

# Homework2-Exam 1 Preparation HW

Homework02Gr6

2025-02-20

**NOTE:** only those who contributes and fully participates in the work will get credit

**Scribe:**

**Moderator:**

**All contributors:**

This homework is accumulative, with the intention of assisting you to prepare for Exam1. It is due Wednesday 11:59pm, you will get extra 10pts for this HW if you finish it by Tuesday 11:59pm. Yes, I know it has lots of problems, but I believe that you want to perform well for the coming exam, so this HW will be very beneficial!

**Attention:** Please finish all 14 problems step by step, rather than using the advance functions `t.test`, `prop.test`. However, you are encouraged to use more advance functions to check your solutions.

##Q1 Devore 2nd edition 7.1.5

*#Given standard deviation*

`sigma <- 0.75`

*# (a) Compute 95% CI for true average porosity (n = 20, sample mean = 4.85)*

`n1 <- 20`

`xbar1 <- 4.85`

`alpha1 <- 0.05` *# 95% CI*

`z1 <- qnorm(1 - alpha1/2)` *# Z-score for 95% CI*

`margin_error1 <- z1 * (sigma / sqrt(n1))`

`CI_95 <- c(xbar1 - margin_error1, xbar1 + margin_error1)`

`print("95% Confidence Interval is:")`

## [1] "95% Confidence Interval is:"

CI\_95

```
## [1] 4.521304 5.178696

## (b) Compute 98% CI for another seam

n2 <- 16
xbar2 <- 4.56
alpha2 <- 0.02 # 98% CI
z2 <- qnorm(1 - alpha2/2) # Z-score for 98% CI

margin_error2 <- z2 * (sigma / sqrt(n2))
CI_98 <- c(xbar2 - margin_error2, xbar2 + margin_error2)
print("98% Confidence Interval is:")

## [1] "98% Confidence Interval is:"

CI_98

## [1] 4.12381 4.99619

## (c) Sample Size for 95% CI Width of 0.40
width <- 0.40 # Total width of CI
margin_error_target <- width / 2 # Half-width

# Solve for sample size:  $n = (z \cdot \sigma / \text{margin\_error})^2$ 
n_required_95 <- (z1 * sigma / margin_error_target)^2
n_required_95 <- ceiling(n_required_95) # Round up to nearest whole
number
print("Required Sample Size is: ")

## [1] "Required Sample Size is: "

n_required_95

## [1] 55

## (d) Sample Size for 99% CI with Width 0.2

width_99 <- 0.2 # Total width
margin_error_target_99 <- width_99 / 2 # Half-width
alpha3 <- 0.01 # 99% CI
z3 <- qnorm(1 - alpha3/2) # Z-score for 99% CI

n_required_99 <- (z3 * sigma / margin_error_target_99)^2
```

```

n_required_99 <- ceiling(n_required_99) # Round up to nearest whole
number
print("Required Sample Size is: ")

## [1] "Required Sample Size is: "

n_required_99

## [1] 374

## Part (a)

# Given data
xbar <- 654.16 # Sample mean
s <- 164.43 # Sample standard deviation
n <- 50 # Sample size
df <- n - 1 # Degrees of freedom
alpha <- 0.05 # 95% confidence level

## Since the sample standard deviation is given and the population
standard deviation is unknown, we must estimate it using sample
standard distribution. Now we can use normal distribution because the
sample size is 50 (>30) but even for larger n, the t-distribution is
more accurate using s instead of sigma

# Compute critical t-value
critical_t <- qt(1 - alpha/2, df)

# Compute margin of error
margin_error <- critical_t * (s / sqrt(n))

# Compute confidence interval
CI_95 <- c(xbar - margin_error, xbar + margin_error)
CI_95

## [1] 607.4295 700.8905

##Part (b)
sigma_est <- 175 # Estimated standard deviation
ME <- 50 / 2 # Margin of error (half of interval width)
alpha <- 0.05 # 95% confidence level
z_alpha <- qnorm(1 - alpha/2) # Z-score for 95% CI

```

```

# Compute required sample size
n_required <- ( (z_alpha * sigma_est) / ME )^2
n_required <- ceiling(n_required) # Round up to nearest whole number
print("The sample size is : ")

## [1] "The sample size is : "

n_required

## [1] 189

# Given data
phat <- 0.53 # Sample proportion
n <- 2343 # Sample size
alpha <- 0.01 # 99% confidence level
z_alpha <- qnorm(1 - alpha/2) # Critical z-value for 99%

# Compute standard error
SE <- sqrt((phat * (1 - phat)) / n)

# Compute margin of error
ME <- z_alpha * SE

# Compute confidence interval
CI_99 <- c(phat - ME, phat + ME)
print("99% confidence interval is :")

## [1] "99% confidence interval is :"

CI_99

## [1] 0.5034406 0.5565594

## Sample Size Calculation for CI Width ≤ 0.05
# Desired margin of error
ME_target <- 0.05 / 2 # Half of the total width
p_star <- 0.5 # Conservative estimate for p

# Compute required sample size
n_required <- ( (z_alpha / ME_target)^2 ) * (p_star * (1 - p_star))
n_required <- ceiling(n_required) # Round up
print("Required sample size is :")

```

```

## [1] "Required sample size is :"

n_required

## [1] 2654

# Given values
# Given data
mean_x <- 370.69
s <- 24.36
n <- 26
alpha <- 0.05 # For a 95% confidence level

# Degrees of freedom
df <- n - 1

# t-value for 95% confidence level
t_value <- qt(1 - alpha, df)

# Calculate the upper confidence bound
upper_conf_bound <- mean_x + t_value * (s / sqrt(n))
upper_conf_bound

## [1] 378.8505

##Part (b)
# Calculate the upper prediction bound for a single worker
upper_pred_bound <- mean_x + t_value * sqrt(s^2 * (1 + 1/n))
upper_pred_bound

## [1] 413.093

##Part (c)
# Calculate the prediction interval for the average of two new workers
t_value_new <- qt(1 - alpha/2, n + 1)

# Calculate the 95% two-sided interval
lower_pred_interval <- mean_x - t_value_new * (s / sqrt(2))
upper_pred_interval <- mean_x + t_value_new * (s / sqrt(2))

c(lower_pred_interval, upper_pred_interval)

## [1] 335.347 406.033

```

### Devore problem D8.1.13

Part (a) : We want to test if the scale is calibrated correctly, meaning if the true average weight reading is equal to 10 kg. The hypotheses are:

- Null Hypothesis ( $H_0$ ):  $m = 10$  kg (the scale is correctly calibrated)
- Alternative Hypothesis ( $H_A$ ):  $m \neq 10$  kg (the scale is not correctly calibrated)

```
# Given data
x_bar <- 9.85 # sample mean
mu_0 <- 10   # hypothesized population mean
sigma <- 0.200 # population standard deviation
n <- 25      # sample size

# Calculate Z score
z_score <- (x_bar - mu_0) / (sigma / sqrt(n))

# Calculate P-value for two-tailed test
p_value <- 2 * (1 - pnorm(abs(z_score)))
p_value

## [1] 0.0001768346
```

At a significance level of 0.01, we reject the null hypothesis.

```
# New true means
m1 <- 10.1
m2 <- 9.8

# Z-scores for m1 and m2
z_score_m1 <- (m1 - mu_0) / (sigma / sqrt(n))
z_score_m2 <- (m2 - mu_0) / (sigma / sqrt(n))

# Calculate the probability of not rejecting the null hypothesis for
m1 and m2
prob_m1 <- pnorm(abs(z_score_m1)) - pnorm(-abs(z_score_m1))
prob_m2 <- pnorm(abs(z_score_m2)) - pnorm(-abs(z_score_m2))

c(prob_m1, prob_m2)

## [1] 0.9875807 0.9999994
```

```

# Given data
x_bar <- 5.25 # sample mean
mu_0 <- 5.5   # hypothesized population mean
s <- 0.3      # population standard deviation
n <- 16       # sample size

# Calculate the t-statistic
t_stat <- (x_bar - mu_0) / (s / sqrt(n))

# Degrees of freedom
df <- n - 1

# Calculate the p-value for a two-tailed test
p_value <- 2 * (1 - pt(abs(t_stat), df))
t_stat

## [1] -3.333333

p_value

## [1] 0.004537856

#Significance level
alpha <- 0.01

if (p_value < alpha) {
  conclusion <- "Reject H0: True average differs from 5.5"
} else {
  conclusion <- "Fail to reject H0: True average doesn't differ from
5.5"
}
conclusion

## [1] "Reject H0: True average differs from 5.5"

#Part (b)
# New true mean
mu_true <- 5.6

# Calculate the non-centrality parameter
delta <- (mu_true - mu_0) / (s / sqrt(n))

```

```

# Calculate the power of the test (probability of detecting the
departure)
power <- 1 - pt(qt(1 - alpha/2, df), df, ncp = delta) + pt(qt(alpha/2,
df), df, ncp = delta)
power

## [1] 0.08331193

#Part (c)
# Desired power and significance level
desired_power <- 0.99
alpha <- 0.01

# Effect size (d) for delta
d <- (mu_true - mu_0) / (s / sqrt(n))

# Calculate the required sample size for given power

library(pwr)
sample_size <- pwr.t.test(d = d, sig.level = alpha, power =
desired_power, type = "two.sample")$n
ceiling (sample_size)

## [1] 29

# Given data
distances <- c(32.1, 30.6, 31.4, 30.4, 31.0, 31.9)
n <- length(distances)
x_bar <- mean(distances) # Sample mean
mu_0 <- 30 # Hypothesized population mean
s <- sd(distances) # Sample standard deviation
alpha <- 0.01 # Significance level

# Calculate the t-statistic
t_stat <- (x_bar - mu_0) / (s / sqrt(n))

# Degrees of freedom
df <- n - 1

# Calculate the p-value for a one-tailed test
p_value <- 1 - pt(t_stat, df)
t_stat

```



```

## [1] 4.384921

p_value

## [1] 0.003560841

if (p_value < alpha) {
  conclusion <- "Reject H0: True average stopping distance is greater
than 30ft."
} else {
  conclusion <- "Fail to reject H0: True average stopping distance is
equal to 30ft."
}
conclusion

## [1] "Reject H0: True average stopping distance is greater than
30ft."

## Part (b)
# Actual means for Type II error calculation
mu_true_31 <- 31
mu_true_32 <- 32

# Non-centrality parameter for each mean
delta_31 <- (mu_true_31 - mu_0) / (s / sqrt(n))
delta_32 <- (mu_true_32 - mu_0) / (s / sqrt(n))

# Calculate the Type II error probabilities (1 - power) for each mean
beta_31 <- pt(qt(1 - alpha, df), df, ncp = delta_31)
beta_32 <- pt(qt(1 - alpha, df), df, ncp = delta_32)

c(beta_31, beta_32)

## [1] 0.395046365 0.004696065

## Part (c)

# New sample standard deviation
s_new <- 0.80

# Non-centrality parameters for the new standard deviation
delta_31_new <- (mu_true_31 - mu_0) / (s_new / sqrt(n))
delta_32_new <- (mu_true_32 - mu_0) / (s_new / sqrt(n))

```

```

# Calculate the Type II error probabilities for the new standard
deviation
beta_31_new <- pt(qt(1 - alpha, df), df, ncp = delta_31_new)
beta_32_new <- pt(qt(1 - alpha, df), df, ncp = delta_32_new)

c(beta_31_new, beta_32_new)

## [1] 0.52980978 0.02451232

## Part (d)

# Desired power and Type II error probability
desired_power <- 0.90
alpha <- 0.01

# Effect size (d) for delta when true mean is 31
delta_size <- (mu_true_31 - mu_0) / (0.65)

# Use the pwr.t.test function to calculate the required sample size
library(pwr)
sample_size <- pwr.t.test(d = delta_size, sig.level = alpha, power =
desired_power, type = "one.sample")$n
ceiling(sample_size)

## [1] 10

# Given data
n <- 51 # sample size
x <- 41 # number of homes with problems
p_hat <- x / n # sample proportion
p_0 <- 0.50 # hypothesized population proportion
alpha <- 0.01 # significance level

# Calculate the z-statistic
z_stat <- (p_hat - p_0) / sqrt((p_0 * (1 - p_0)) / n)

# Calculate the p-value for a one-tailed test
p_value <- 1 - pnorm(z_stat)
z_stat

## [1] 4.340868

```

```

p_value
## [1] 7.09604e-06

if (p_value < alpha) {
  conclusion <- "Reject H0: More than 50% of homes with Chinese
drywall have electrical/environmental problems.."
} else {
  conclusion <- "Fail to reject H0: 50% of homes with Chinese drywall
have electrical/environmental problems"
}
conclusion

## [1] "Reject H0: More than 50% of homes with Chinese drywall have
electrical/environmental problems.."

## Part (b)
# Z-score for 99% confidence level
z_alpha <- qnorm(1 - alpha)

# Calculate the lower confidence bound
lower_conf_bound <- p_hat - z_alpha * sqrt((p_hat * (1 - p_hat)) / n)
lower_conf_bound

## [1] 0.674588

## Part (c)
# Actual true proportion for Type II error calculation
p_true <- 0.80

# Calculate the non-centrality parameter (delta)
delta <- (p_true - p_0) / sqrt((p_0 * (1 - p_0)) / n)

# Calculate the probability of failing to reject H0 (Type II error)
beta <- pnorm(qnorm(1 - alpha) - delta)
beta

## [1] 0.02508515

# Given data
x_bar <- 18.12 # Sample mean for modified mortar
y_bar <- 16.87 # Sample mean for unmodified mortar
s1 <- 1.6 # Standard deviation for modified mortar

```

```

s2 <- 1.4 # Standard deviation for unmodified mortar
m <- 40   # Sample size for modified mortar
n <- 32   # Sample size for unmodified mortar
alpha <- 0.01 # Significance level

# Calculate the z-statistic for the difference in means
z_stat <- (x_bar - y_bar) / sqrt((s1^2 / m) + (s2^2 / n))

# Calculate the p-value for a one-tailed test
p_value <- 1 - pnorm(z_stat)
z_stat

## [1] 3.532004

p_value

## [1] 0.0002062118

if (p_value < alpha) {
  conclusion <- "Reject H0: Modified mortar has a higher average
tension bond strength than unmodified mortar"
} else {
  conclusion <- "Fail to reject H0: Unmodifier mortar has a higher
average tension bond strength than unmodified mortar"
}
conclusion

## [1] "Reject H0: Modified mortar has a higher average tension bond
strength than unmodified mortar"

##Part (b)
# True difference in means for Type II error calculation
true_diff <- 1

# Non-centrality parameter
delta <- true_diff / sqrt((s1^2 / m) + (s2^2 / n))

# Calculate the Type II error probability (beta)
beta <- pnorm(qnorm(1 - alpha) - delta)
beta

## [1] 0.3087999

```

```

##Part (c)

# Desired power
desired_power <- 0.90
alpha_new <- 0.05

# Effective size (d) for delta when true difference is 1
d <- true_diff / sqrt((s1^2 / m) + (s2^2 / n))

# Use the pwr package to calculate the required sample size
library(pwr)
sample_size <- pwr.t.test(d = d, sig.level = alpha_new, power =
desired_power, type = "two.sample")$n
ceiling (sample_size)

## [1] 4

## part (d)
# Pooled standard deviation (when s1 and s2 are unknown)
s_pooled <- sqrt(((m - 1) * s1^2 + (n - 1) * s2^2) / (m + n - 2))

# Calculate the t-statistic for the test
t_stat_unknown <- (x_bar - y_bar) / (s_pooled * sqrt(1/m + 1/n))

# Degrees of freedom
df_unknown <- m + n - 2

# Calculate the p-value for a one-tailed t-test
p_value_unknown <- 1 - pt(t_stat_unknown, df_unknown)
t_stat_unknown

## [1] 3.479565

p_value_unknown

## [1] 0.0004342387

##Data

# Blackbirds Experimental Location
t_blackbirds_expt1 <- 13.4
se_blackbirds_expt1 <- 2.05

```

```

n_blackbirds_exptl <- 65

# Blackbirds Natural Location
t_blackbirds_natural <- 9.7
se_blackbirds_natural <- 1.76
n_blackbirds_natural <- 50

#Silvereyes Experimental location
t_silvereyes_exptl <- 49.4
se_silvereyes_exptl <- 4.78
n_silvereyes_exptl <- 34

# Silvereyes Natural Location
t_silvereyes_natural <- 38.4
se_silvereyes_natural <- 5.06
n_silvereyes_natural <- 46

#Part a
#Upper Confidence Bound for Blackbirds at Experimental Location

library(stats)
alpha <- 0.05 # 95% confidence level
t_crit <- qt(1 - alpha, df = n_blackbirds_exptl - 1)
upper_bound <- t_blackbirds_exptl + t_crit * se_blackbirds_exptl
upper_bound

## [1] 16.82148

#Part b
#Hypothesis Test: Does Blackbirds' Time at Experimental Location
Exceed Natural Location?

# Hypotheses
# H0:  $\mu_{\text{blackbirds\_exptl}} \leq \mu_{\text{blackbirds\_natural}}$ 
# HA:  $\mu_{\text{blackbirds\_exptl}} > \mu_{\text{blackbirds\_natural}}$ 

t_stat <- (t_blackbirds_exptl - t_blackbirds_natural) /
sqrt(se_blackbirds_exptl^2/n_blackbirds_exptl +
se_blackbirds_natural^2/n_blackbirds_natural)
df <- min(n_blackbirds_exptl - 1, n_blackbirds_natural - 1)

```

```

p_value <- 1 - pt(t_stat, df)
print ("t_statistic && p_value: ")

## [1] "t_statistic && p_value: "

list(t_statistic = t_stat, p_value = p_value)

## $t_statistic
## [1] 10.3986
##
## $p_value
## [1] 2.708944e-14

#Part (c)
#Confidence Interval for Difference Between Silvereyes and Blackbirds
at Natural Location
diff_means <- t_silvereyes_natural - t_blackbirds_natural
se_diff <- sqrt(se_silvereyes_natural^2/n_silvereyes_natural +
se_blackbirds_natural^2/n_blackbirds_natural)
t_crit_diff <- qt(1 - alpha/2, df = min(n_silvereyes_natural - 1,
n_blackbirds_natural - 1))
ci_lower <- diff_means - t_crit_diff * se_diff
ci_upper <- diff_means + t_crit_diff * se_diff

#Confidence interval
list(mean_difference = diff_means, ci_lower = ci_lower, ci_upper =
ci_upper)

## $mean_difference
## [1] 28.7
##
## $ci_lower
## [1] 27.11595
##
## $ci_upper
## [1] 30.28405

#Point Estimate with Standard Error for Difference Between Silvereyes
and Blackbirds at Natural Location
diff_means <- t_silvereyes_natural - t_blackbirds_natural
se_diff <- sqrt(se_silvereyes_natural^2/n_silvereyes_natural +
se_blackbirds_natural^2/n_blackbirds_natural)

```

```

list(mean_difference = diff_means, standard_error = se_diff)

## $mean_difference
## [1] 28.7
##
## $standard_error
## [1] 0.7864808

## Data Entry

# Lactation (L) and Postweaning (P) TBBMC data
L <- c(1928, 2549, 2825, 1924, 1628, 2175, 2114, 2621, 1843, 2541)
P <- c(2126, 2885, 2895, 1942, 1750, 2184, 2164, 2626, 2006, 2627)
diff <- P - L # Differences (P - L)
n <- length(diff)

#Part (a)

## (a) Hypothesis Test: Does Postweaning TBBMC Exceed Lactation by
More Than 25g?

library(stats)
alpha <- 0.05
mu_0 <- 25 # Hypothesized difference
x_bar <- mean(diff)
s <- sd(diff)
t_stat <- (x_bar - mu_0) / (s / sqrt(n))
df <- n - 1
p_value <- 1 - pt(t_stat, df)

list(mean_difference = x_bar, t_statistic = t_stat, p_value = p_value)

## $mean_difference
## [1] 105.7
##
## $t_statistic
## [1] 2.457468
##
## $p_value
## [1] 0.01815458

```



```

if (p_value < alpha) {
  conclusion <- "Reject H0: There is significant evidence that the
true average TBBMC postweaning exceeds lactation by more than 25g."
} else {
  conclusion <- "Fail to reject H0: We do not have strong enough
evidence to claim that the increase is more than 25g"
}
conclusion

## [1] "Reject H0: There is significant evidence that the true average
TBBMC postweaning exceeds lactation by more than 25g."

## (b) Upper Confidence Bound for True Average Difference
conf_level <- 0.95
t_crit <- qt(conf_level, df)
upper_bound <- x_bar + t_crit * (s / sqrt(n))
upper_bound

## [1] 165.897

## (c) Two-Sample t Test Comparison
t_test_result <- t.test(P, L, var.equal = TRUE, alternative =
"greater")
t_test_result

##
## Two Sample t-test
##
## data: P and L
## t = 0.58872, df = 18, p-value = 0.2817
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
## -205.6358 Inf
## sample estimates:
## mean of x mean of y
## 2320.5 2214.8

### Explanation:
print ("The paired t-test in (a) takes into account the natural
pairing of measurements within subjects, whereas the two-sample t-test
assumes independent samples. From two sample test, it leads to
conclusion that we fail to reject Null Hypothesis since p_value is

```

higher than alpha. The reason maybe due to the loss of paired structure, affecting variability and statistical power.")

```
## [1] "The paired t-test in (a) takes into account the natural pairing of measurements within subjects, whereas the two-sample t-test assumes independent samples. From two sample test, it leads to conclusion that we fail to reject Null Hypothesis since p_value is higher than alpha. The reason maybe due to the loss of paired structure, affecting variability and statistical power."
```

```
#Data
```

```
# Observed data
```

```
n_perdue <- 80
```

```
contaminated_perdue <- 35
```

```
non_contaminated_perdue <- n_perdue - contaminated_perdue
```

```
n_tyson <- 80
```

```
contaminated_tyson <- 66
```

```
non_contaminated_tyson <- n_tyson - contaminated_tyson
```

```
# Sample proportions
```

```
p_hat_perdue <- non_contaminated_perdue / n_perdue
```

```
p_hat_tyson <- non_contaminated_tyson / n_tyson
```

```
## (a) Hypothesis Test: Difference in True Proportion of Non-Contaminated Broilers
```

```
library(stats)
```

```
# H0: p_perdue = p_tyson
```

```
# HA: p_perdue ≠ p_tyson
```

```
p_hat_diff <- p_hat_perdue - p_hat_tyson
```

```
p_pool <- (non_contaminated_perdue + non_contaminated_tyson) /  
(n_perdue + n_tyson)
```

```
se_pool <- sqrt(p_pool * (1 - p_pool) * (1/n_perdue + 1/n_tyson))
```

```
z_stat <- p_hat_diff / se_pool
```

```
p_value <- 2 * (1 - pnorm(abs(z_stat)))
```

```
list(z_statistic = z_stat, p_value = p_value)
```

```

## $z_statistic
## [1] 5.079664
##
## $p_value
## [1] 3.781029e-07

alpha <- 0.01 #significance level
if (p_value < alpha) {
  conclusion <- "Reject H0: True proportion of non- contaminated
Perdue broilers differs from that for the Tyson brand."
} else {
  conclusion <- "Fail to reject H0: True proportion of non-
contaminated Perdue broilers does not differ from that for the Tyson
brand"
}
conclusion

## [1] "Reject H0: True proportion of non- contaminated Perdue
broilers differs from that for the Tyson brand."

## (b) Power Analysis: Probability of Rejecting H0 When True
Proportions Are 0.50 and 0.25

# Given true proportions
p_perdue_true <- 0.50
p_tyson_true <- 0.25
p_diff_true <- p_perdue_true - p_tyson_true

se_true <- sqrt((p_perdue_true * (1 - p_perdue_true) / n_perdue) +
                (p_tyson_true * (1 - p_tyson_true) / n_tyson))

z_beta <- (p_diff_true / se_true) - qnorm(1 - 0.01/2) # Two-tailed
test at alpha = 0.01
power <- 1 - pnorm(z_beta)

list(power = power)

## $power
## [1] 0.2104711

# Given data
s_control <- 32 # Standard deviation for control group

```

```

n_control <- 23 # Sample size for control group

s_low_dose <- 54 # Standard deviation for low-dose group
n_low_dose <- 20 # Sample size for low-dose group

library(stats)
# H0:  $\sigma^2_{\text{low\_dose}} = \sigma^2_{\text{control}}$  (Equal variances)
# HA:  $\sigma^2_{\text{low\_dose}} > \sigma^2_{\text{control}}$  (Greater variability in low-
dose group)

f_stat <- (s_low_dose^2) / (s_control^2)
df_num <- n_low_dose - 1
df_den <- n_control - 1
p_value <- 1 - pf(f_stat, df_num, df_den)

list(f_statistic = f_stat, p_value = p_value)

## $f_statistic
## [1] 2.847656
##
## $p_value
## [1] 0.01008296

if (p_value < 0.05) {
  conclusion <- "Reject H0: There is significant evidence that the
low-dose group has greater variability in weight gain"
} else {
  conclusion <- "Fail to reject H0: No significant evidence of greater
variability in the low-dose group "
}
conclusion

## [1] "Reject H0: There is significant evidence that the low-dose
group has greater variability in weight gain"

# Given data
s_cotton <- 0.79 # Standard deviation for Cotton fabric
n_cotton <- 10 # Sample size for Cotton fabric

s_triacetate <- 3.59 # Standard deviation for Triacetate fabric
n_triacetate <- 10 # Sample size for Triacetate fabric

```

### ***## 95% Upper Confidence Bound for the Ratio of Standard Deviations***

```
library(stats)
alpha <- 0.05 # 95% confidence level
f_stat <- (s_triacetate^2) / (s_cotton^2)
df_num <- n_triacetate - 1
df_den <- n_cotton - 1

f_crit <- qf(1 - alpha, df_num, df_den)
upper_bound <- f_stat / f_crit

list(f_statistic = f_stat, upper_confidence_bound = upper_bound)

## $f_statistic
## [1] 20.6507
##
## $upper_confidence_bound
## [1] 6.496191
```