Data Science in Life Science - Worksheet 01 Group 1111

2024-04-21

Question 1

Load library and constant parameters

```
library(ggplot2)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.5
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ lubridate 1.9.3     ✔ tibble    3.2.1
## ✔ purrr     1.0.2     ✔ tidyr     1.3.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(stats)
library(extraDistr)
```

```
##
## Attaching package: 'extraDistr'
##
## The following object is masked from 'package:purrr':
##
##     rdunif
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

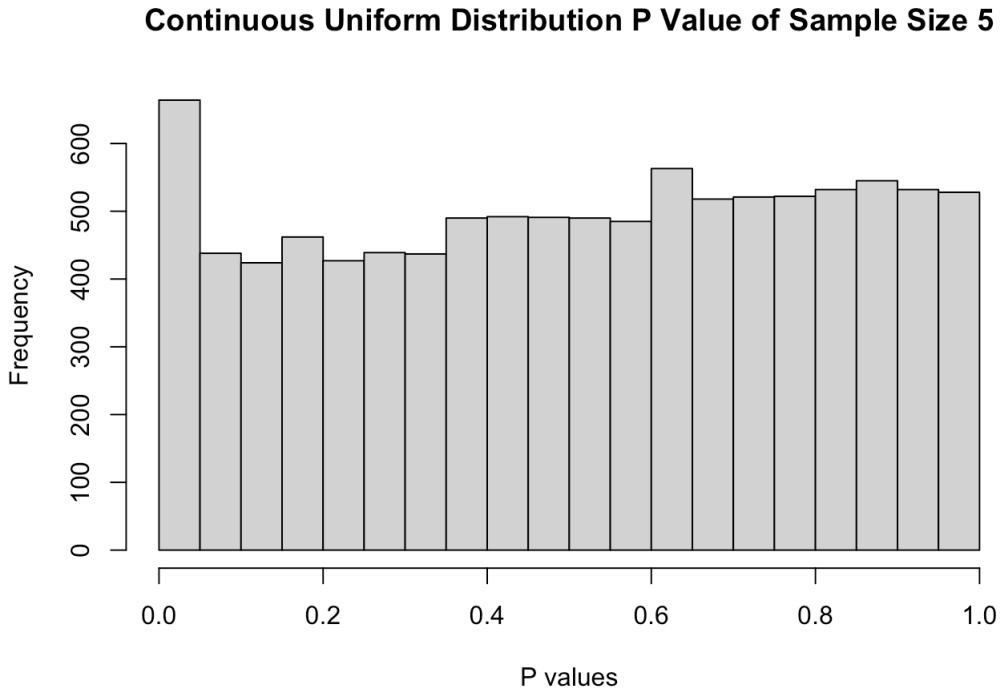
```
library(reshape)
```

```
##
## Attaching package: 'reshape'
##
## The following objects are masked from 'package:reshape2':
##
##     colsplit, melt, recast
##
## The following object is masked from 'package:lubridate':
##
##     stamp
##
## The following object is masked from 'package:dplyr':
##
##     rename
##
## The following objects are masked from 'package:tidyr':
##
##     expand, smiths
```

```
# create a vector to hold the list of sample sizes
sample_sizes <- c(5, 10, 20, 50, 100)

# number of times to repeat sample
n_rounds <- 10000

# set alpha level for two-tailed (0.05 / 2)
sig_level <- 0.025
```

sample from continuous uniform distribution

```
# function for sampling from continuous uniform distribution [-1,1]
continuous_uniform_dist <- function(i){ # where i = # samples drawn
  t.test(x = runif(i, min = -1, max = 1), mu = 0, alternative = "two.sided", paired = FALSE, conf.level = 0.95)
$p.value
}

set.seed(123) # seed for random sampling
cont_unif_dist_pval <- replicate(n = n_rounds, expr = {lapply(sample_sizes, continuous_uniform_dist)}, simplify =
TRUE) # repeat sampling n_rounds times

# plot p-value distribution for smallest sample size 5
cont_unif_dist_pval_sample_size_five <- as.vector(cont_unif_dist_pval[1,], mode = "numeric") # extract from df p
value generated from 10000 x sample_size = 5 and convert to numeric vector

hist(x = cont_unif_dist_pval_sample_size_five,
     main = paste("Continuous Uniform Distribution P Value of Sample Size 5"),
     xlab = paste("P values"))
```

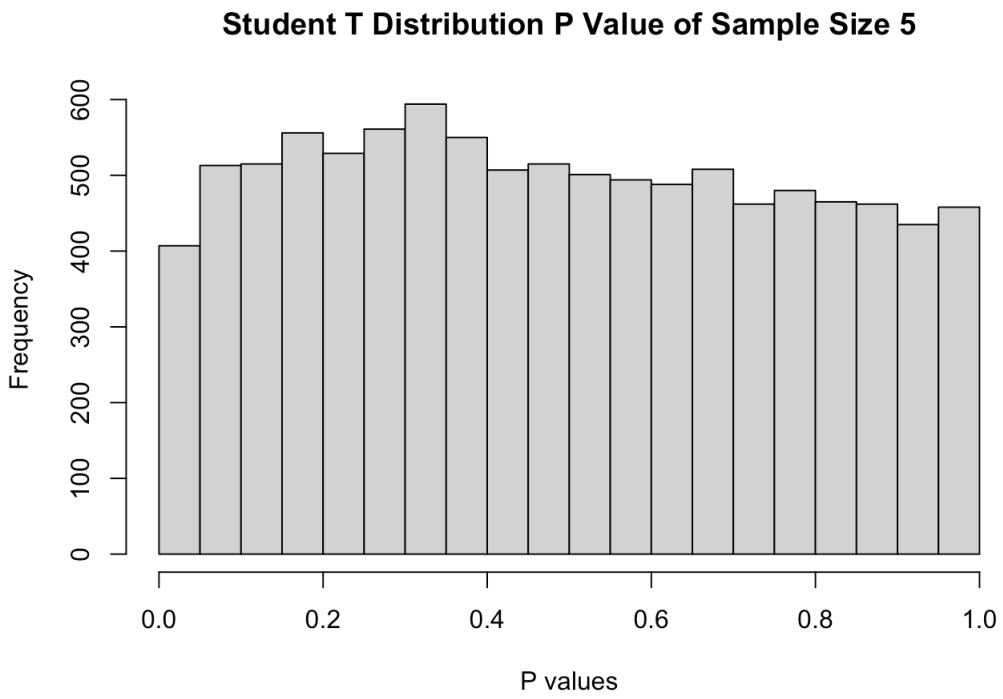



```
print(paste("The number of instances drawn from continuous uniform distribution with below nominal significance l
evel is", sum(cont_unif_dist_pval_sample_size_five < sig_level), "out of 10000."))
```

```
## [1] "The number of instances drawn from continuous uniform distribution with below nominal significance level
is 412 out of 10000."
```

sampling from student t distribution with parameter v=1

```
v <- 1 # parameter for degrees of freedom

student_t_dist <- function(i){
  t.test(x = rt(n = i, df = i - v ), mu = 0, alternative = "two.sided", paired = FALSE, conf.level = 0.95)$p.valu
e
}

# sample 10000 rounds for each sample size
set.seed(123) # seed for random sampling
student_t_dist_pval <- replicate(n = n_rounds, expr = {lapply(sample_sizes, student_t_dist)}, simplify = TRUE) #
repeat sample generate n times

# plot p-value distribution for smallest sample size
student_t_dist_pval_sample_size_five <- as.vector(student_t_dist_pval[1,], mode = "numeric") # extract from df p
value generated from 10000 x sample_size = 5 and convert to numeric vector

hist(x = student_t_dist_pval_sample_size_five,
     main = paste("Student T Distribution P Value of Sample Size 5"),
     xlab = paste("P values"))
```

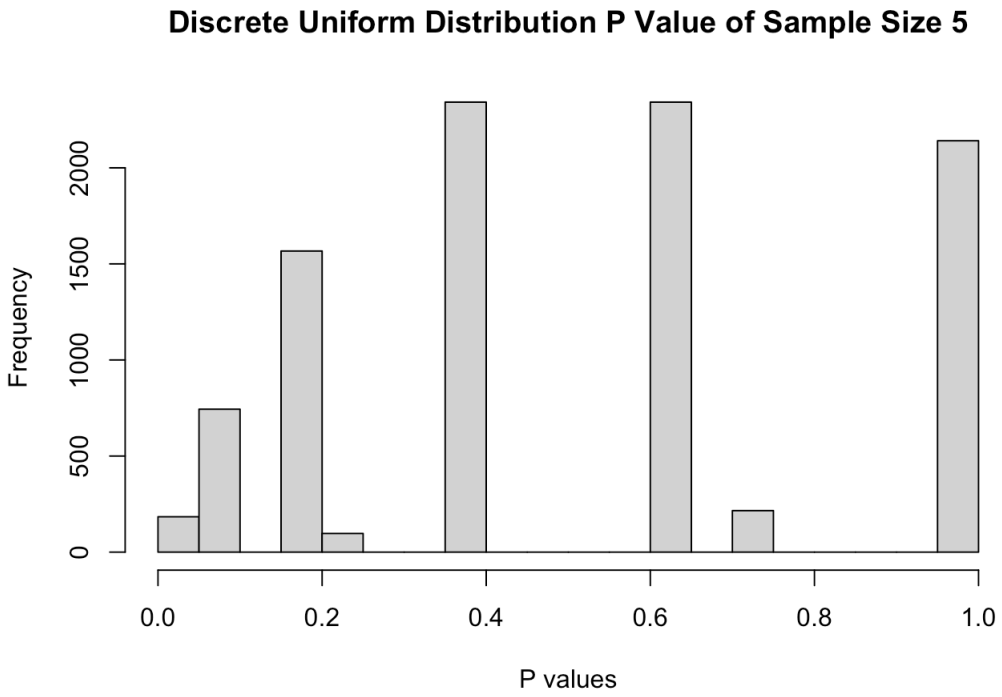



```
print(paste("The number of instances drawn from student t distribution with below nominal significance level is",
sum(student_t_dist_pval_sample_size_five < sig_level), "out of 10000"))
```

```
## [1] "The number of instances drawn from student t distribution with below nominal significance level is 199 ou
t of 10000"
```

sampling from discrete uniform distribution

```
discrete_uniform_dist <- function(i){
  obj <- try((t.test(x = round(runif(i, min = -1, max = 1),0), alternative = "two.sided", paired = FALSE, conf.le
vel = 0.95)), silent=TRUE)
  ifelse (is(obj,"try-error"), NaN, obj$p.value)
}

set.seed(123) # seed for random sampling
discrete_unif_dist_pval <- replicate(n = n_rounds, expr = {lapply(sample_sizes, discrete_uniform_dist)}, simplify
= TRUE) # repeat sample generate n times

# plot p-value distribution for smallest sample size
discrete_unif_dist_pval_sample_size_five <- as.vector(discrete_unif_dist_pval[1,], mode = "numeric") # extract fr
om df p value generated from 10000 x sample_size = 5 and convert to numeric vector

hist(x = discrete_unif_dist_pval_sample_size_five,
     main = paste("Discrete Uniform Distribution P Value of Sample Size 5"),
     xlab = paste("P values"))    # plot hist generated p value for distribution of smallest sample size
```

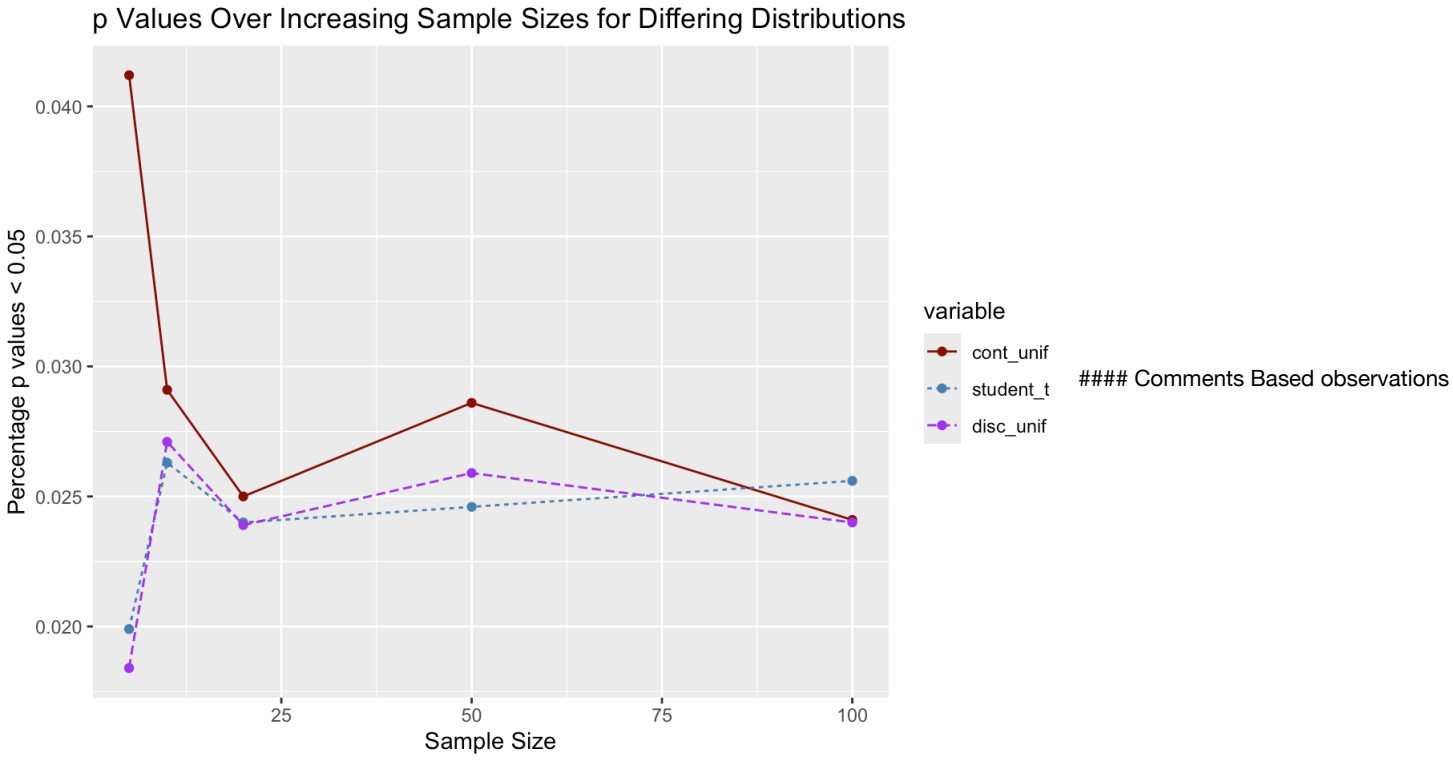



```
print(paste("The number of instances drawn from a discrete distribution with below nominal significance level i
s", sum(na.omit(discrete_unif_dist_pval_sample_size_five) < sig_level), "out of ",length(na.omit(discrete_unif_di
st_pval_sample_size_five))))#10000"))
```

```
## [1] "The number of instances drawn from a discrete distribution with below nominal significance level is 184 o
ut of  9633"
```

plot fraction of significant results for all distribution results as sample size increases

```
# count by row with significant results

student_t_frac <- as.data.frame(student_t_dist_pval)
student_t_frac <- (rowSums(student_t_frac < sig_level) / n_rounds)

cont_unif_frac <- as.data.frame(cont_unif_dist_pval)
cont_unif_frac <- (rowSums(cont_unif_frac < sig_level) / n_rounds)

disc_unif_frac_0 <- as.data.frame(discrete_unif_dist_pval)
disc_unif_notna_count <- rowSums(!is.na(disc_unif_frac_0))

# notna percentage
notna_percentage = function(row){
  rowsum <- sum(row[!is.na(row)] < sig_level)   # calculate number of elements lower than 0.025
  elementNumber <- length(!is.na(row))   # calculate number of total non-na elements
  return (rowsum / elementNumber)
}

disc_unif_frac <- apply(disc_unif_frac_0, 1 ,notna_percentage)

frac <- data.frame(cont_unif = c(cont_unif_frac), student_t = c(student_t_frac), disc_unif = c(disc_unif_frac))
frac$sample_size <- sample_sizes
frac <- frac %>% melt(id = c("sample_size"))

plot_f <- ggplot(frac, aes(x = sample_size, y = value)) +
  geom_line(aes(color = variable, linetype = variable)) +
  geom_point(aes(color = variable)) +
  scale_color_manual(values = c("darkred", "steelblue", "purple"))
plot_f + ggtitle("p Values Over Increasing Sample Sizes for Differing Distributions") + xlab("Sample Size") + yla
b("Percentage p values < 0.05")
```



from the above chart, it did not exhibit the expected inverse relationship between sample size and significant results (as sample increases, % significant results decreases (or at least stabilises)). Part of the variability could be explained by randomness.

Question 2

Read the data

```
lung_data <- read.csv("./lung_data.csv")
```

View the first few rows of the data

```
head(lung_data)
```

	Subject.id <int>	Lung.function <dbl>	Trial.arm <chr>
1	1	11.0	Control
2	2	11.1	Control
3	3	9.5	Control
4	4	10.1	Control
5	5	9.7	Control
6	6	9.4	Control

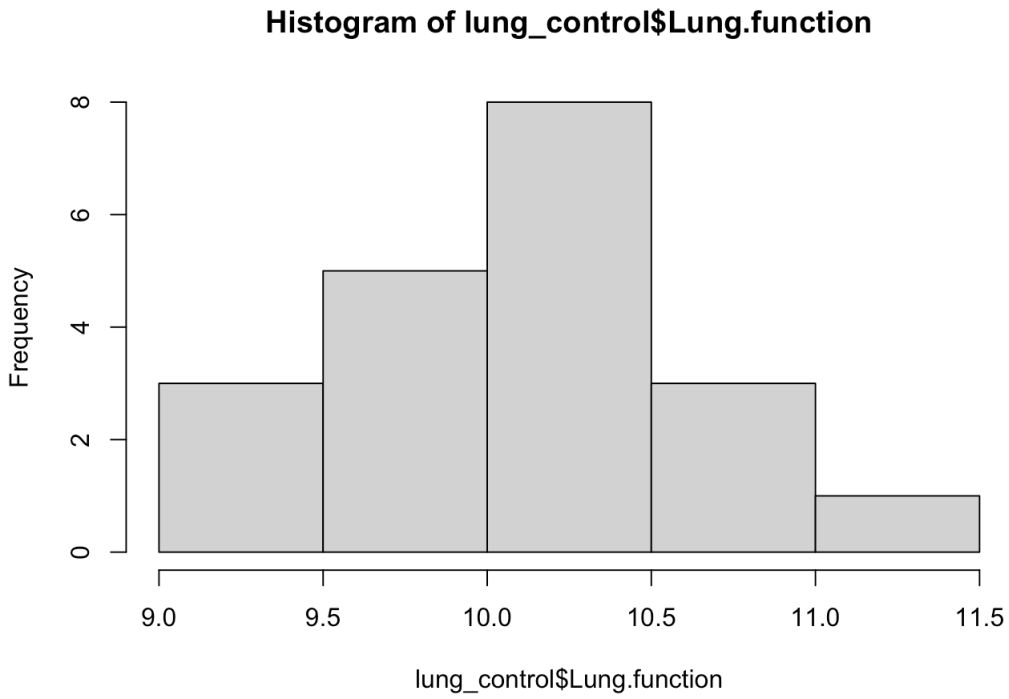
6 rows

Summary statistics

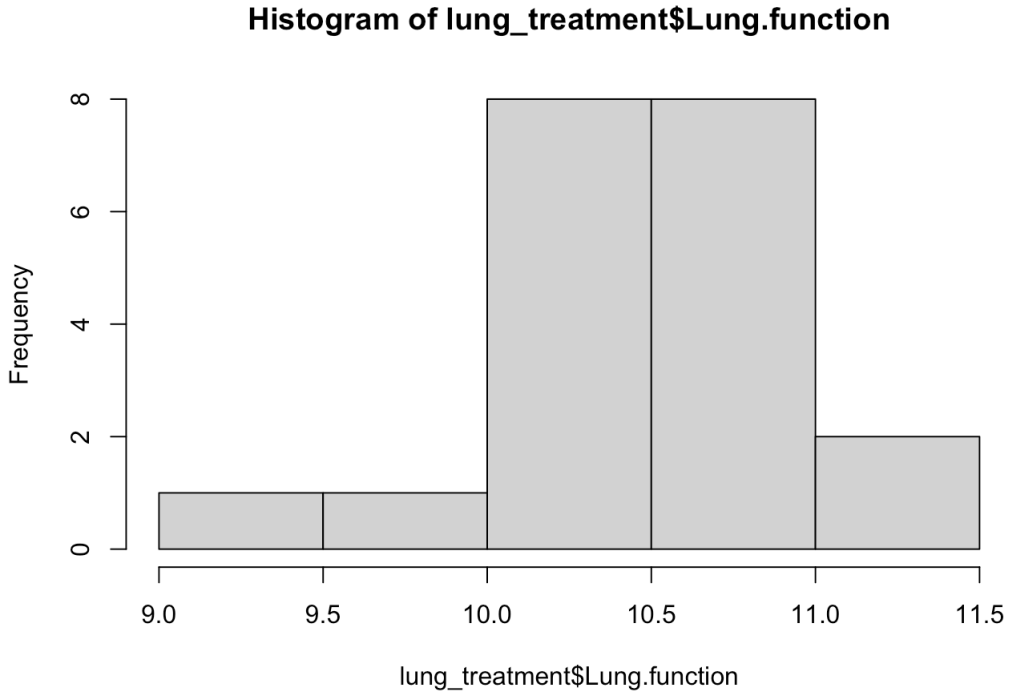
```
summary(lung_data)
```

```
##    Subject.id    Lung.function    Trial.arm
##  Min.   : 1.00   Min.   : 9.40   Length:40
##  1st Qu.:10.75   1st Qu.:10.07   Class :character
##  Median :20.50   Median :10.30   Mode  :character
##  Mean   :20.50   Mean   :10.31
##  3rd Qu.:30.25   3rd Qu.:10.62
##  Max.   :40.00   Max.   :11.40
```

```
lung_control <- subset(lung_data, Trial.arm=="Control")
hist(lung_control$Lung.function)
```




```
lung_treatment <- subset(lung_data, Trial.arm=="Treatment")
hist(lung_treatment$Lung.function)
```



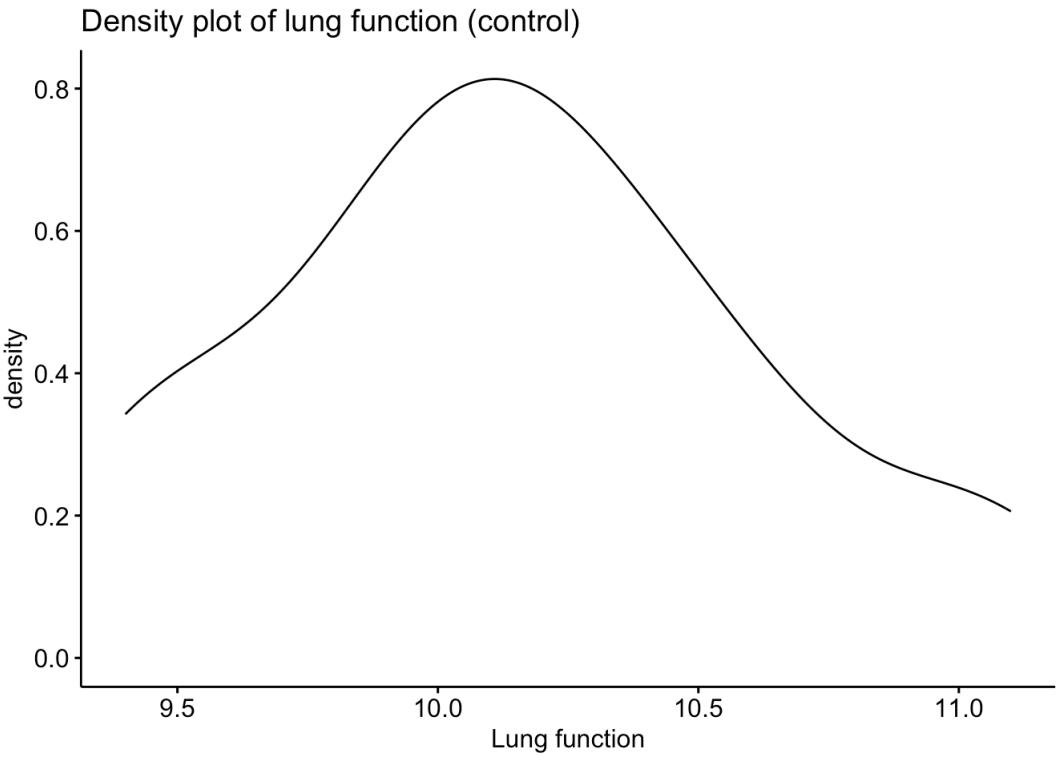

```
#These look close to normal.

shapiro.test(lung_control$Lung.function)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  lung_control$Lung.function
## W = 0.97119, p-value = 0.7797
```

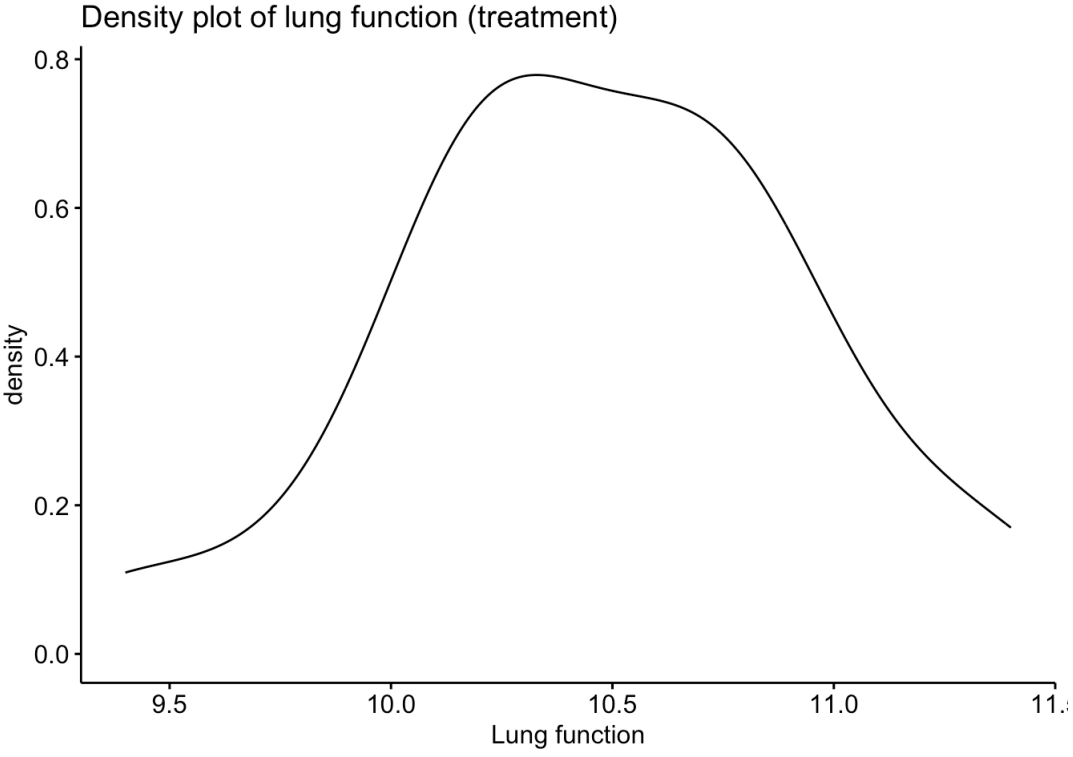
```
shapiro.test(lung_treatment$Lung.function)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  lung_treatment$Lung.function
## W = 0.9795, p-value = 0.9275
```

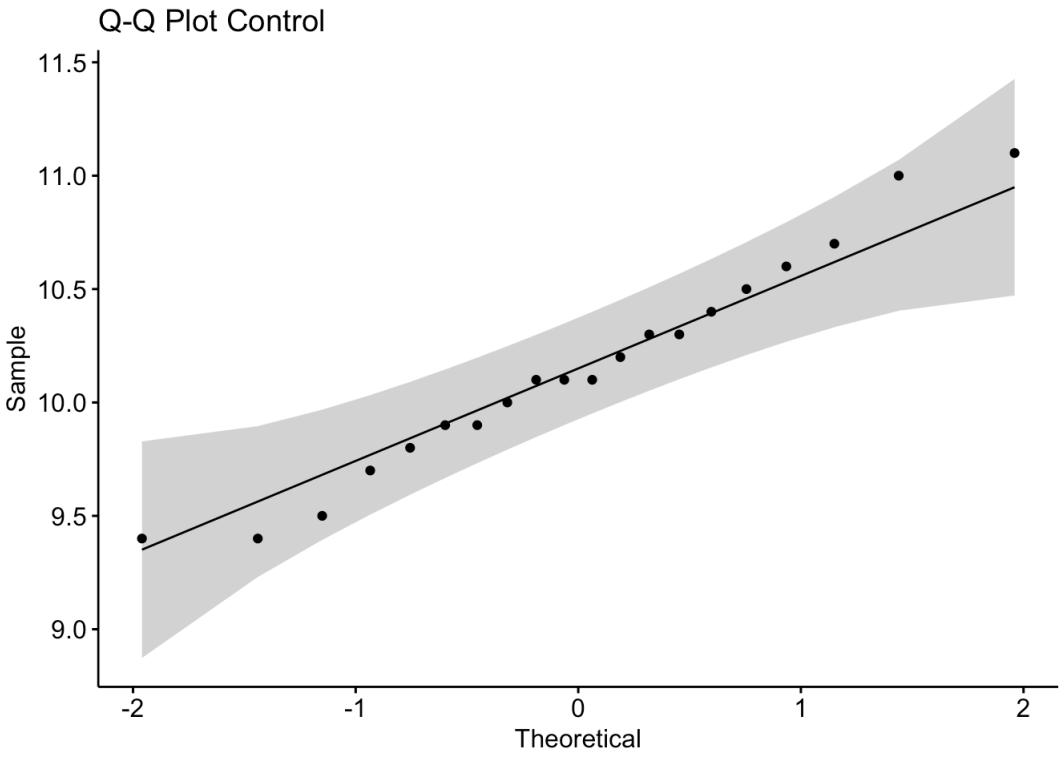
```
library(ggpubr)
ggdensity(lung_control$Lung.function,
          main = "Density plot of lung function (control)",
          xlab = "Lung function")
```



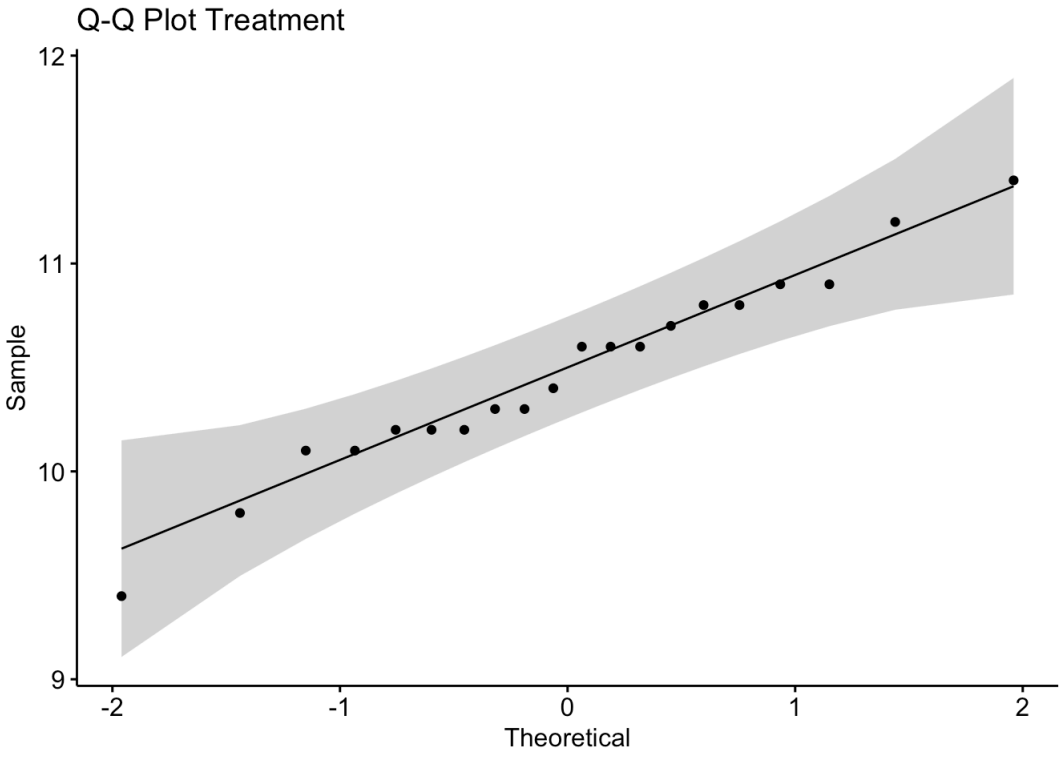

```
ggdensity(lung_treatment$Lung.function,
          main = "Density plot of lung function (treatment)",
          xlab = "Lung function")
```




```
ggqqplot(lung_control$Lung.function,
         main = "Q-Q Plot Control")
```



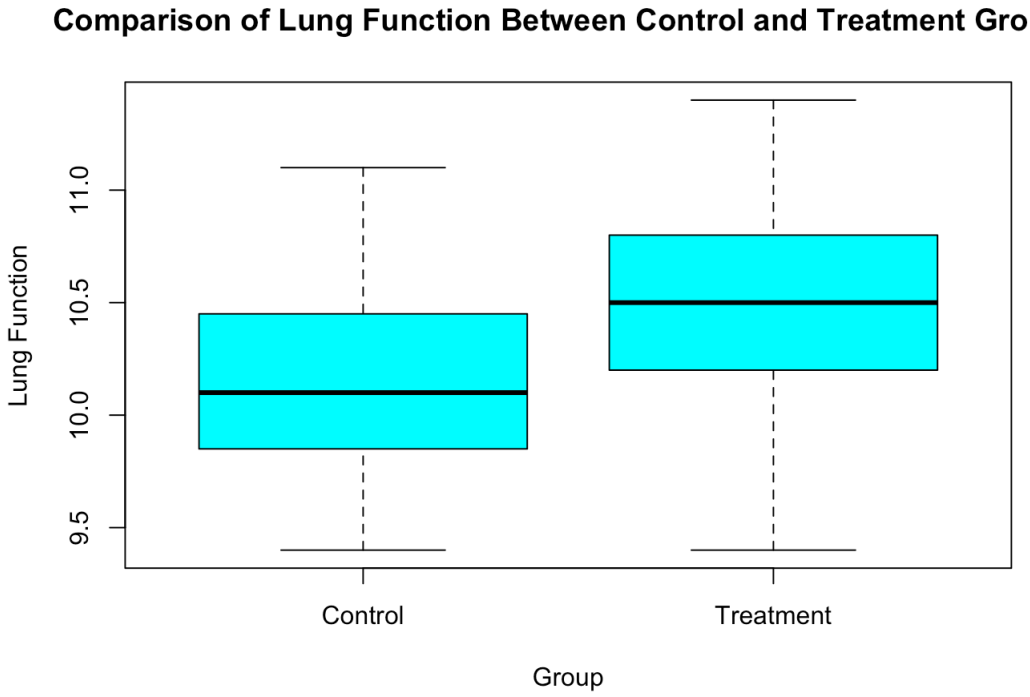

```
ggqqplot(lung_treatment$Lung.function,
         main = "Q-Q Plot Treatment")
```




```
#Though shapiro tests for these small samples are unlikely to be problematic, the histograms and q-q plots look o
k.
```

Visualize the data for each group

```
boxplot(Lung.function ~ Trial.arm, data = lung_data,
main="Comparison of Lung Function Between Control and Treatment Groups",
xlab="Group",
ylab="Lung Function",
col="cyan",
border="black"
)
```




```
# Test for difference in lung function between the two groups
t_test_result <- t.test(Lung.function ~ Trial.arm, data = lung_data)
t_test_result
```

```
##
##  Welch Two Sample t-test
##
## data:  Lung.function by Trial.arm
## t = -2.1569, df = 37.988, p-value = 0.0374
## alternative hypothesis: true difference in means between group Control and group Treatment is not equal to 0
## 95 percent confidence interval:
##  -0.63003538 -0.01996462
## sample estimates:
##   mean in group Control mean in group Treatment
##                  10.150                  10.475
```

Statistical analysis

The t-test yielded a p-value of 0.0374, which is below the standard significance level of 0.05. As there is a statistically significant difference in mean lung function between the control and treatment groups,we reject the null hypothesis. To conclude, Based on the results of the Welch Two Sample t-test, there is sufficient evidence to suggest that the observed mean difference in lung function (of 0.325) is not due to random variation (The new drug has an effect on lung function compared to the placebo).

Question 3

#3

FALSE: p-values are used to calculate the probability of the null hypothesis given the data. Why: p-values are a measure of surprise and calculated under the assumption that the null hypothesis is true.

TRUE: The significance level alpha is the probability of rejecting the null hypothesis when it is true.

FALSE: The Central Limit Theorem only holds if the population from which we are sampling is normally distributed. Why: The Central Limit Theorem states that for large sample sizes, the sample mean is approximately normally distributed, regardless of the distribution.

FALSE: As the sample size gets larger, the standard error of the sampling distribution of the sample mean gets larger as well. Why: The calculation of the standard error includes the root of sample size in the denominator and should decrease with increased sample size.

FALSE: The statistical power of a hypothesis test it the probability of not rejecting the null when H1 is true. Why: Statistical power is the probability that “one will correctly reject the null hypothesis” if the alternative hypothesis is true. (The alternative hypothesis being that the null hypothesis is false).

FALSE: The statistical power of a hypothesis test is the probability of rejecting H1 when H1 is true. Why: Statistical power is the probability that “one will correctly reject the null hypothesis” if the alternative hypothesis is true. (The alternative hypothesis being that the null hypothesis is false).

```
```