",
          lambda, " is = ", z_cover))
```

```
## Coverage Probability of Z interval for n:  5  and lambda:  0.01  is =  0.8142
```

```
cat(paste("\nCoverage Probability of bootstrap interval for n: ", n,
          " and lambda: ", lambda, " is = ", b_cover))
```

```
##
## Coverage Probability of bootstrap interval for n:  5  and lambda:  0.01  is =  0.8954
```

```
funcs = c(z.ci, b.ci)
n.vals = c(5,10,30,100)
lambda.vals = c(0.01,0.1,1,10)
```

Part (b)

Table for coverage probabilities for Z Confidence Intervals

n is the first column ranging from 5 to 100 while lambdas are the each one column named with their values.

```
z_df = expand.grid(n.vals, lambda.vals)
z_df$var3 = mcmapply(calculate_coverage_probabilities,
                     5000, c(z.ci), z_df$Var1, z_df$Var2, mc.cores=7)
colnames(z_df) <- c("n", "lambda", "coverage_prob")
z_table = data.table(z_df)
data.table::dcast(z_table, n ~ lambda, value.var = "coverage_prob")
```

```
##      n   0.01    0.1      1     10
## 1:   5 0.8162 0.8110 0.8094 0.8072
## 2:  10 0.8708 0.8676 0.8610 0.8694
## 3:  30 0.9174 0.9160 0.9228 0.9120
## 4: 100 0.9416 0.9426 0.9432 0.9366
```

Table for coverage probabilities for Bootstrap Confidence Intervals

n is the first column ranging from 5 to 100 while lambdas are the each one column named with their values.

```
b_df = expand.grid(n.vals, lambda.vals)

b_df$var3 = mcmapply(calculate_coverage_probabilities,
                     5000, c(b.ci), b_df$Var1, b_df$Var2, mc.cores=7)

colnames(b_df) <- c("n", "lambda", "coverage_prob")
b_table = data.table(b_df)
data.table::dcast(b_table, n ~ lambda, value.var = "coverage_prob")
```

```
##       n   0.01    0.1      1     10
## 1:    5 0.9010 0.8960 0.8994 0.9004
## 2:   10 0.9226 0.9210 0.9176 0.9188
## 3:   30 0.9382 0.9414 0.9374 0.9368
## 4:  100 0.9406 0.9486 0.9416 0.9512
```

Part (c)

As evident from the tables, we can see that the effect of lambda is neglible on the coverage probability for a given n-value.

Both for Bootstrap CIs & Z CIs, the probabilities increase we the increase in n. And as n tends to 100 the probabilities tend to the 1-alpha value, i.e., the 95% confidence interval expected.

If time and compute power allowed, we could perform multiple runs for each n and lambda value and find the mean tendency for the CP, and then calculating the variance of mean CP for a constant and varied lambda would most likely tend to 0.

Specific interpretations.

1. Large-Sample Interval n-size for accuracy ~ n=100 (mean cp ~ 0.9381). Since the mean-cp for that n value has about 1% error wrt 95% CI.
2. Bootstrap Interval n-size for accuracy ~ n=30 (mean cp ~ 0.9389). Since the mean-cp for that n value has about 1% error wrt 95% CI.
3. No, these answers are independent of lambda.
4. Yes, Bootstrap interval method consistently outperforms large sample interval for all input sizes hence it would be by recommended method. Bootstrap CI is indeed more accurate than Large-Sample CI.

```
# Something along the lines of this...
mean_coverage_prob <- function(n, lambda) {

  x = replicate(
    10,
    calculate_coverage_probabilities(5000, b.ci, n, lambda)
  )
  return(mean(x))

}

mean_coverage_prob(5, 0.01)
```

```
## [1] 0.89872
```

Part (d)

We can say even though we fixed the value of lambda in advance these conclusions will hold irrespective, since they're largely dependent on n.

```
linear_correlation_n_cp_for_const_lambda.bootstrap = b_df %>%
  group_by(lambda) %>%
  summarise(n_prob_correlation = cor(n, coverage_prob))

linear_correlation_n_cp_for_const_lambda.bootstrap
```

```
## # A tibble: 4 x 2
##   lambda n_prob_correlation
##    <dbl>              <dbl>
## 1   0.01              0.719
## 2   0.1               0.777
## 3   1                 0.772
## 4  10                 0.872
```

```
cor(linear_correlation_n_cp_for_const_lambda.bootstrap$lambda,
    linear_correlation_n_cp_for_const_lambda.bootstrap$n_prob_correlation)
```

I will stand corrected, lambda does influence the linear correlation between, "n" & "coverage probability" in case of bootstrap CI estimates, and in fact has a strong linear correlation itself $\rho = 0.941$

```
linear_correlation_n_cp_for_const_lambda.z = z_df %>%
  group_by(lambda) %>%
  summarise(n_prob_correlation = cor(n, coverage_prob))

linear_correlation_n_cp_for_const_lambda.z
```

```
## # A tibble: 4 x 2
##   lambda n_prob_correlation
##    <dbl>              <dbl>
## 1   0.01              0.815
## 2   0.1               0.820
## 3   1                 0.808
## 4  10                 0.800
```

```
cor(linear_correlation_n_cp_for_const_lambda.z$lambda,
    linear_correlation_n_cp_for_const_lambda.z$n_prob_correlation)
```

In case of Large Interval CIs there is no dependence on lambda as $\rho = 0.124$