Worksheet3

Jingwen Luo (5597340) Daeho Lee (5597689) Gyeongsil kim (5597789)

2025-05-05

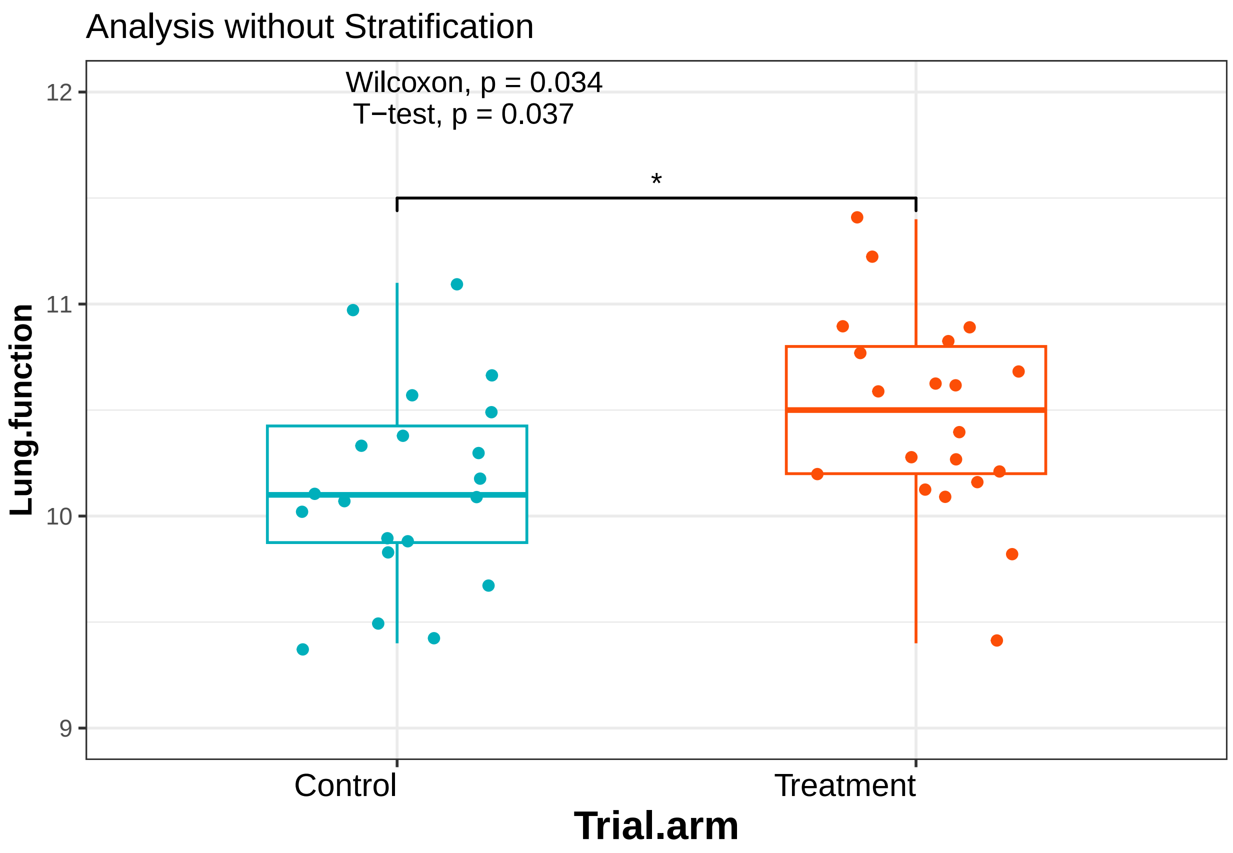
Exercise 1 - Stratification

Inspired by the lecture you asked the clinician from Worksheet 1 to send you the lung function data with the participant's sex included (lung_data_all.csv).

```
data_lung <- read.table("../Data/lung_data_all.csv",sep=",",header = T)
```

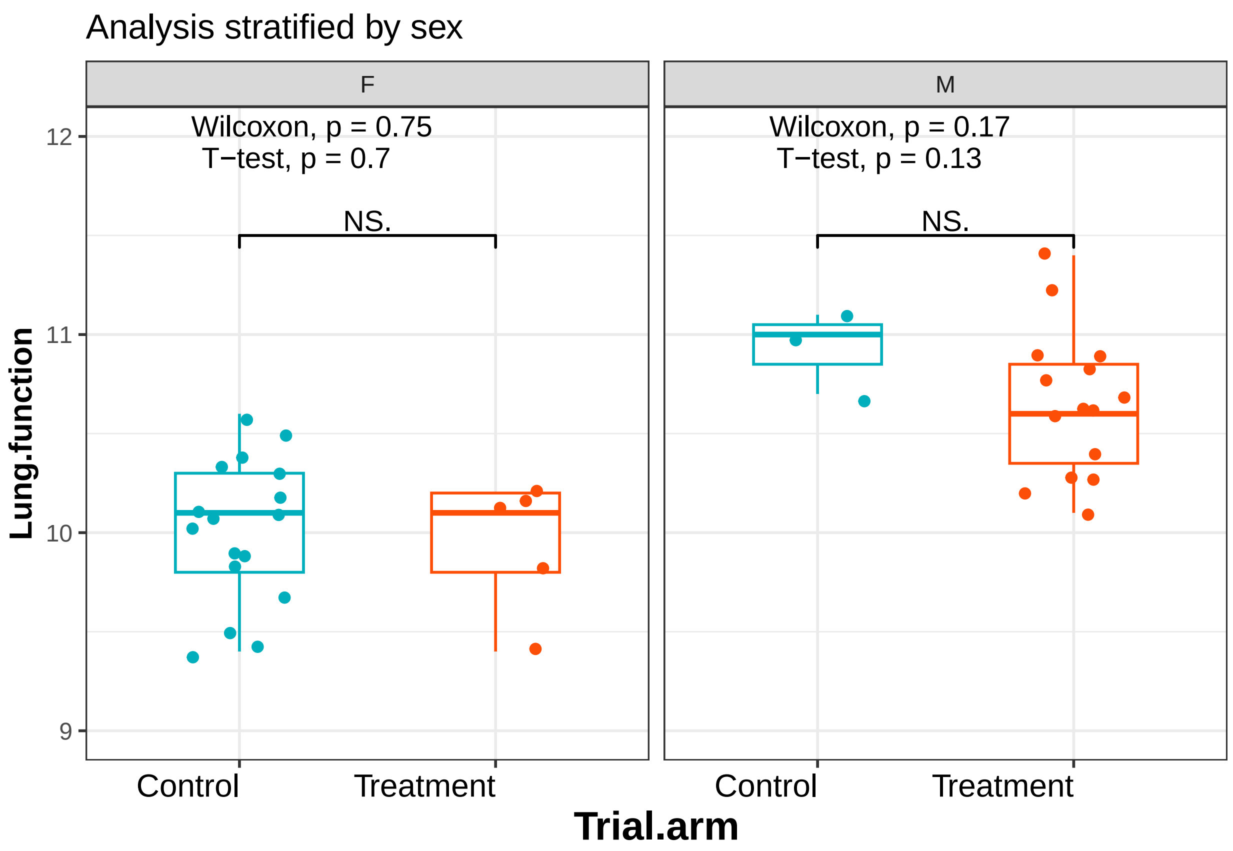
- Rerun the analysis to test whether the treatment shows any effect.

```
data_lung %>%
  mutate(Trial = Trial.arm, Lung = Lung.function) %>%
  ggboxplot(x = "Trial", y = "Lung",  color = "Trial",
            palette = c("#00AFBB", "#FC4E07"),
            ylab = "Lung.function", xlab = "Trial.arm",
            add = "jitter",add.params = list(size = 1.5),
            width = 0.5)+
  stat_compare_means(method = "wilcox.test",size=4)+
  stat_compare_means(method = "t.test",size=4, vjust = 1.5)+
  geom_signif(map_signif_level = TRUE,
              comparisons = list( c("Control","Treatment")),
              test = "t.test",vjust=0.1)+
  scale_y_continuous(limits = c(9,12))+
  labs(title = "Analysis without Stratification")+
  theme_bw()+
  theme(legend.position = "none",
        axis.title = element_text(size = 15,face = "bold"),
        axis.text.x = element_text(size = 12,hjust = 1,color = "black"),
        axis.title.y = element_text(size = 12,color = "black"))
```



- Run an analysis stratified by sex.

```
data_lung %>%
  mutate(Trial = Trial.arm, Lung = Lung.function) %>%
  ggboxplot(x = "Trial", y = "Lung",  color = "Trial",
            palette = c("#00AFBB", "#FC4E07"),
            ylab = "Lung.function", xlab = "Trial.arm",
            add = "jitter",add.params = list(size = 1.5),
            width = 0.5)+
  stat_compare_means(method = "wilcox.test",size=4)+
  stat_compare_means(method = "t.test",size=4, vjust = 1.5)+
  geom_signif(map_signif_level = TRUE,
              comparisons = list( c("Control","Treatment")),
              test = "t.test",vjust=0.1)+
  facet_grid(.~Sex)+
  scale_y_continuous(limits = c(9,12))+
  labs(title = "Analysis stratified by sex")+
  theme_bw()+
  theme(legend.position = "none",
        axis.title = element_text(size = 15,face = "bold"),
        axis.text.x = element_text(size = 12,hjust = 1,color = "black"),
        axis.title.y = element_text(size = 12,color = "black"))
```



- Summarize your findings in a statistical report.

If we run the analysis without knowing Gender information, we found it's significant different between two groups. However, We found if we stratified the data by sex, we will conclude that the treatment shows no effect on both male and female group.

Exercise 2 - - Confounders

To explore the effect of confounders on statistical analyses, generate data from the following simulation, for reasonable N:

```
generate <- function(N){
  w <- 1 + rnorm(N)
  x_0 <- rnorm(N)
  y_0 <- rnorm(N)
  x <- x_0 + w
```

```r
  y <- y_0 - w
  return(data.frame(x = x, y = y))
}

n_simul <- 500
result <- data.frame(x = numeric(0),y = numeric(0))

for (i in 1:n_simul) {
  result <- rbind(result,generate(i))
}
```

- What is the distribution of x and y (i.e. what family of distributions do the two variables belong to and with which parameters)? Plot the histograms of their sampled values to check.

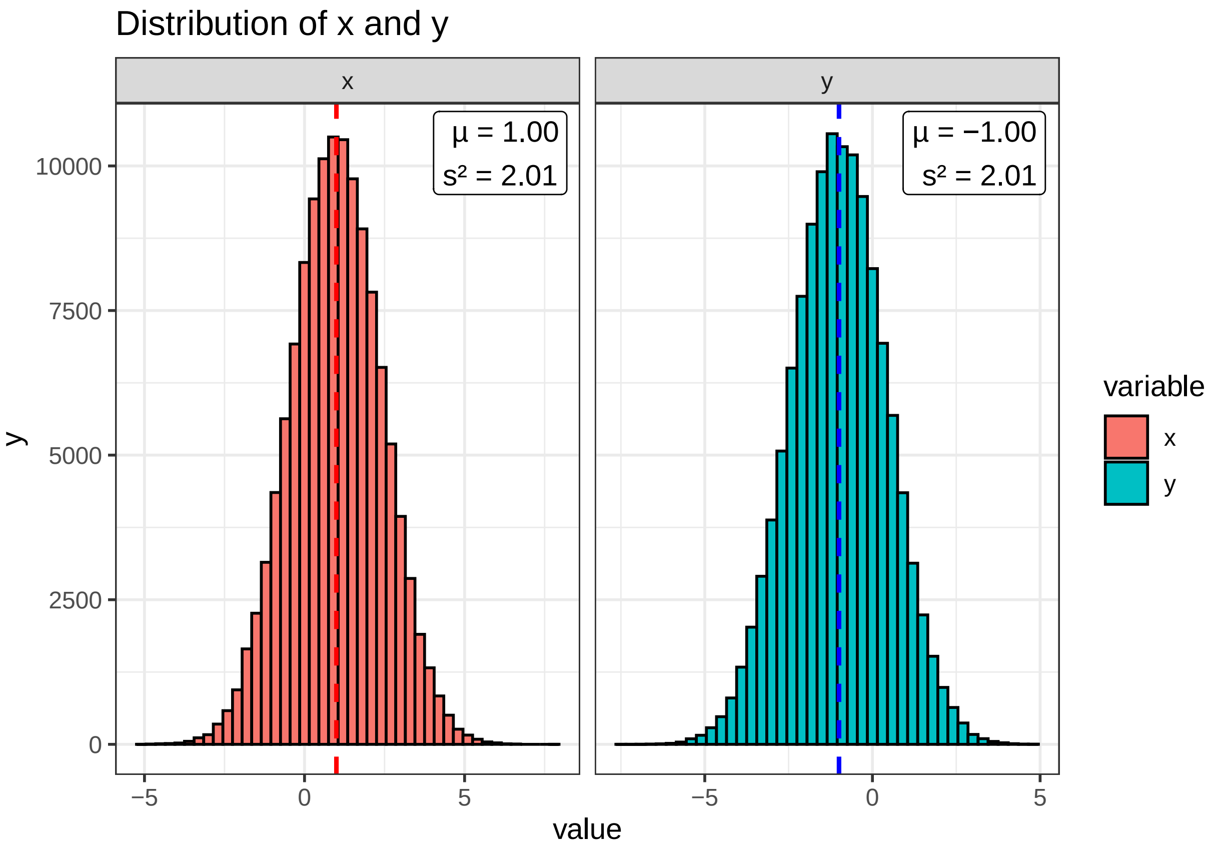
x_0 ~ N(0, 1) and y_0 ~ N(0, 1), w ~ N(1, 1). x_0, y_0 and w are independent to each other.

Theoretically, X ~ N(1, 2) and Y ~ N(-1, 2)

```r
distribution <- result %>%
  pivot_longer(
    cols = x:y,
    names_to = c("variable"),
    values_to = "value")

stats_df <- distribution %>%
  group_by(variable) %>%
  summarise(mean = mean(value, na.rm = TRUE),
            var = var(value, na.rm = TRUE))

stats_df
```

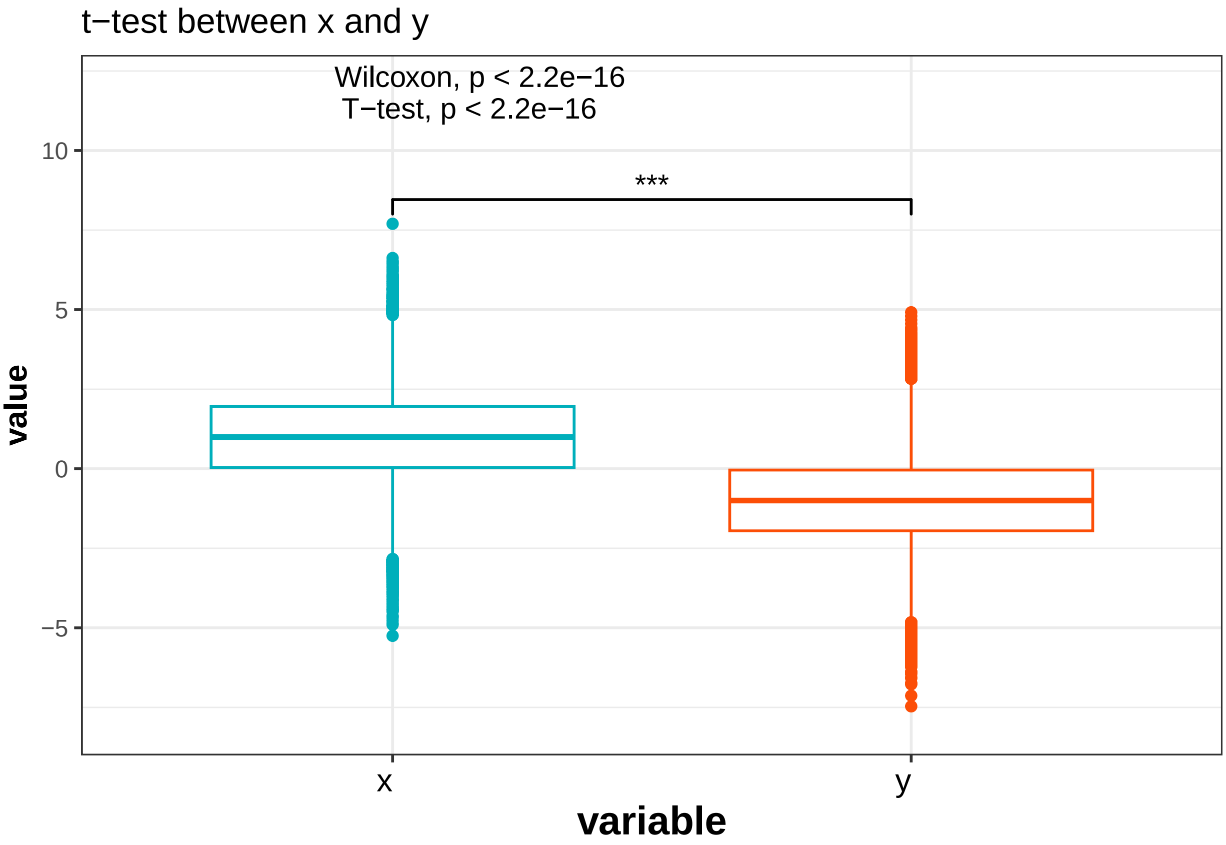
```
## # A tibble: 2 x 3
##   variable   mean   var
##   <chr>     <dbl> <dbl>
## 1 x         0.995  2.01
## 2 y        -0.997  2.01
```

```r
distribution %>%
  ggplot(aes(x = value, fill = variable)) +
  geom_histogram(color = "black", binwidth = 0.3, position = "dodge") +
  geom_vline(data = stats_df,aes(xintercept = mean, color = variable),
    linetype = "dashed",linewidth = 0.8,show.legend = FALSE ) +
  geom_label(data = stats_df,aes(x = Inf, y = Inf,
            label = sprintf("  = %.2f\n ² = %.2f", mean, var)),
            hjust = 1.1, vjust = 1.1,color = "black",fill = "white",size = 4)+
  theme_bw() +
  facet_grid(. ~ variable, scales = 'free') +
  scale_color_manual(values = c("x" = "red", "y" = "blue")) +
  labs(title = "Distribution of x and y")
```



- Compute a t-test between x and y, and report and interpret your findings.

```
distribution %>%
  ggboxplot(x = "variable", y = "value",  color = "variable",
            palette = c("#00AFBB", "#FC4E07"),
            ylab = "value", xlab = "variable")+
  stat_compare_means(method = "wilcox.test",size=4)+
  stat_compare_means(method = "t.test",size=4, vjust = 1.5)+
  geom_signif(map_signif_level = TRUE,comparisons = list( c("x","y")),
              test = "t.test",vjust=0.1)+
  scale_y_continuous(limits = c(-8,12))+
  labs(title = "t-test between x and y")+
  theme_bw()+
  theme(legend.position = "none",
        axis.title = element_text(size = 15,face = "bold"),
        axis.text.x = element_text(size = 12,hjust = 1,color = "black"),
        axis.title.y = element_text(size = 12,color = "black"))
```



Rather than adding w to x0 to generate x and subtracting it from y0 to generate y, now randomize the adding/subtracting of w to generate two new vectors x' and y':

- For each element of w randomly (uniformly) decide whether to multiply it by either +1 or -1 (randomly and uniformly) to either keep or swap its sign. Store the vector as wp. Generate xp and yp according to:

```
generate_new <- function(N){
  w <- 1 + rnorm(N)
  x_0 <- rnorm(N)
  y_0 <- rnorm(N)
  signs <- sample(c(1, -1), N, replace = TRUE)
  wp <- w * signs

  x_p <- x_0 + wp
  y_p <- y_0 - wp
  return(data.frame(x_p = x_p, y_p = y_p, wp = wp))
}

n_simul <- 1000
result_new <- data.frame(x_p = numeric(0),y_p = numeric(0),wp = numeric(0))

for (i in 1:n_simul) {
  result_new <- rbind(result_new,generate_new(i))
}
```

Compute a t-test between xp and yp, and report and interpret your findings.

Let $w \sim \mathcal{N}(1,1)$ and let the random sign variable $s \in \{-1,+1\}$ be sampled uniformly. Define $w_p = s \cdot w$. Then the distribution of w_p is:

$$w_p \sim \frac{1}{2}\mathcal{N}(1,1) + \frac{1}{2}\mathcal{N}(-1,1)$$

This is a *bimodal mixture of Gaussians*, and thus not normally distributed.

We can still compute the first and second moments:

$$\mathbb{E}[w_p] = \mathbb{E}[s] \cdot \mathbb{E}[w] = 0 \cdot 1 = 0$$

$$\mathrm{Var}(w_p) = \mathbb{E}[w_p^2] - (\mathbb{E}[w_p])^2 = \mathbb{E}[w^2] = \mathrm{Var}(w) + (\mathbb{E}[w])^2 = 1 + 1 = 2$$

Let $x_0, y_0 \sim \mathcal{N}(0,1)$, independent of w. Define:

$$x_p = x_0 + w_p, \quad y_p = y_0 - w_p$$

Then, since $x_0 \sim \mathcal{N}(0,1)$ and w_p has mean 0 and variance 2:

$$x_p \approx \mathcal{N}(0,3), \quad y_p \approx \mathcal{N}(0,3)$$

Therefore, the t-test comparing x_p and y_p shows no significant difference,
which is theoretically supported by their approximate equality in distribution.

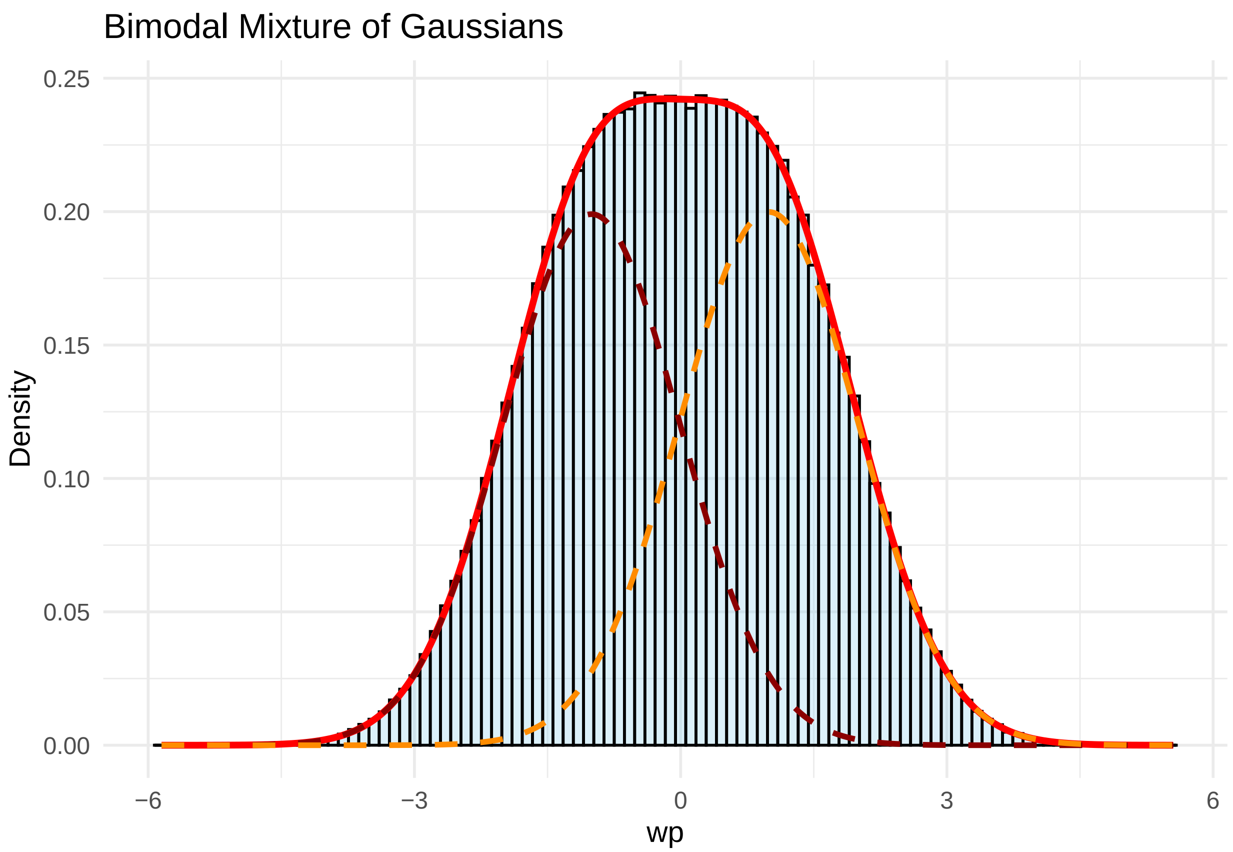
```r
distribution_new <- result_new %>%
  pivot_longer(
    cols = x_p:wp,
    names_to = c("variable"),
    values_to = "value")

stats_df_new <- distribution_new %>%
  group_by(variable) %>%
  summarise(mean = mean(value, na.rm = TRUE),
            var = var(value, na.rm = TRUE))

stats_df_new
```

```
## # A tibble: 3 x 3
##   variable     mean   var
##   <chr>       <dbl> <dbl>
## 1 wp        0.00145  2.00
## 2 x_p       0.00406  2.99
## 3 y_p      -0.00166  3.00
```

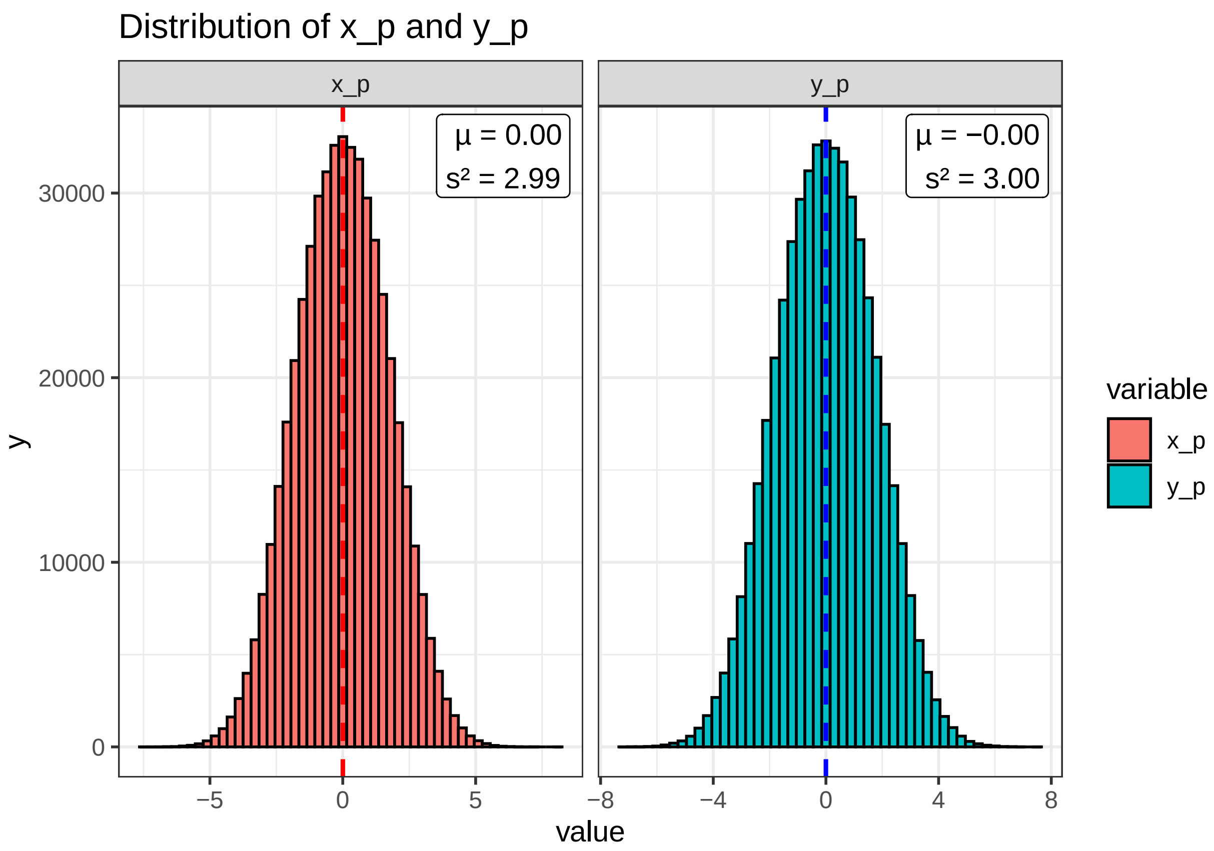
```r
stats_df_new2 <- stats_df_new %>%  filter(variable != "wp")

w_p <- distribution_new %>%
  filter(variable == "wp")

hist_data <- hist(w_p$value, breaks = 100, plot = FALSE)
df <- data.frame(
  x = (head(hist_data$breaks, -1) + tail(hist_data$breaks, -1)) / 2,
  y = hist_data$density
)

#normal distribution
gauss <- function(x, mu, sigma, A) {
  A * exp(-(x - mu)^2 / (2 * sigma^2))
}

bimodal <- function(x, mu1, sigma1, A1, mu2, sigma2, A2) {
  gauss(x, mu1, sigma1, A1) + gauss(x, mu2, sigma2, A2)
}
```

```r
fit <- nls(
  y ~ gauss(x, mu1, sigma1, A1) + gauss(x, mu2, sigma2, A2),
  data = df,
  start = list(mu1 = -1, sigma1 = 1, A1 = 0.5,
               mu2 = 1, sigma2 = 1, A2 = 0.5),
  control = list(maxiter = 500)
)

x_fit <- seq(min(df$x), max(df$x), length.out = 500)
params <- as.list(coef(fit))
fit_df <- tibble(
  x = x_fit,
  total = bimodal(x_fit, !!!params),
  comp1 = gauss(x_fit, params$mu1, params$sigma1, params$A1),
  comp2 = gauss(x_fit, params$mu2, params$sigma2, params$A2)
)

# visualization of wp
ggplot() +
  geom_histogram(data = w_p, aes(x = value, y = after_stat(density)),
                 bins = 100, fill = "skyblue", alpha = 0.3, color = "black") +
  geom_line(data = fit_df, aes(x = x, y = total), color = "red", size = 1.2) +
  geom_line(data = fit_df, aes(x = x, y = comp1), color = "darkred",
            linetype = "dashed", size = 1) +
  geom_line(data = fit_df, aes(x = x, y = comp2), color = "darkorange",
            linetype = "dashed",size = 1) +
  labs(title = "Bimodal Mixture of Gaussians", x = "wp", y = "Density") +
  theme_minimal()
```

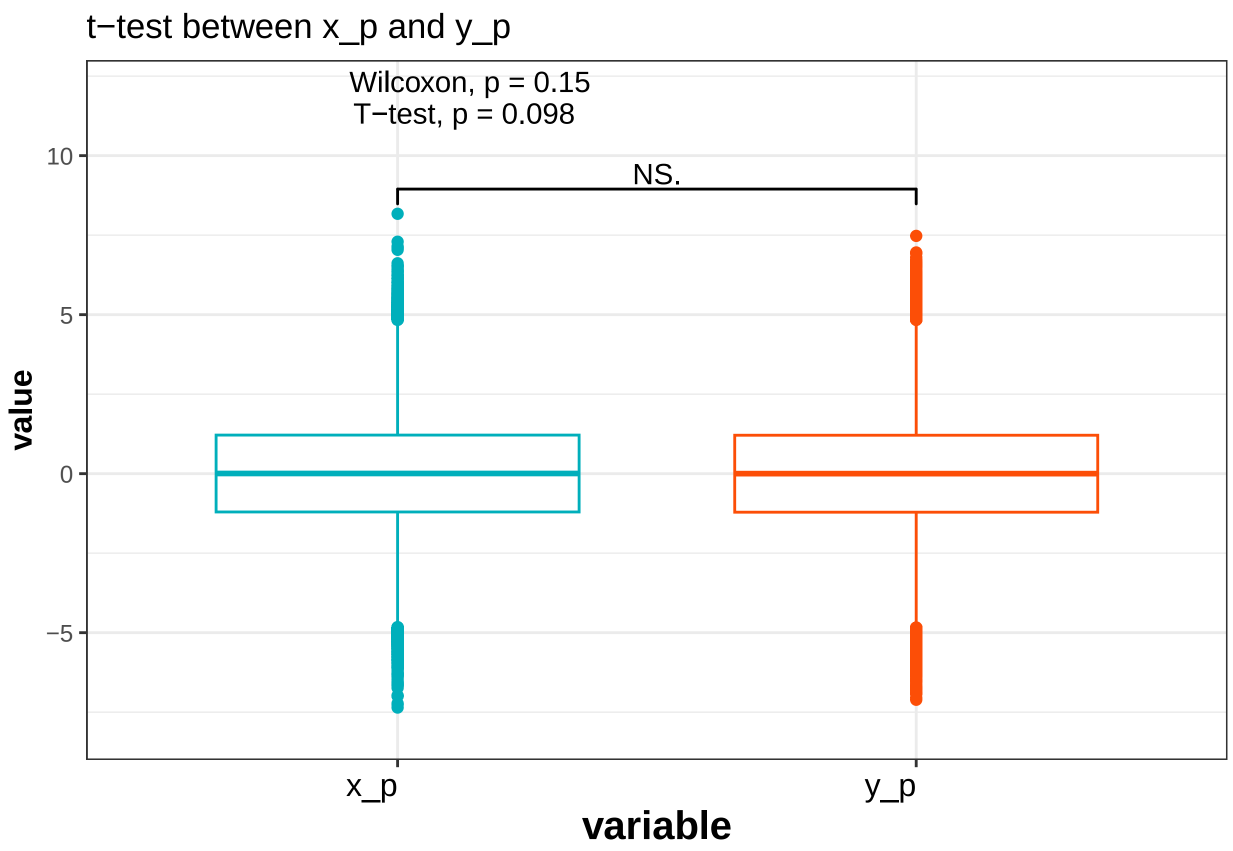
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
distribution_new %>%
  filter(variable!="wp") %>%
  ggplot(aes(x = value, fill = variable)) +
  geom_histogram(color = "black", binwidth = 0.3, position = "dodge") +
  geom_vline(data = stats_df_new2,
             aes(xintercept = mean, color = variable),
    linetype = "dashed",linewidth = 0.8,show.legend = FALSE ) +
  geom_label(data = stats_df_new2,aes(x = Inf, y = Inf,
            label = sprintf("  = %.2f\n ² = %.2f", mean, var)),
            hjust = 1.1, vjust = 1.1,color = "black",fill = "white",size = 4)+
  theme_bw() +
  facet_grid(. ~ variable, scales = 'free') +  # 分面显示
  scale_color_manual(values = c("x_p" = "red", "y_p" = "blue")) +
  labs(title = "Distribution of x_p and y_p")
```



```
distribution_new %>%
  filter(variable!="wp") %>%
  ggboxplot(x = "variable", y = "value",  color = "variable",
            palette = c("#00AFBB", "#FC4E07"),
            ylab = "value", xlab = "variable")+
  stat_compare_means(method = "wilcox.test",size=4)+
  stat_compare_means(method = "t.test",size=4, vjust = 1.5)+
  geom_signif(map_signif_level = TRUE,comparisons = list( c("x_p","y_p")),
    test = "t.test",vjust=0.1,)+
  scale_y_continuous(limits = c(-8,12))+
  labs(title = "t-test between x_p and y_p")+
  theme_bw()+
  theme(legend.position = "none",
        axis.title = element_text(size = 15,face = "bold"),
        axis.text.x = element_text(size = 12,hjust = 1,color = "black"),
        axis.title.y = element_text(size = 12,color = "black"))
```



Exercise 3 - Simple linear regression

Read the chocolate.csv data set and fit a simple linear regression model.Useful library for plotting datapoints as flags: ggflags

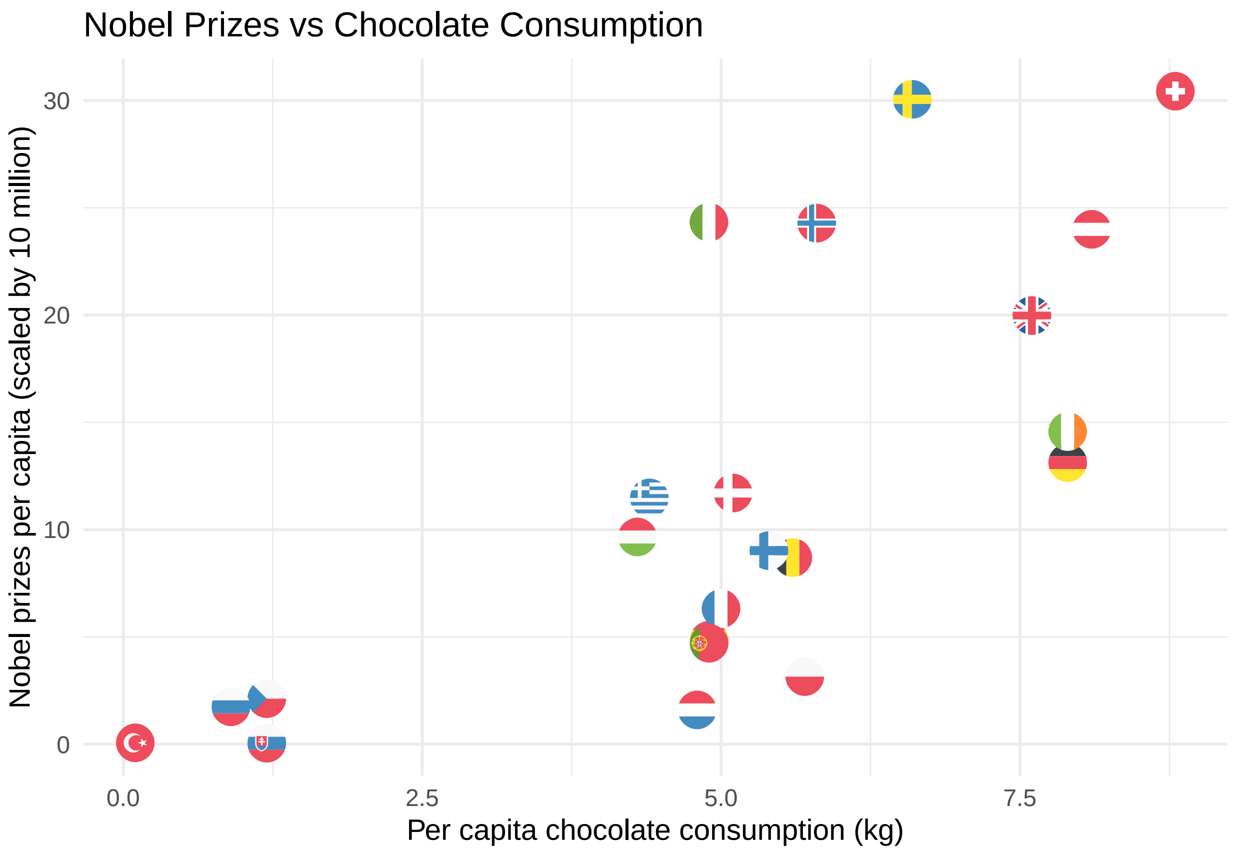
- What are the dependent and independent variables for your model? Visualize both using a scatter-plot.

Nobel prizes per capita (scaled by 10 million) is the dependent variable and Per capita chocolate consumption (kg) is the independent variable.

```
chocolate <- read_csv("../Data/chocolate.csv")
```

```
## Rows: 22 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Country
## dbl (2): Nobel prizes per capita (scaled by 10 million), Per capita chocolat...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
dependent_var <- "Nobel prizes per capita (scaled by 10 million)"
independent_var <- "Per capita chocolate consumption (kg)"

country_codes <- c("CH", "AT", "IE", "DE", "GB", "SE", "NO", "PL", "BE", "FI",
                   "DK", "FR", "IT", "ES", "PT", "NL", "GR", "HU", "CZ", "SK",
                   "RU", "TR")
chocolate <- chocolate %>% mutate(country_code = tolower(country_codes))


ggplot(chocolate, aes(x = .data[[independent_var]],
                      y = .data[[dependent_var]])) +
  geom_flag(aes(country = country_code), size = 6) +
  labs(title = "Nobel Prizes vs Chocolate Consumption") +
  theme_minimal()
```



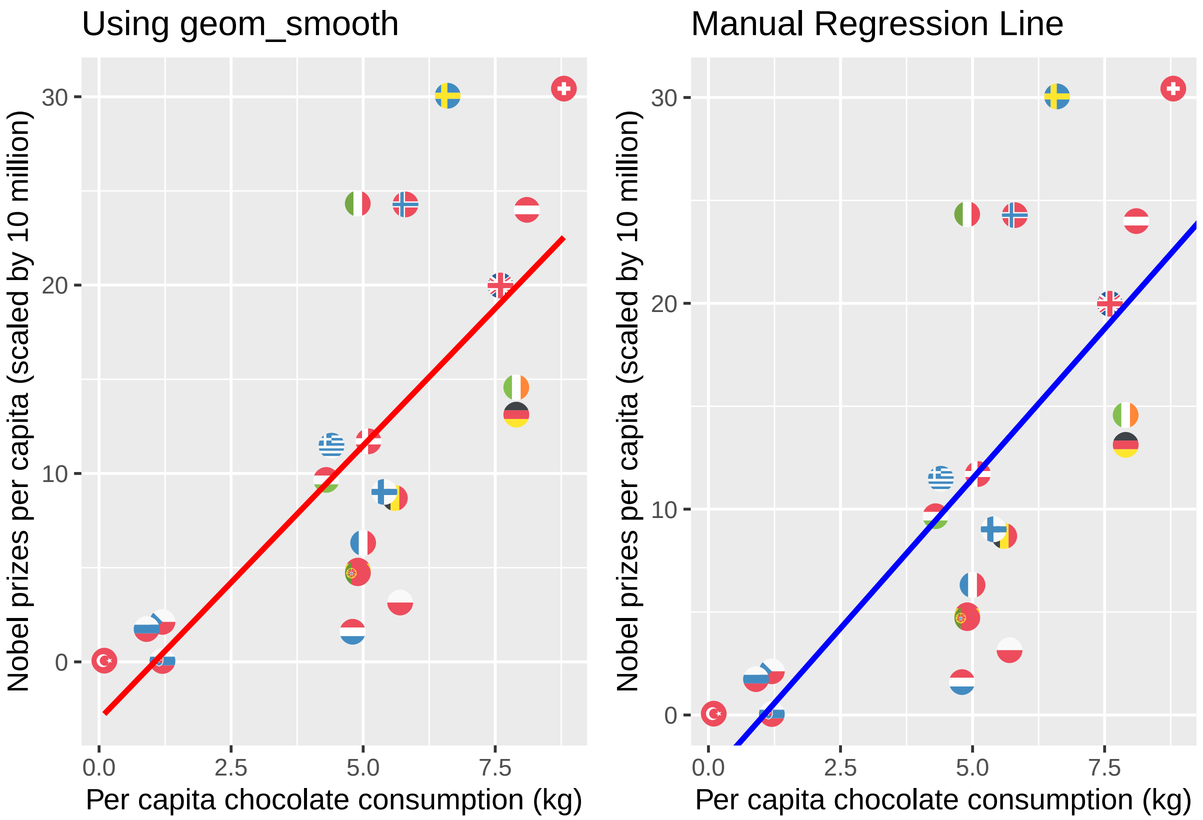
- Compute the intercept and slope of the regression line through the data and the cofficient of determination R2.

```r
model <- lm(
  `Nobel prizes per capita (scaled by 10 million)` ~
  `Per capita chocolate consumption (kg)`,
  data = chocolate
)

intercept <- round(coef(model)[1], 2)
slope <- round(coef(model)[2], 2)
r_squared <- round(summary(model)$r.squared, 3)
cat(sprintf("Regression Function :
            Nobel = %.2f + %.2f * Chocolate\nR² = %.3f",
            intercept, slope, r_squared))
```

```
## Regression Function :
##             Nobel = -3.06 + 2.91 * Chocolate
## R² = 0.513
```

- Add the computed regression line to the scatterplot both using geom_smooth and manually from the slope and intercept.

```r
# geom_smooth
p1 <- ggplot(chocolate,
             aes(x = .data[[independent_var]],
                 y = .data[[dependent_var]])) +
  geom_flag(aes(country = country_code), size = 4) +
  geom_smooth(method = "lm", formula = y ~ x,
              color = "red", se = FALSE) +
  labs(title = "Using geom_smooth")

# manually
p2 <- ggplot(chocolate,
             aes(x = .data[[independent_var]],
                 y = .data[[dependent_var]])) +
  geom_flag(aes(country = country_code), size = 4) +
  geom_abline(intercept = intercept,slope = slope,
    color = "blue",linewidth = 1) +
  labs(title = "Manual Regression Line")

p1 + p2
```

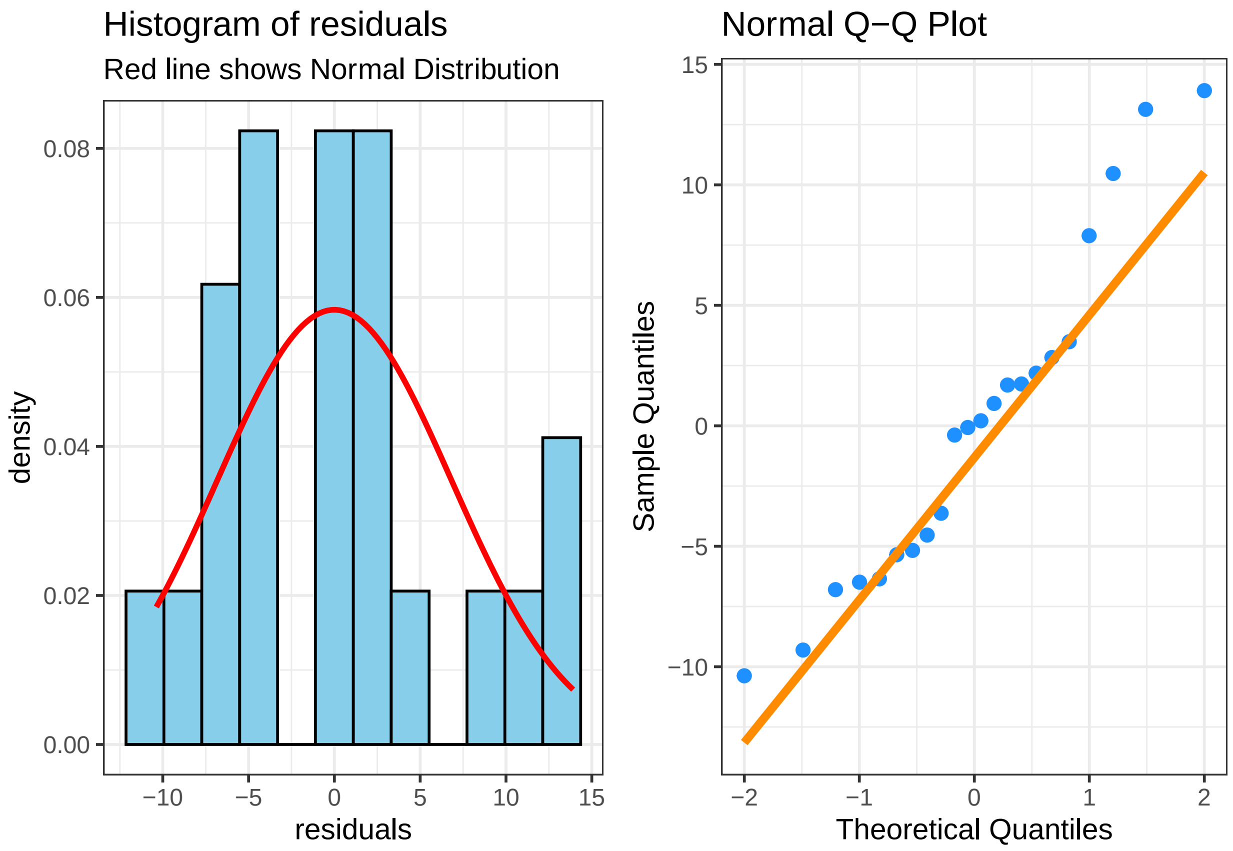


- Check if the assumptions of your model are fulfilled with visualizations.

According to visualizations of histogram and Q-Q plot of residuals,it is not very normally distributed and has heavier tail on the right, which means model are not well fulfilled.

```
# Residuals of model
residuals <- resid(model)

# Histogram of residuals
p_hist <- ggplot(data.frame(residuals), aes(x = residuals)) +
  geom_histogram(aes(y = after_stat(density)), bins = 12,
                 fill = "skyblue", color = "black") +
  stat_function(fun = dnorm,
    args = list(mean = mean(residuals), sd = sd(residuals)),
    color = "red", linewidth = 1) +
  labs(title = "Histogram of residuals",
       subtitle = "Red line shows Normal Distribution") +
  theme_bw()

# Q-Q plot
```

```
p_qq <- ggplot(data.frame(residuals), aes(sample = residuals)) +
  stat_qq(color = "dodgerblue", size = 2) +
  stat_qq_line(color = "darkorange",linewidth=1.5) +
  labs(title = "Normal Q-Q Plot")+
  xlab("Theoretical Quantiles")+
  ylab("Sample Quantiles") +
  theme_bw()

grid.arrange(p_hist, p_qq,ncol = 2)
```



# Exercise 4 - Review questions

(a) Is it a reasonable choice to use the chi square test to analyse your data? If not, what other test would you chose? No, the chi-square test is not appropriate here because we are comparing means of a continuous variable (blood pressure), not categorical data.T-test is better in this case.

(b) You want to do a power analysis before you start the study. What quantities do you need to know or estimate to do this?

We need to know or estimate: n - number of observations (per group) delta - true difference in means sd - standard deviation significance level (usually 0.05)

(c) Your power analysis shows a rather weak power. What factor is the easiest to change to increase the power?

Increasing the sample size

(d) If you put an equal amount of female/male participants in the high/low group this can be described as blocking. (T)

(e) Since you have many people in your study, it is necessary to correct for multiple testing. (F)

Multiple testing corrections are needed when performing multiple comparisons (e.g., testing multiple hypotheses)

(f) In a good experimental design you only randomize background effects that you cannot block. (T)

(g) What term describes the role of “smoking habit” in your first study?

Smoking habit acts as a confounder.

(h) Explain the contradictory results of your second study.

The contradiction arises due to confounding by obesity. In the second study, the control group had more obese individuals, which masked the true effect of alcohol.

(i) It was not necessary to do the second study, because the first study already showed the effect of alcohol consumption on blood pressure (F)

Both studies are biased due to different confounders.

(j) Your second study is proof that regular consumption of high amounts of alcohol increases blood pressure(F)

Confounding by obesity invalidates causal claims.