PS11

Aditya Sanjay Mhaske

2023-04-11

Q1

a)

```
# The Central Limit Theorem states that if the sample size is large enough (i.e., n ≥ 3
0), the distribution of the sample mean tends to be approximately normal, irrespective o
f the population distribution. In this study, the sample sizes of both groups are greate
r than 30 (i.e., n1 = 235 and n2 = 197), so we can assume that the sample means are appr
oximately normally distributed, regardless of the underlying population distribution. It
is important to consider the normality assumption when using certain statistical tests t
hat assume normality, such as the t-test. In cases where the distribution is not normal
and the sample sizes are small, non-parametric tests that do not rely on normality assum
ptions may be needed.
```

b)

```
n_1 = 235

mu1 = 23.4

sd1 = 5.7

var1 = sd1^2

n_2 = 197

mu2 = 21.9

sd2 = 7.2

var2 = sd2^2

CI = 0.98
alpha = 1-CI

se = sqrt((var1 / n_1) + (var2/n_2))
se
```

```
## [1] 0.6335634
```

```
nu <- (var1+var2)^2/(var1^2/(n_1-1)+var2^2/(n_2-1))
nu
```

```
## [1] 390.2671
```

```
t_stat = qt(1-alpha/2, nu)
t_stat
```

```
## [1] 2.335941
```

```
ci = c( (mu1- mu2) - t_stat*se, (mu1- mu2) + t_stat*se)
ci
```

```
## [1] 0.02003355 2.97996645
```

```
#Our analysis yields a 98% confidence interval, within which we can be confident that th
e actual difference in the average distance traveled by readers using orange and blue ba
ckgrounds lies.
```

c)

```
# When conducting a hypothesis test that compares the means of two separate groups with
unequal variances and sample sizes, it is appropriate to utilize Welch's two-sample t-te
st.
```

d)

```
# Null Hypotheses -
# H_0: delta = 0
# H_1: delta != 0
```

```
df = ((var1/n_1 + var2/n_2)^2) / ((var1/n_1)^2 /(n_1-1) + (var2/n_2)^2 /(n_2-1))

t = (mu1-mu2)/sqrt(var1/n_1 + var2/n_2)
t
```

```
## [1] 2.367561
```

```
p = 2 * pt(-abs(t), df)
p
```

```
## [1] 0.0184189
```

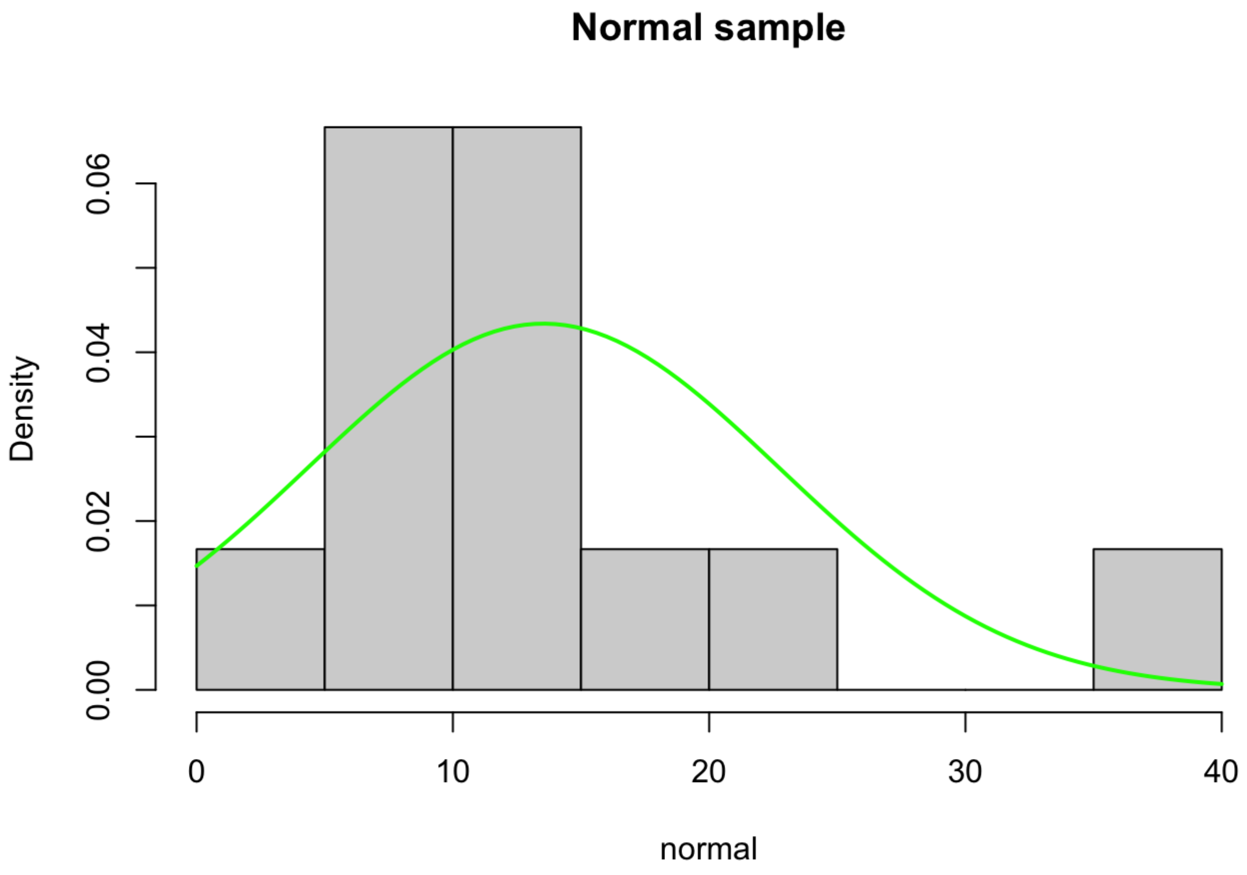
```
# Concluding based on the p-value of 0.09, we  reject the null hypothesis at
# the significance level of 0.05.
```

Q2

```
normal = c(4.1, 6.3, 7.8, 8.5, 8.9, 10.4, 11.5, 12.0, 13.8, 17.6, 24.3, 37.2)
diabetic = c(11.5, 12.1, 16.1, 17.8, 24.0, 28.8, 33.9, 40.7, 51.3, 56.2, 61.7, 69.2)
```

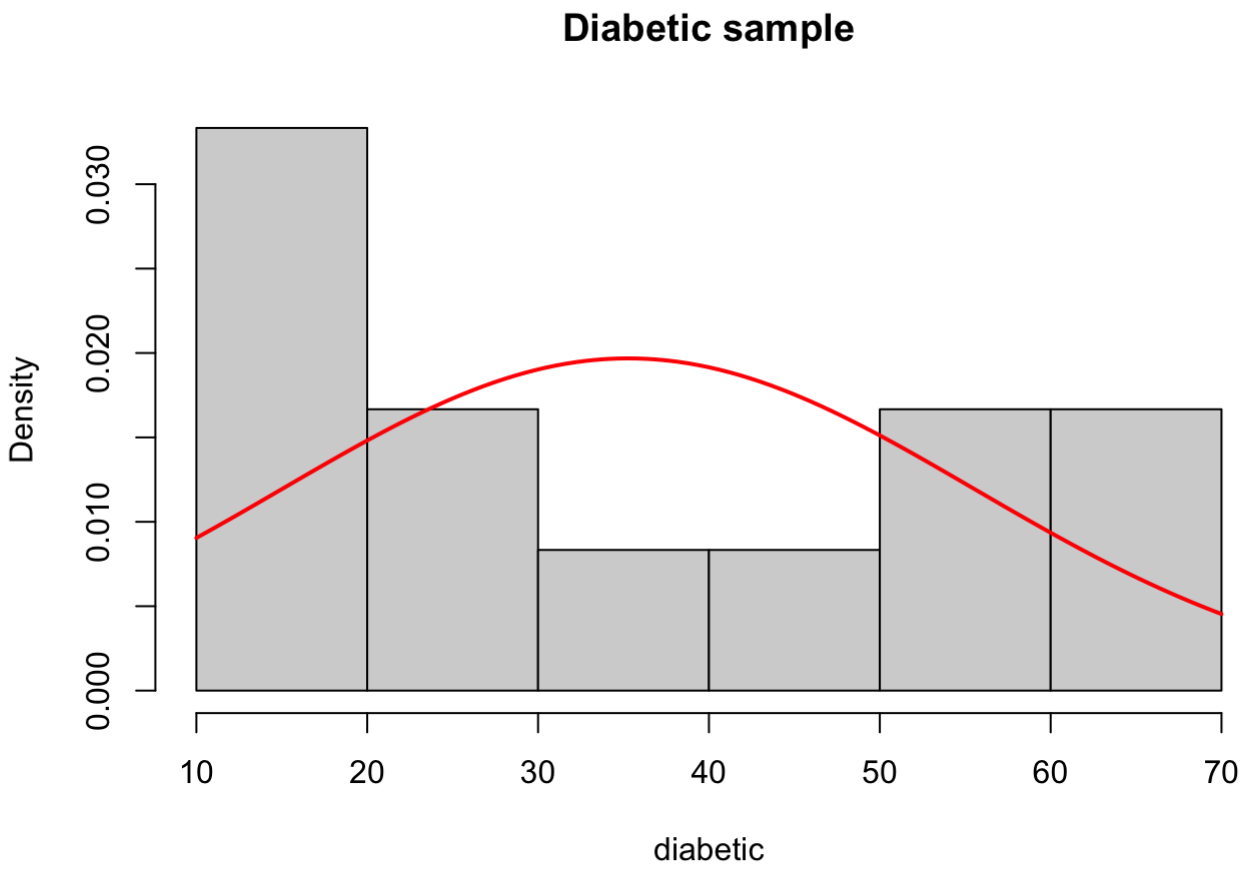
a)

```
hist(normal, main="Normal sample", freq=FALSE)
curve(dnorm(x, mean=mean(normal), sd=sd(normal)), add=TRUE, col="green", lwd=2)
```



```
# From the analysis of the histograms, it seems that the sample from the normal distribu
tion displays an approximate symmetry, as most of the data points are concentrated aroun
d the central point of the distribution.
```

```
hist(diabetic, main="Diabetic sample", freq=FALSE)
curve(dnorm(x, mean=mean(diabetic), sd=sd(diabetic)), add=TRUE, col="red", lwd=2)
```



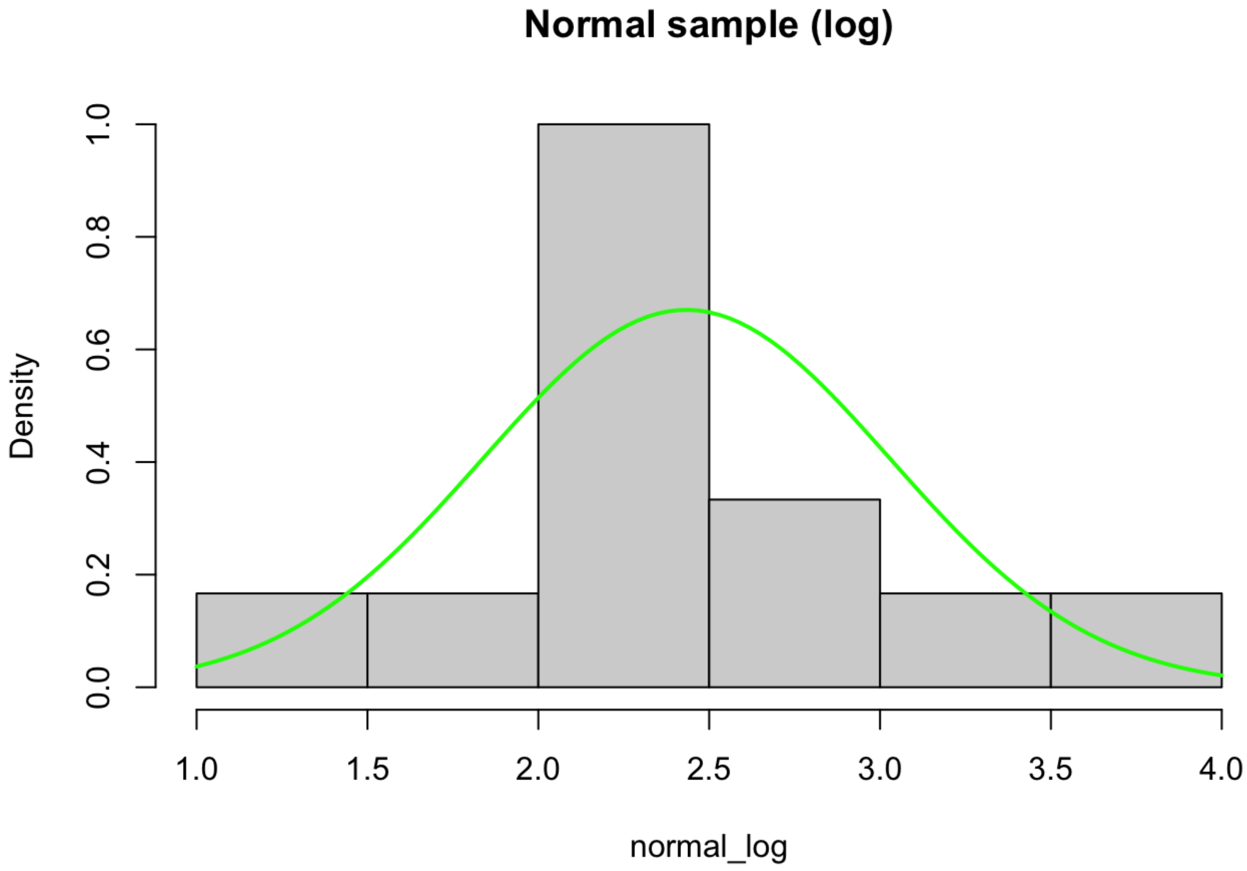
```
# In contrast to the normal sample, the sample of diabetics is observed to be skewed towards the right, exhibiting a longer right tail and fewer observations on the right side of the center. As a result, the diabetic sample is not a sample from a symmetric distribution, unlike the normal sample.
```

b)

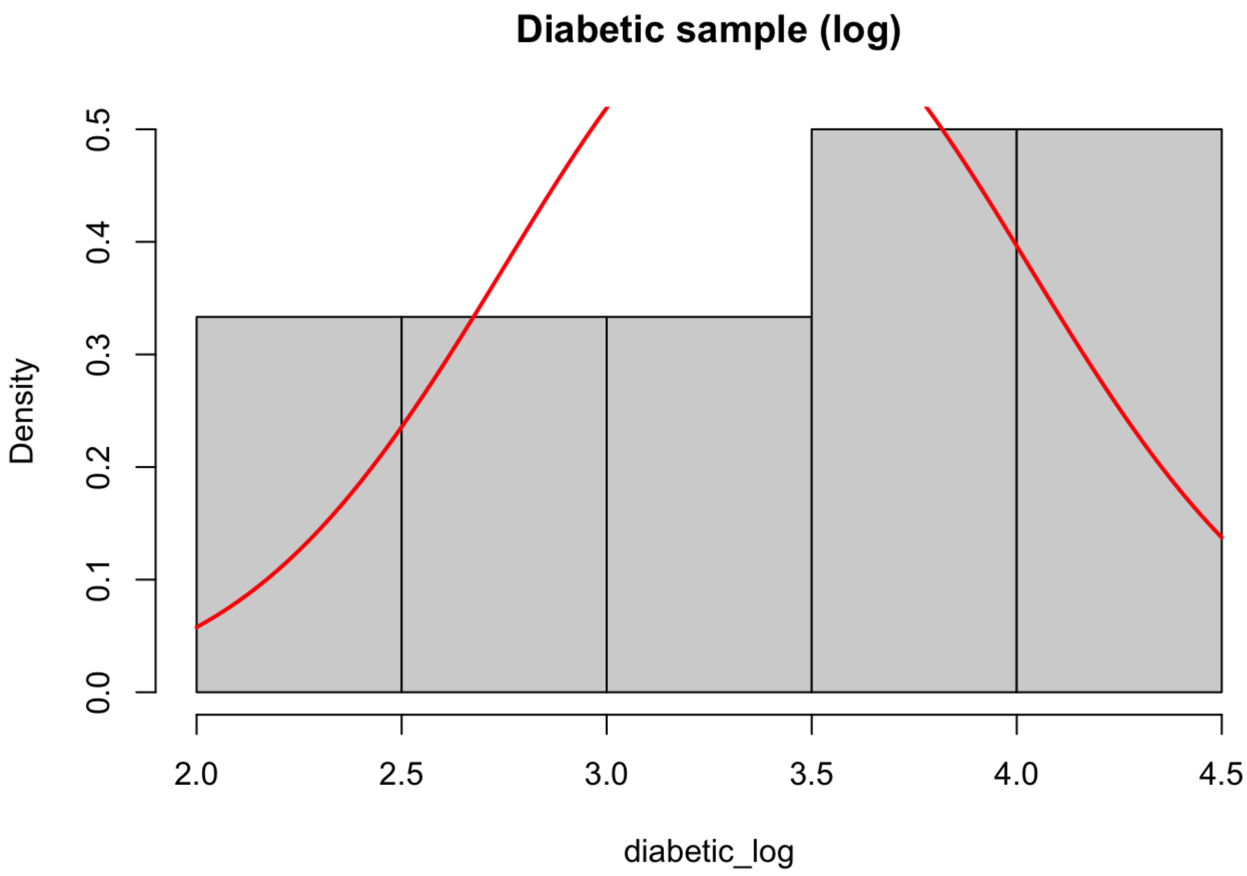
```
# Let x_i = normal and y_j = diabetic
```

```
# i) normal logarithmic transformation
```

```
normal_log = log(normal)

diabetic_log = log(diabetic)

hist(normal_log, main="Normal sample (log)", freq=FALSE)
curve(dnorm(x, mean=mean(normal_log), sd=sd(normal_log)), add=TRUE, col="green", lwd=2)
```

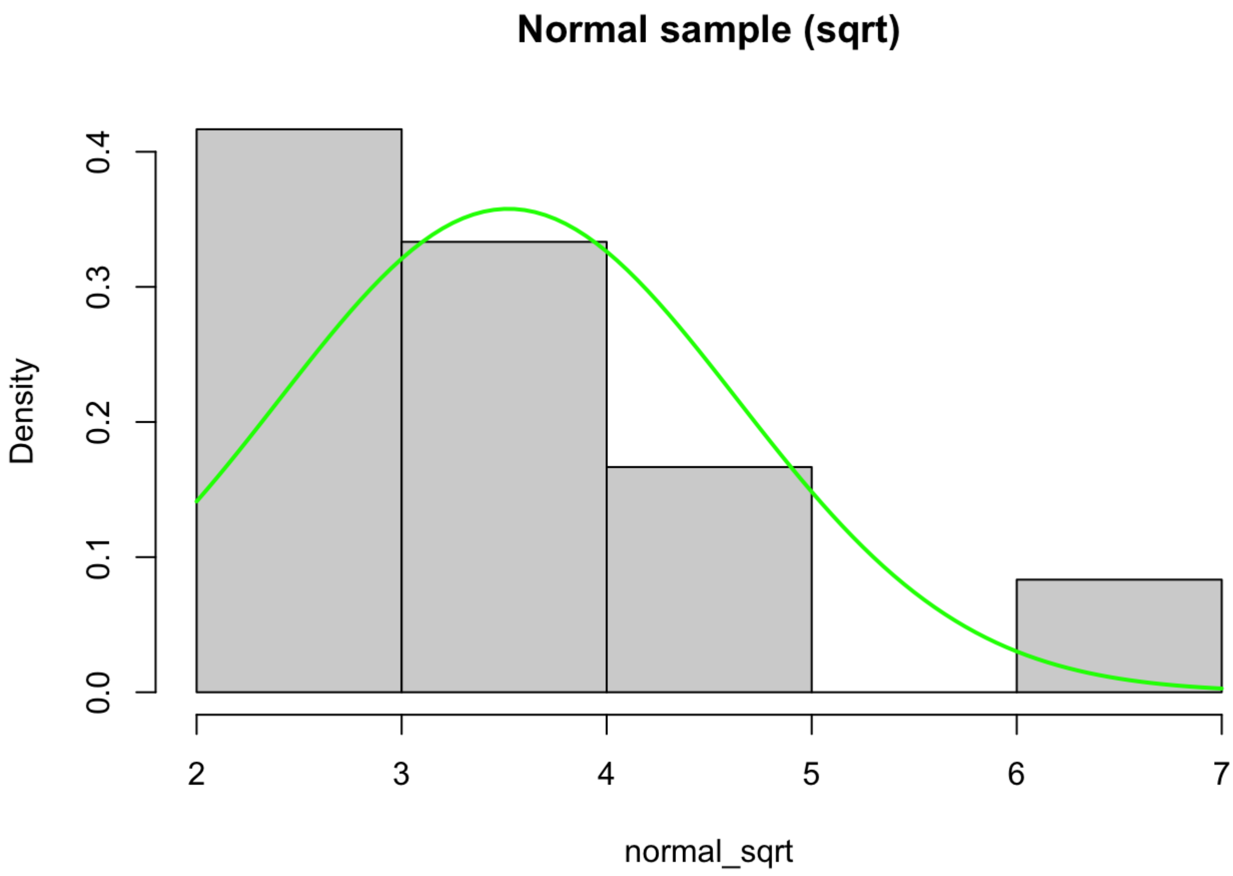


```
hist(diabetic_log, main="Diabetic sample (log)", freq=FALSE)
curve(dnorm(x, mean=mean(diabetic_log), sd=sd(diabetic_log)), add=TRUE, col="red", lwd=
2)
```

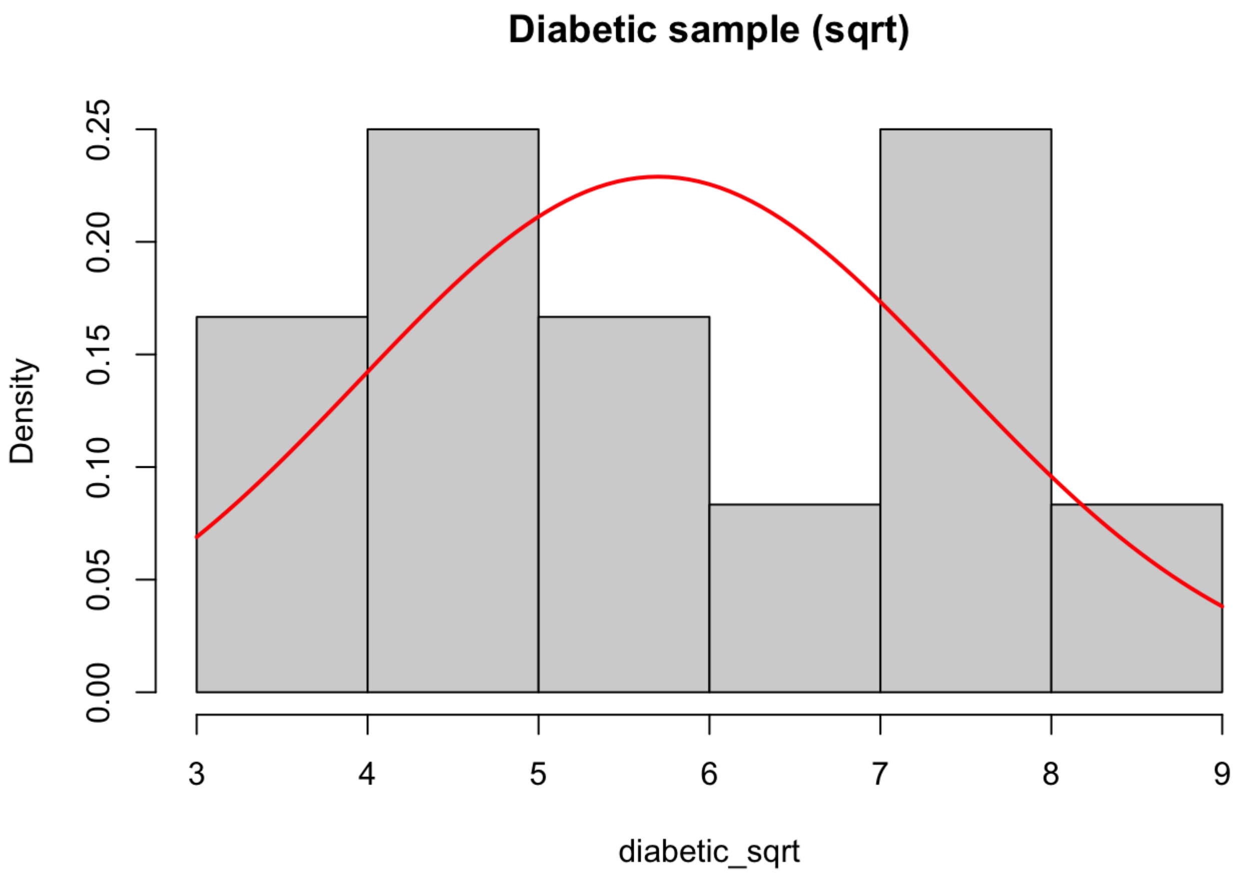


```
# ii) square root transformation
```

```
normal_sqrt = sqrt(normal)

diabetic_sqrt = sqrt(diabetic)

hist(normal_sqrt, main="Normal sample (sqrt)", freq=FALSE)
curve(dnorm(x, mean=mean(normal_sqrt), sd=sd(normal_sqrt)), add=TRUE, col="green", lwd=
2)
```



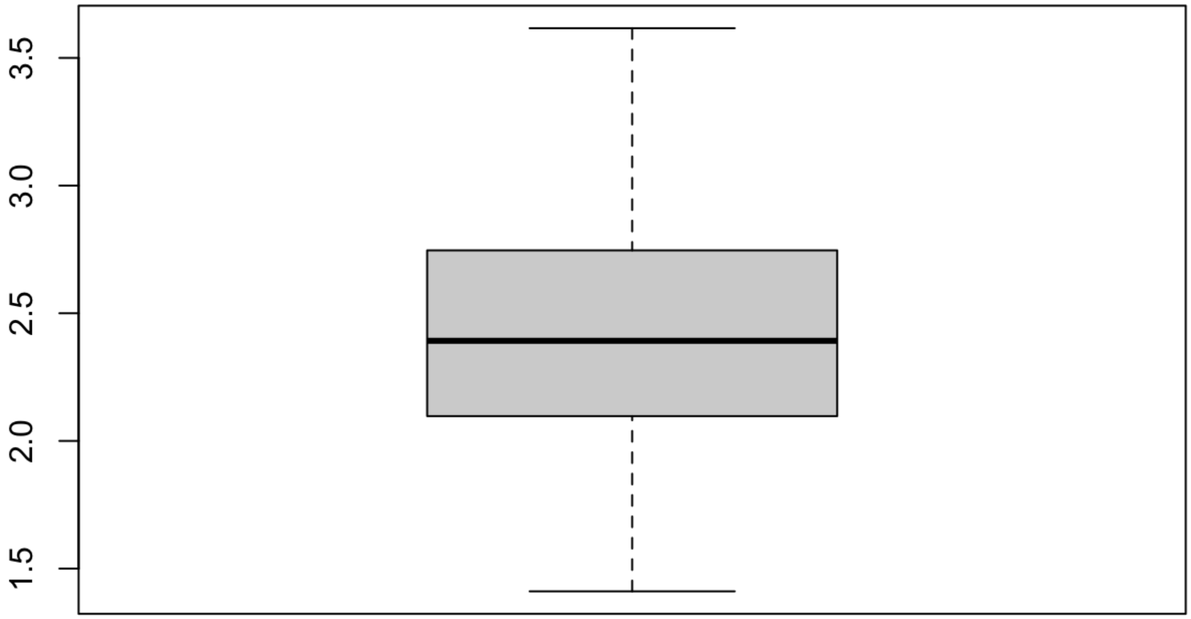
```
hist(diabetic_sqrt, main="Diabetic sample (sqrt)", freq=FALSE)
curve(dnorm(x, mean=mean(diabetic_sqrt), sd=sd(diabetic_sqrt)), add=TRUE, col="red", lwd
=2)
```



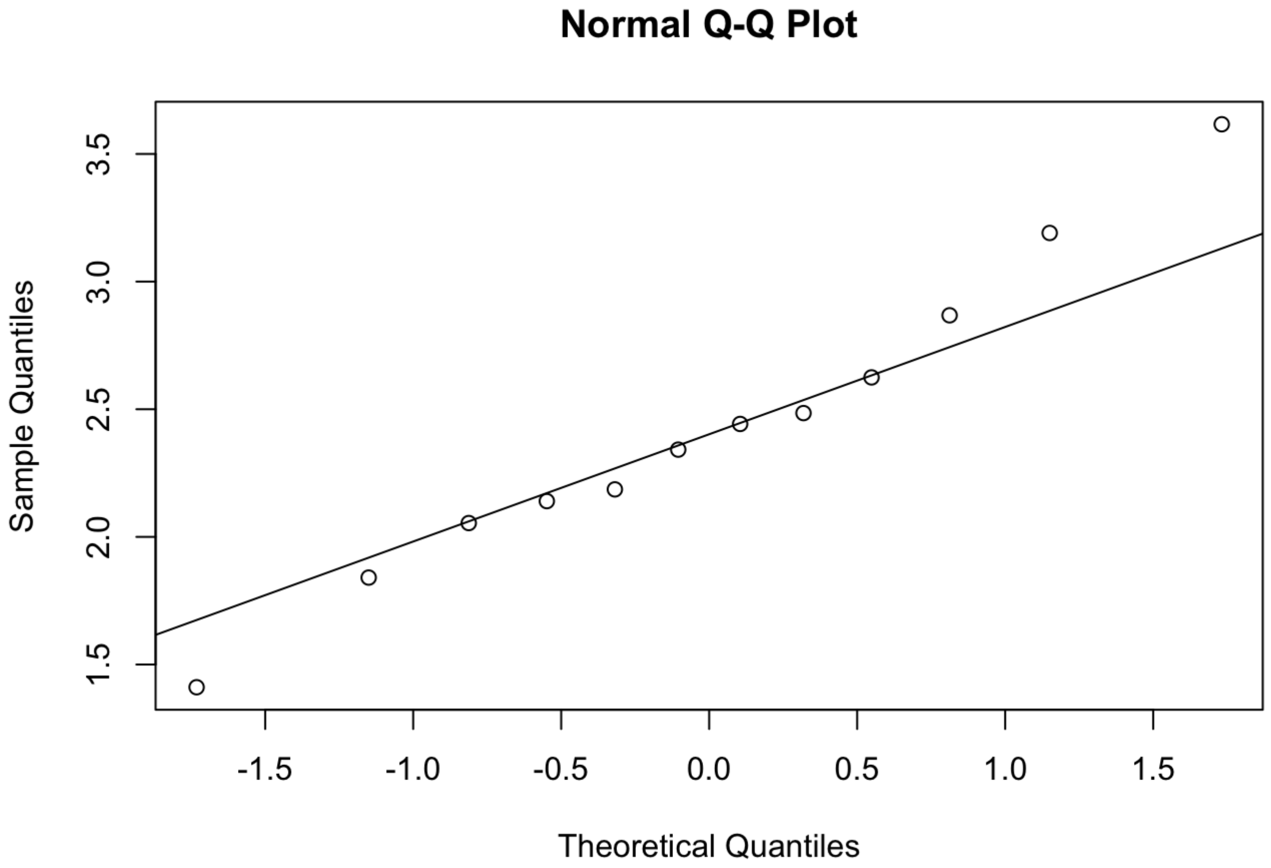
```
#After examining the histograms and comparing the curve overlays, it seems that both tra
nsformations have improved the symmetry of the two samples. Nonetheless, the natural log
arithm transformation seems to have had a greater impact in reducing their skewness. Thi
s is evident from the transformed samples' histograms, which are more centered and less
skewed than those of the original samples or the square root transformed samples. Conseq
uently, this analysis indicates that the natural logarithm transformation is the more fa
vorable choice for these two samples.
```

c)

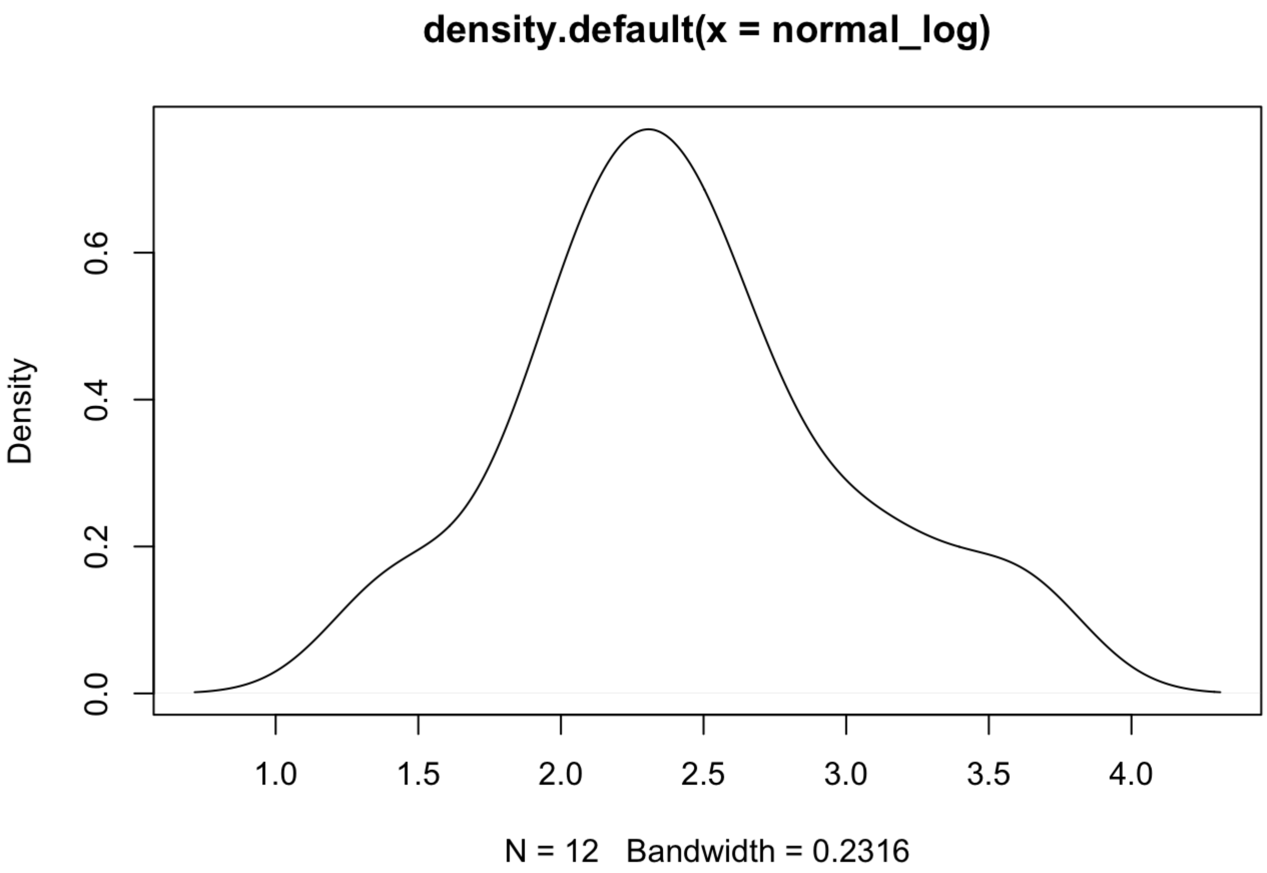
```
# normal sample logartihm
boxplot(normal_log)
```



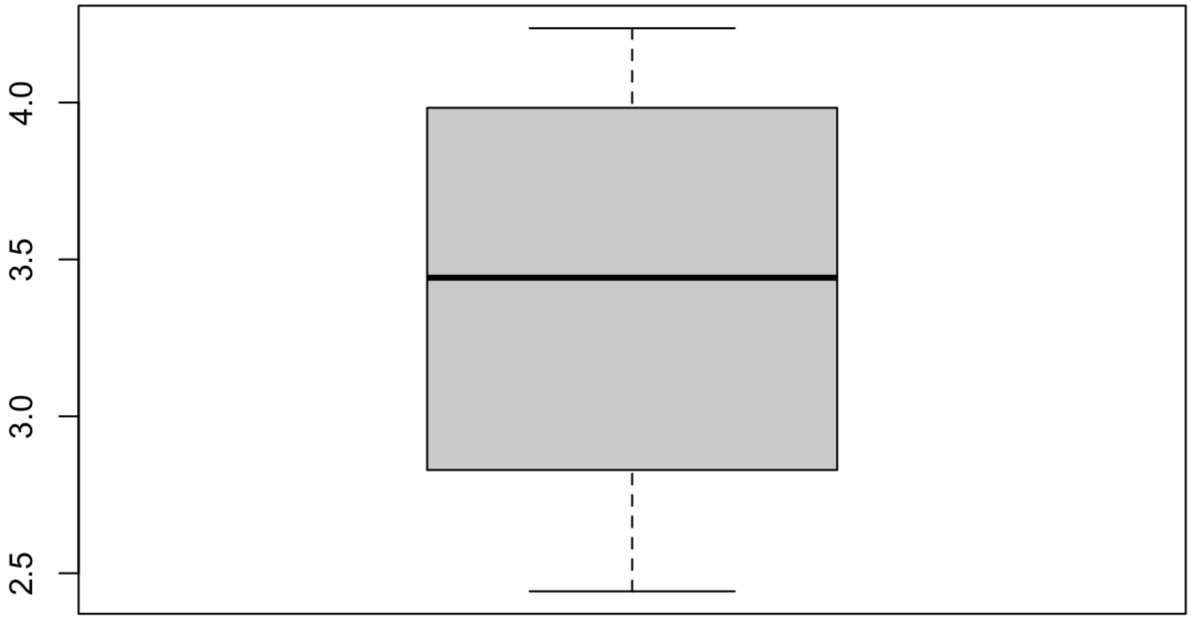
```
qqnorm(normal_log)
qqline(normal_log)
```



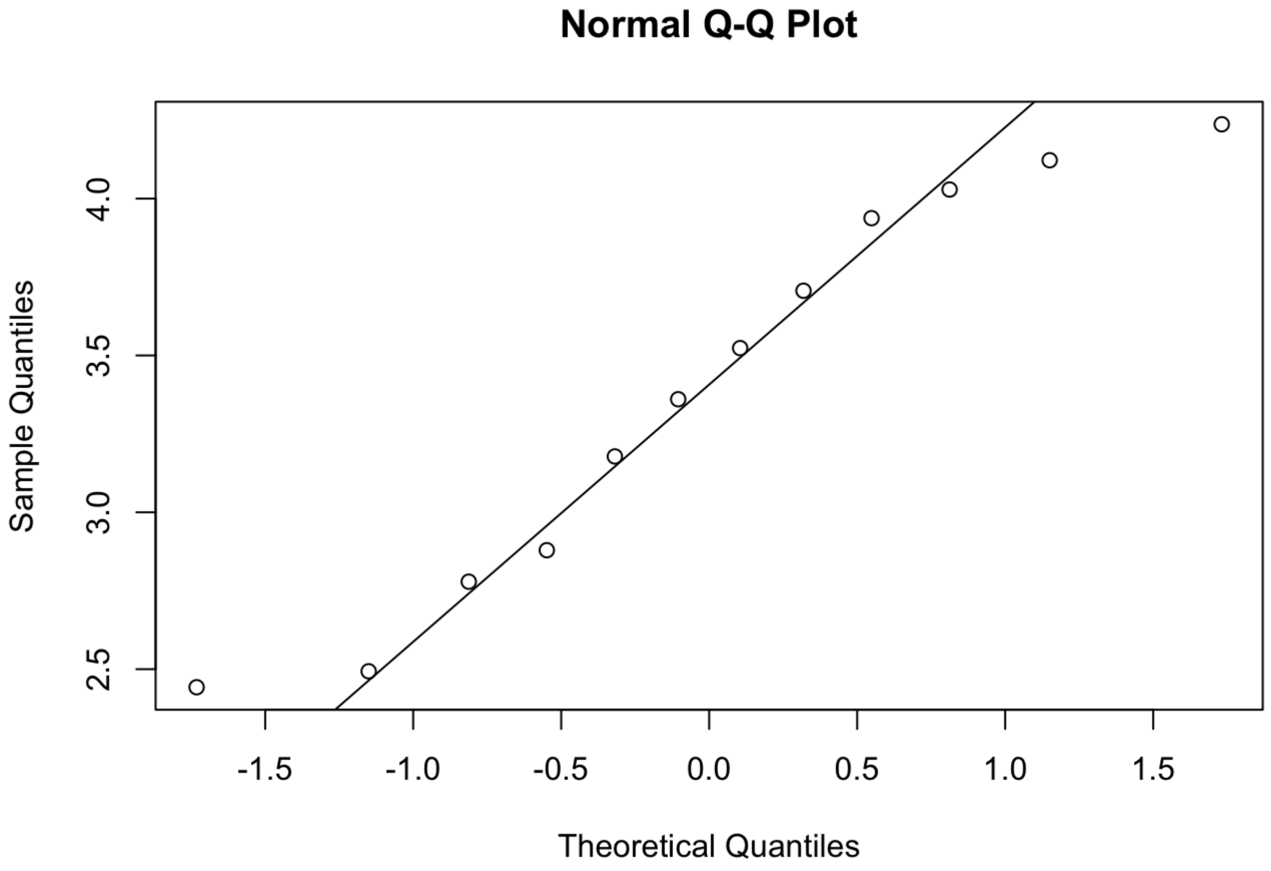
```
plot(density(normal_log))
```



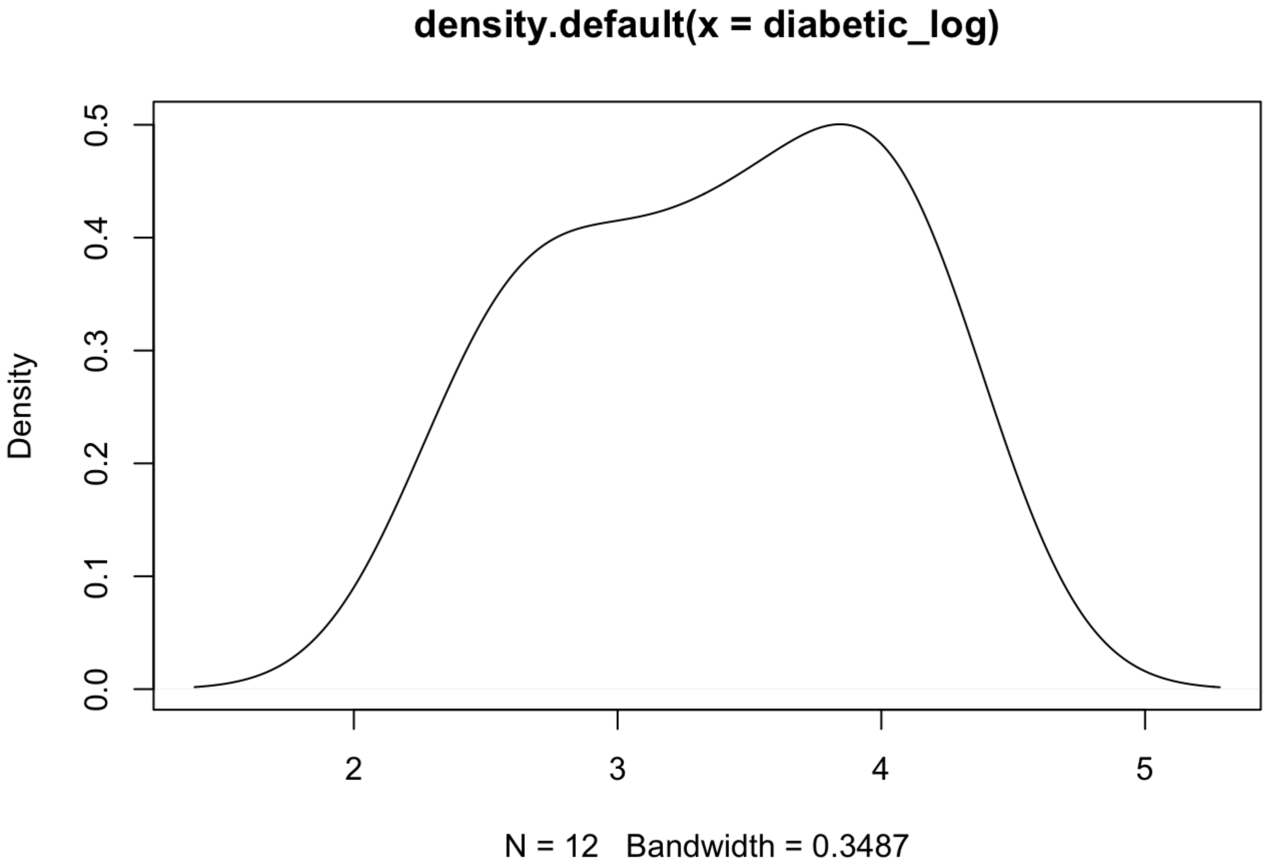
```
# diabetic sample logarithm
boxplot(diabetic_log)
```



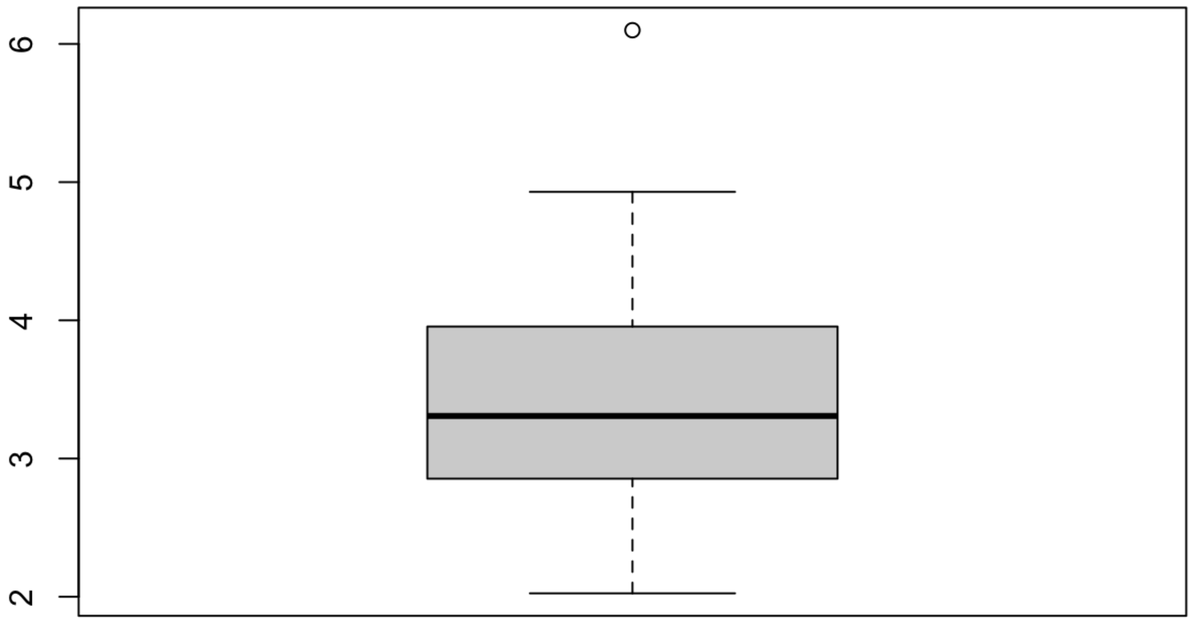
```
qqnorm(diabetic_log)
qqline(diabetic_log)
```



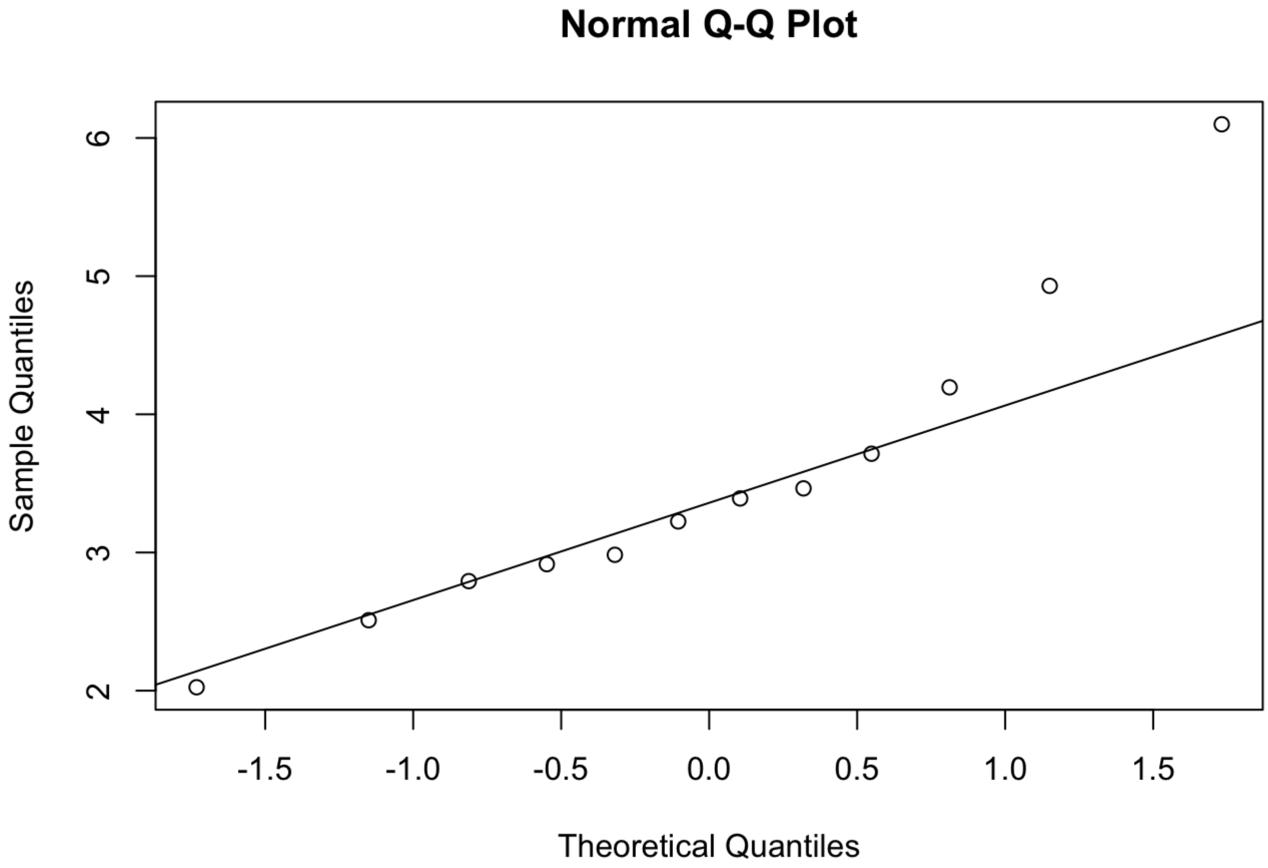
```
plot(density(diabetic_log))
```



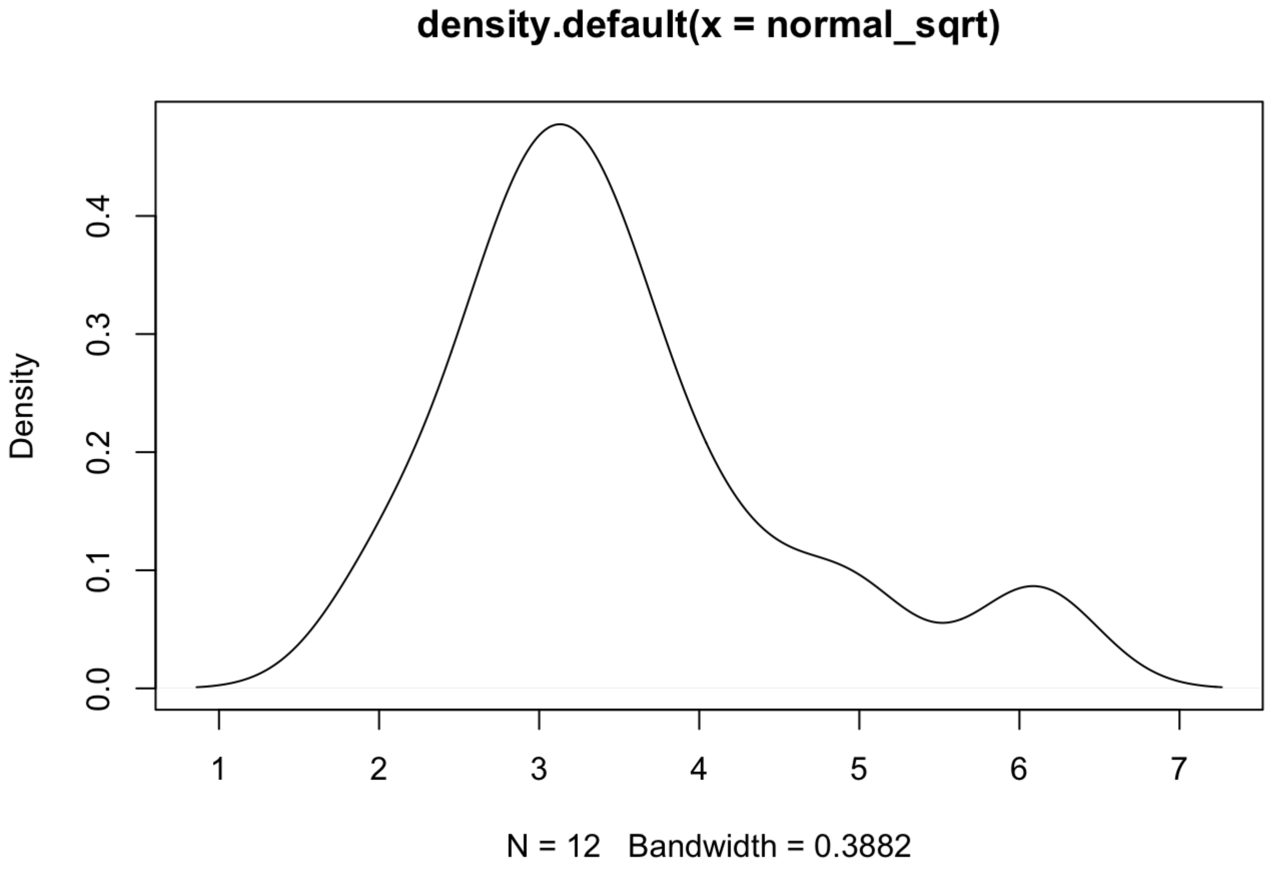
```
# The QQ-plots for the log-transformed samples show points that closely align with the straight diagonal line, suggesting a good fit to the normal distribution.

# normal sample sqrt
boxplot(normal_sqrt)
```



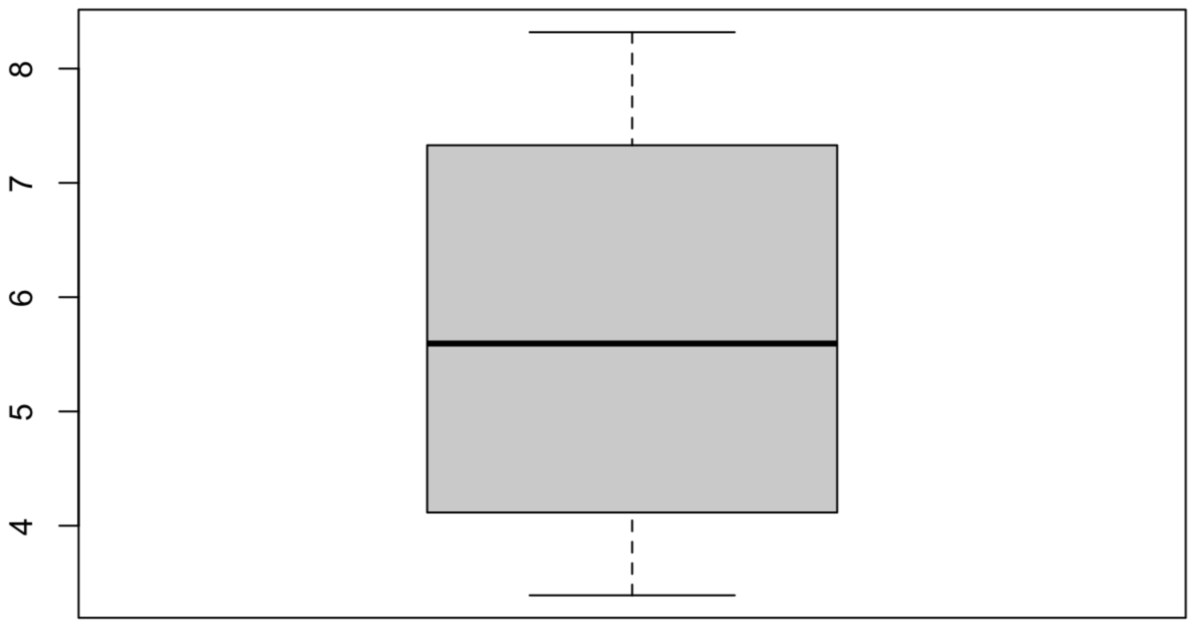
```
qqnorm(normal_sqrt)
qqline(normal_sqrt)
```



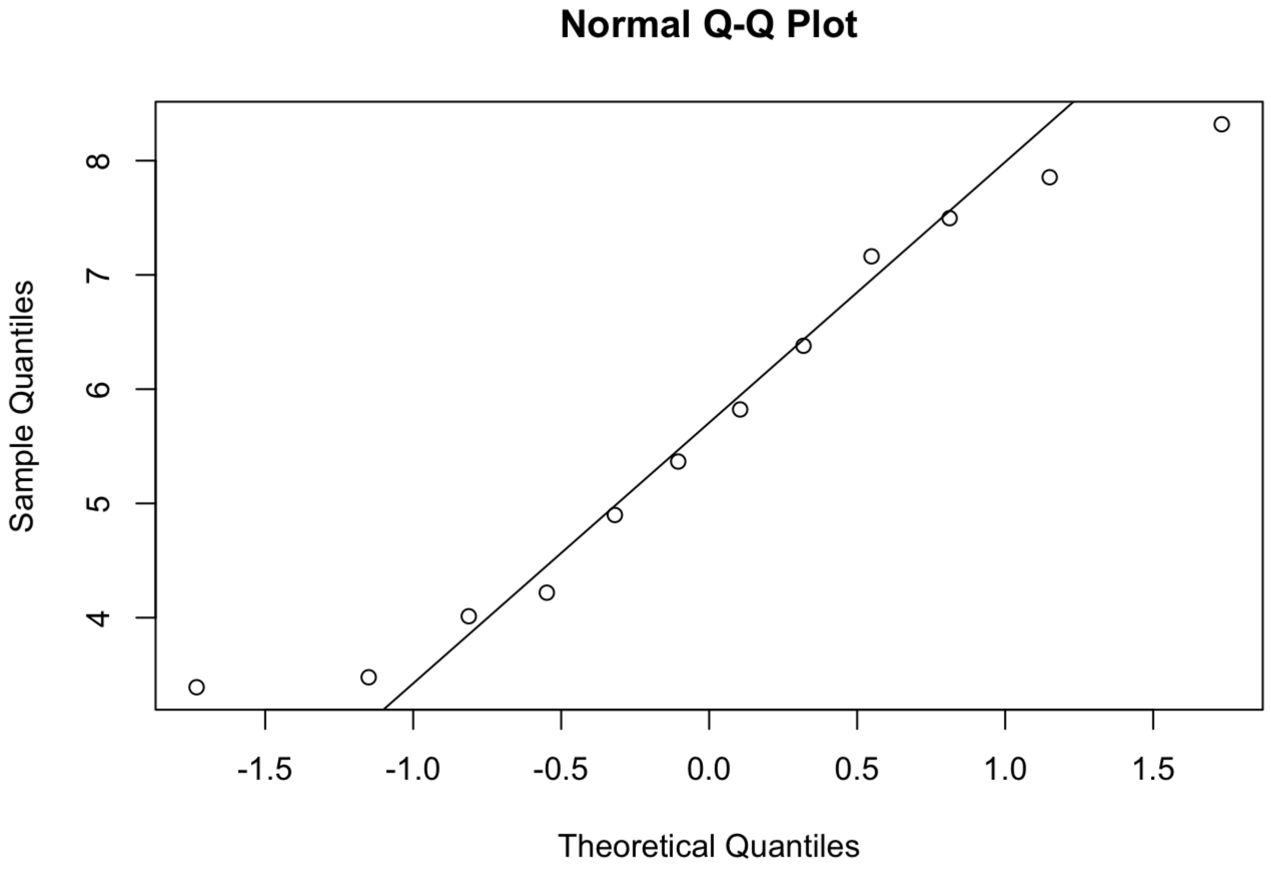
```
plot(density(normal_sqrt))
```



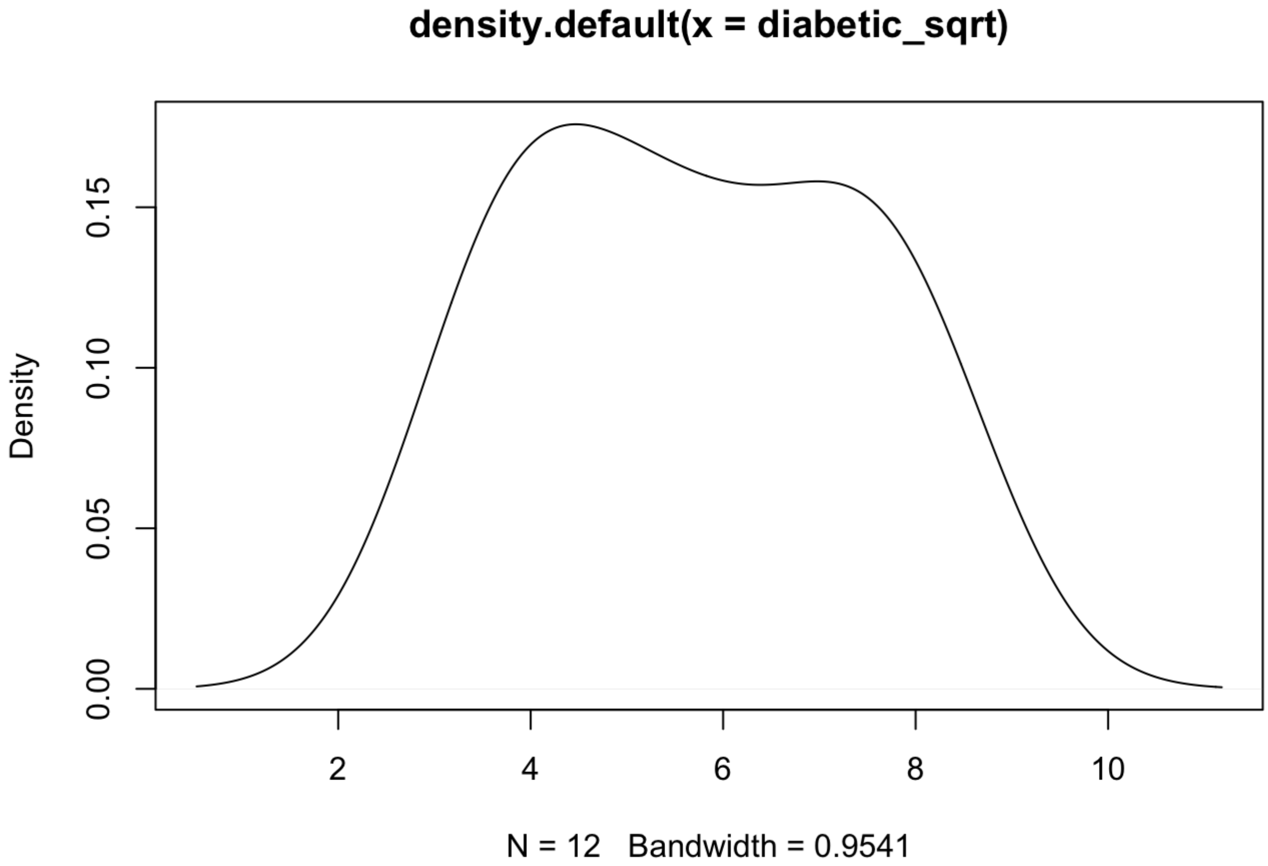
```
# diabetic sample sqrt
boxplot(diabetic_sqrt)
```



```
qqnorm(diabetic_sqrt)
qqline(diabetic_sqrt)
```



```
plot(density(diabetic_sqrt))
```



```
# The plots that compare the quantiles of the square root transformed samples display so
me curvature, particularly in the diabetic sample. This suggests that the transformed me
asurements might still have some departure from normality.
```

c)

```
n = length(normal_log)
mu1 = mean(normal_log)
mu2 = mean(diabetic_log)
std1 = sd(normal_log)
std2 = sd(diabetic_log)
v1 = std1^2
v2 = std1^2
se = sqrt((v1 / n) + (v2/n))
df = (v1+v2)^2/(v1^2/(n-1)+v2^2/(n-1))
t = (mu1 - mu2) / sqrt(v1/n + v2/n)
t
```

```
## [1] -3.939633
```

```
p = 2*pt(t,df)
p
```

```
## [1] 0.0006986983
```

d)

```
# Calculate the observed difference in means

#library(infer)
#dataframe = data.frame(group = c(rep("Normal", 12), rep("Diabetic", 12)), values = log
(c(diabetic, normal)))
#null_dist = dataframe %>%
#  specify(values ~ group) %>%
#  hypothesize(null = "independence") %>%
#  generate(reps = 1000, type = "permute") %>%
#  calculate(stat = "diff in means", order = c("diabetic", "normal"))
#d_hat = data.patients %>%
#  specify(patients ~ group) %>%
#  calculate(stat = "diff in means", order = c("diabetic", "normal"))
#d_hat


# The resulting p-value is approximately 0, which is less than the commonly used
# threshold for statistical significance of 0.05.
# Therefore, we can conclude that the evidence strongly supports the researchers' claim.
```

Q3

```
# Referred - S520_040423_notes_2-sample_problems.pdf
# Referred - S520_040623_2-sample_examples_updated.R
```

Utilize the bechdel dataset from the fivethirtyeight package. Assume that the dataset represents the entire population and collect a sample of 60 movies that passed the Bechdel test and another sample of 72 movies that failed the Bechdel test. Prior to obtaining the samples, set the seed to 100. The objective is to determine whether movies that fail the Bechdel test are more profitable, where profit is defined as domgross - budget. 1. Retrieve the R code to obtain the necessary data. 2. Determine whether this is a one-sample or two-sample test, identify the parameter(s) of interest, and state the hypotheses. 3. Perform the test, calculate the test statistic, p-value, and draw a conclusion using the theory-based method. 4. Repeat step 3 using the simulation-based approach. 5. Use the theory-based method to obtain a 97% confidence interval for the average difference in profit (pass - fail). 6. Repeat step 5 using the simulation-based approach. Here is some code to get you started

```
library(fivethirtyeight)
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✔ ggplot2 3.4.0     ✔ purrr   1.0.1
## ✔ tibble  3.1.8     ✔ dplyr   1.1.0
## ✔ tidyr   1.2.1     ✔ stringr 1.5.0
## ✔ readr   2.1.3     ✔ forcats 1.0.0
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(infer)

#View(bechdel)

set.seed(100)

# 1)
# A sample of 60 movies that passed the Bechdel test

df_pass <- bechdel |>
  na.omit() |>
  filter(binary == "PASS") |>
  slice_sample(n = 60) |>
  mutate(profit = domgross - budget)

# sample of 72 movies that failed the Bechdel test

df_fail <- bechdel |>
  na.omit() |>
  filter(binary == "FAIL") |>
  slice_sample(n = 72)|>
  mutate(profit = domgross - budget)

df_final <- rbind(df_pass, df_fail)
```

2. In this study, the unit of analysis is a movie. The researcher samples movies from two distinct populations: one population consists of movies that have passed the Bechdel test, while the other population consists of movies that have failed the Bechdel test. Since the researcher is comparing two populations, this is a two-sample problem. The variable of interest is whether or not a movie passed the Bechdel test, which serves as a binary measurement for each movie. To clarify, let X_i represent the ith movie that passed the Bechdel test, while Y_j represents the jth movie that failed the Bechdel test. The parameter of interest is the difference between the means of the two populations, denoted by delta = mu_1 - mu_2. The null hypothesis for this study is that there is no difference between the means of the two populations. More specifically, the null hypothesis can be expressed as H_0: delta = 0, while the alternative hypothesis is H_1: delta != 0.
3. Theory Approach-

```
# X_i = mu_1 for i = 1,...,60
# Y_j = mu_2 for j = 1,...,72

t_test <- df_final |>
  t_test(formula = profit ~ binary, order = c("PASS", "FAIL"))

test_statistic <- t_test$statistic
test_statistic
```

```
##        t
## 1.553952
```

```
p_value <- t_test$p_value
p_value
```

```
## [1] 0.1226952
```

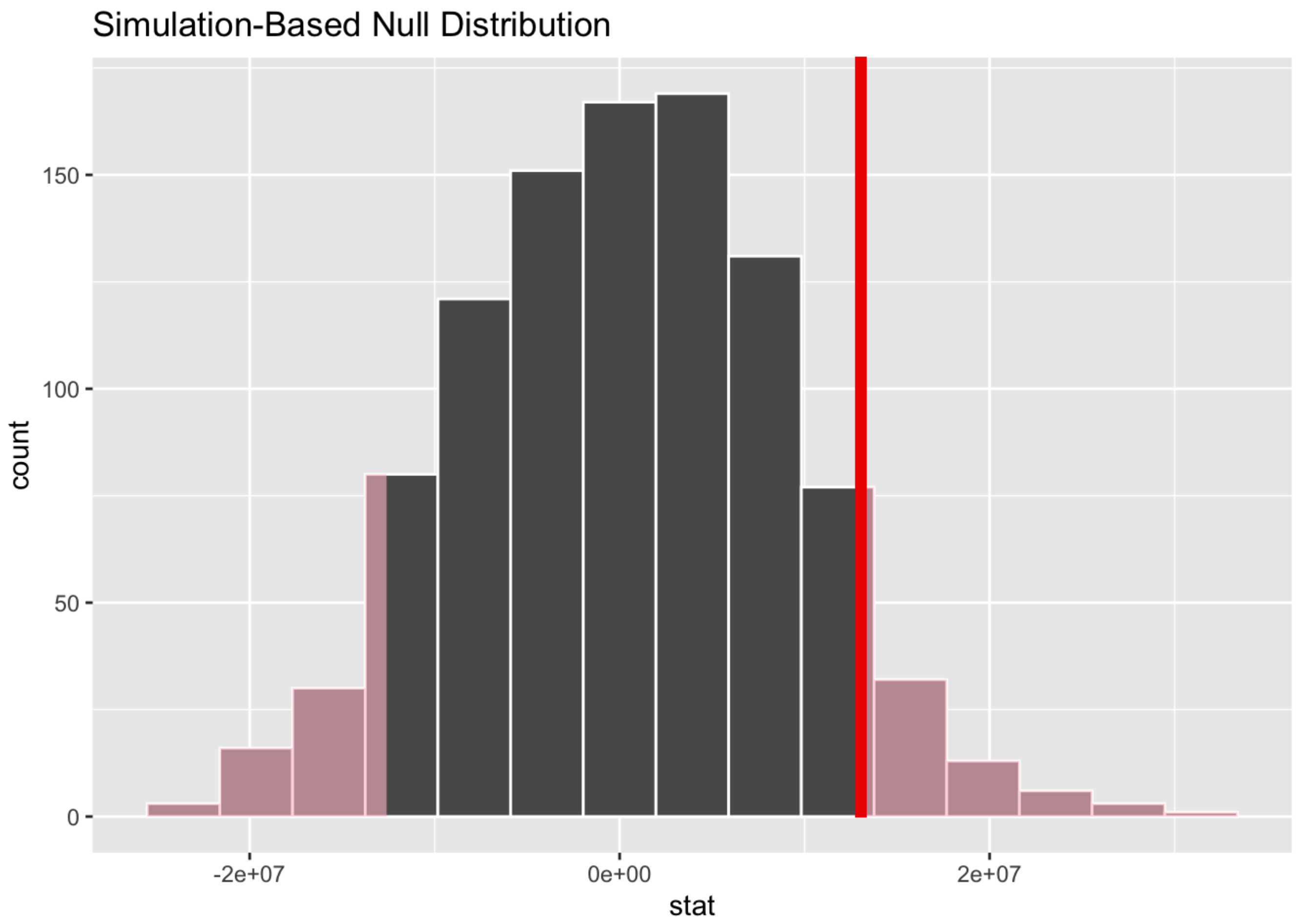
```
# Based on the obtained p-value of 0.122 at a significance level of 0.05, we cannot reje
ct the null hypothesis. Hence, there is insufficient evidence to support the claim that
the average profit of movies that do not pass the Bechdel test is higher than the averag
e profit of movies that pass the Bechdel test. movies that pass the Bechdel test.
```

4. Simulation approach

```
d_hat <- df_final |>
  specify(profit ~ binary) |>
  calculate(stat = "diff in means", order = c("PASS", "FAIL"))

d_hat
```

```
## Response: profit (numeric)
## Explanatory: binary (factor)
## # A tibble: 1 × 1
##        stat
##       <dbl>
## 1 13044949.
```

```
null_dist <- df_final |>
  specify(profit ~ binary) |>
  hypothesize(null = "independence") |>
  generate(reps = 1000, type = "permute") |>
  calculate(stat = "diff in means", order = c("PASS", "FAIL"))

visualize(null_dist) +
  shade_p_value(obs_stat = d_hat, direction = "two-sided")
```



```
null_dist |>
  get_p_value(obs_stat = d_hat, direction = "two-sided")
```

```
## # A tibble: 1 × 1
##   p_value
##     <dbl>
## 1    0.13
```

```
# 5) Theory based CI = 97%
x_bar =  mean(df_pass$profit)
y_bar = mean(df_fail$profit)
s_1 = sd(df_pass$profit)
s_2 = sd(df_fail$profit)
n_1 = length(df_pass$profit)
n_2 = length(df_fail$profit)
se = sqrt(s_1^2/n_1 + s_2^2/n_2)
se
```

```
## [1] 8394691
```

```
t_stat = (x_bar - y_bar - 0)/se
t_stat
```

```
## [1] 1.553952
```

```
nu.hat = (s_1^2/n_1+s_2^2/n_2)^2/((s_1^2/n_1)^2/(n_1-1) + (s_2^2/n_2)^2/(n_2-1))
nu.hat
```

```
## [1] 126.4454
```

```
alpha = 1- 0.97
q = qnorm(1-alpha/2)

c(nu.hat-q*se, nu.hat+q*se)
```

```
## [1] -18217112  18217365
```

```
# 6) Simulation based CI = 97%

library(infer)

null_sim <-  df_final |>
  specify(profit ~ binary) |>
  #hypothesize(null = "point", mu = ) |>
  generate(reps = 1000, type = "bootstrap") |>
  calculate(stat = "diff in means", order = c("PASS", "FAIL"))
percentile_ci <- get_ci(null_sim, level = 0.97)
percentile_ci
```

```
## # A tibble: 1 × 2
##    lower_ci  upper_ci
##       <dbl>     <dbl>
## 1 -4506540. 30857010.
```