# DATA2002 Report Assignment

SID 500666580

19/09/2021

## Contents



## Introduction

With the advent of a novel coronavirus, it is evident that the COVID-19 pandemic has damaged humankind in terms of both physical and mental health. According to a recent study conducted in the UK, it appears that mental well-being was destroyed during the COVID-19 pandemic (Savage et al., 2020). Thus, this report aims to investigate the relationship between Australian students' mental health and various factors, such as year of study, staying-in-Australia status, or gender. A survey has been conducted at the University of Sydney in September 2021, and the required data has been obtained from 696 students taking a DATA2002 course and 58 students taking a DATA2902 (Advanced) course. However, such a survey requires careful consideration due to its several drawbacks.

First, as this survey only considered students taking a particular course at the University of Sydney, the sample students are not randomly selected nor representative of the whole group of Australian students, an entire population. Likewise, students' mental health was quantified as self-assessed stress scale data on a scale from 0 representing the smallest amount of stress to 10 representing the largest amount of stress. One potential bias with this stress data is that it is hard to measure students' proper mental health status with a single factor, stress. Furthermore, since these students are not professional mental health-related personnel, they may have measured their current mental health status inaccurately.

On the other hand, non-responses and inconsistency of data have been detected across several survey questions. Although the amount of such invalid data that are related to this analysis is small, it requires further studies with larger-sized random sample data. Notably, this analysis has cleaned up data to deal with such issues related to inconsistent and missing data. Overall, even if this report provides readers with remarkable results, further research is required to validate our findings.

# The Language of Statistics

Before taking a look at the analyses, it is worthwhile to know statistical notations and terminologies that will be frequently used to interpret our findings below. The language of statistics required for readers to know is the following:

1. Hypothesis tests

   In general, hypothesis tests refer to statistical tests that are utilized to make statistical inferences by either rejecting a null hypothesis in favor of an alternative hypothesis or not rejecting the null hypothesis. The null hypothesis is a statement about equalities (i.e., status-quo) between two different population parameters (the numerical facts about the entire population), and it is denoted by $H_0$. While an alternative hypothesis is a statement about inequalities and such inequalities could be "greater than", "less than", or "not equal to" between two different parameters used in $H_0$. The alternative hypothesis is denoted by $H_1$ or $H_a$.

   In a hypothesis test, we first assume $H_0$ is true. Then under $H_0$, we compute a single statistic called a test statistic from our observed data. Such an observed test statistic, $t_0$ is utilized as a rejection criterion given a particular significance level $\alpha$. The significance level $\alpha$, which is also known as a false alarm rate, measures the probability that we incorrectly reject true $H_0$, and hence we want it to be small. The most commonly used value of $\alpha$ is 0.05 and the significance level of 0.05 will also be used in this report.

   Lastly, we compute a p-value to decide whether we reject $H_0$ or not. p-values are computed in different ways depending on the statement of $H_1$ and types of hypothesis tests. For instance, in a general hypothesis test, if $H_1 : \mu > \mu_0$, where $\mu$ is a true population mean of a certain status, while $\mu_0$ is a null hypothesized mean, the p-value is calculated as the proportion of test statistics $T$ that are as ore more extreme than our observed test statistic $t_0$, under $H_0$ (i.e., $P(T \geq t_0)$). In such a case, we reject $H_0$ if and only if $P(T \geq t_0)$ is less than a pre-specified $\alpha$, and we do not reject $H_0$, otherwise.

2. Confidence intervals

   Another way of making statistical inferences is to use a confidence interval. Confidence intervals are intervals that are likely to capture a true population parameter at a certain percentage of level $c$. The value of $c$ is called critical value and is determined by the significance level $\alpha$. For example, if $\alpha = 0.05$, we use $c = 0.95$ and construct a 95% confidence interval.

More specific terminologies will be introduced in section 3 Results / Analysis.

# Resulsts / Analysis

## Data Cleaning

```r
# load necessary libraries
library(tidyverse)
library(janitor)
library(visdat)
library(cowplot)
library(naniar)
library(gendercoder)
library(grid)
library(gridExtra)

# Data Cleaning
data_path = "/Users/sanghyunkim/Desktop/University of Sydney/2021 Semester 2/DATA2002 Data Analytics Le
survey = readr::read_csv(data_path)

# 1. rename column names
short_colnames = c("time","covid_tests","living_arrangements","height",
                   "wednesday","in_aus","math_ability","r_ability",
                   "data2002","year","webcam","vaccination","social_media",
                   "gender","steak_preference","dominant_hand","stress",
                   "lonely","emails","sign_off","salary","unit","major","exercise")
colnames(survey) = short_colnames

# 2. 'height' data cleaning
## keep data consistency of 'height' column
survey = survey %>%
  dplyr::mutate(height = readr::parse_number(height),
                height = case_when(
                  height <= 2.5 ~ height * 100,
                  height <= 9 ~ NA_real_,
                  TRUE ~ height
                ))

# 3. 'wednesday' data cleaning
## remove any punctuation in the 'wednesday' column'
survey = survey %>% mutate(wednesday = stringr::str_replace_all(wednesday, "[[::pucnt::]]", ""),
                           wednesday = stringr::str_to_title(wednesday),
                           wednesday = fct_lump_n(wednesday, n = 2))

## check the new and clean 'wednesday' data
janitor::tabyl(survey, wednesday)

# 4. 'year' data cleaning
survey = survey %>%
  mutate(year = case_when(
   stringr::str_detect(tolower(year), "first") ~ "first-year",
   stringr::str_detect(tolower(year), "second") ~ "second-year",
   stringr::str_detect(tolower(year), "third") ~ "third-year",
   TRUE ~ NA_character_
  ))

## check the new and clean 'year' data
janitor::tabyl(survey, year)
```

```r
# 5. 'gender' data cleaning
survey = survey %>%
  mutate(
    gender = case_when(
      gender == "Woman/Female" ~ "female",
      TRUE ~ gender),
    gender = gendercoder::recode_gender(gender)
    )

## check the new and clean 'gender' data
janitor::tabyl(survey, gender)

# 6. 'living_arrangements' data cleaning
survey = survey %>% mutate(
  living_arrangements = fct_lump_n(living_arrangements, n = 5)
)
## check the new and clean 'living_arrangements' data
janitor::tabyl(survey, living_arrangements)

# 7. 'salary' data cleaning
survey = survey %>% mutate(salary = tolower(gsub("[.,$/? ]", "", salary)),
                           salary = case_when(
                             stringr::str_detect(salary, "^[a-z]") ~ NA_real_,
                             stringr::str_detect(salary,"-") ~ NA_real_,
                             stringr::str_detect(salary, "k") ~ as.numeric(parse_number(salary)) * 1000
                             stringr::str_detect(salary, "week") ~ as.numeric(parse_number(salary)) * 5
                             stringr::str_detect(salary, "pw") ~ as.numeric(parse_number(salary)) * 52,
                             stringr::str_detect(salary, "month") ~ as.numeric(parse_number(salary)) *
                             stringr::str_detect(salary, "pm") ~ as.numeric(parse_number(salary)) * 12,
                             stringr::str_detect(salary, "pa") ~ as.numeric(parse_number(salary)),
                             stringr::str_detect(salary, "year") ~ as.numeric(parse_number(salary)),
                             stringr::str_detect(salary, "aud") ~ as.numeric(parse_number(salary)),
                             stringr::str_detect(salary, "[[:alpha:]]") ~ NA_real_,
                             TRUE ~ as.numeric(salary)
                           ))
survey = survey %>%
  mutate(salary = case_when(
    salary <= 20000 ~ NA_real_,
    TRUE ~ salary
  ))

survey = survey %>% mutate(major = stringr::str_replace(major, "science", "Science"),
                           major = stringr::str_replace(major, "physics", "Physics"),
                           major = stringr::str_replace(major, "Analyst", "Analytics"),
                           major = stringr::str_replace(major, "analytics", "Analytics"),
                           major = stringr::str_replace(major, " \\s*\\([^\\)]+\\)", ""),
                           major = case_when(
                             stringr::str_detect(major, "an elective") ~ "Elective",
                             TRUE ~ major))
janitor::tabyl(survey, major)
```

## COVID-19 Tests Distribution Analysis

In statistics, knowing how well our sample data fit one of the well-known statistical distributions (e.g., Poisson distributions) enables us to model data, and thereby, have a reasonable expectation about what we are likely to find in the entire population. A chi-squared goodness-of-fit test is one type of hypothesis test where statisticians measure how well-observed data fit any given discrete distribution where the number of outcomes is discrete (i.e., integer-valued data). Therefore, in this analysis, we will perform a chi-squared goodness-of-fit test where $H_0$ claims that the observed counts of the number of times students have taken COVID-19 tests follow a Poisson distribution, while $H_1$ states that they do not.

Before conducting a chi-squared goodness-of-fit test, we excluded missing data to derive a meaningful result. As shown in Graph 3.2.1 below, there were initially 1.4% of missing data. Since we do not know whether these missing values represent 0 times or any other possible number of times students have taken COVID-19 tests, we will exclude those missing data. Graph 3.2.2 shows an approximate distribution of our observed sample data. As this histogram provides insufficient information about its underlying distribution, we tabulated expected counts for each corresponding number of times students have taken COVID-19 tests. (Table 3.2.1)

At this stage, it is noteworthy to check a significant assumption for chi-squared tests. Chi-squared tests work if and only if the expected frequencies are greater than or equal to 5, and the observations must be independent of each other. According to Table 3.2.1, although the observed frequencies for the corresponding number of times COVID-19 tests taken by students are independent of one another, the expected frequencies are less than 5 for some groups. Thus, we combine some of those groups to satisfy the chi-squared test assumption, and the result was tabulated in Table 3.2.2. Since the combined expected count is 4.3 for students who have taken COVID-19 tests four or more times, we classify the original data into four different groups as shown in Table 3.2.2.

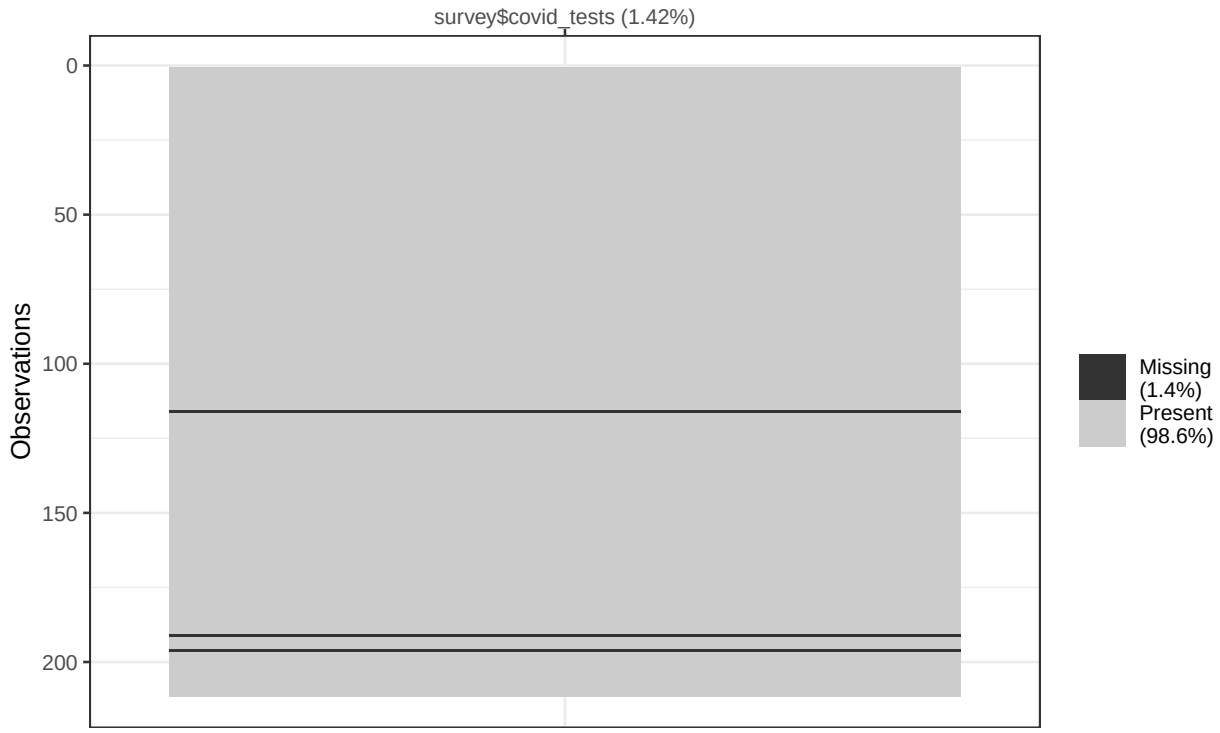
With this final observed counts data, we perform a chi-squared goodness-of-test, and the result is shown in Table 3.2.3. The observed test statistic is calculated from the observed data, and degrees of freedom ($df$) is determined by a formula:

$$df = k - 1 - q$$

where $k$ and $q$ represent the number of groups and the number of estimated parameters, respectively. As we have estimated a parameter $\lambda$ to compute Poisson distribution probabilities for each of those 4 groups, we subtract one degree of freedom, and therefore, the resulting degrees of freedom is $4 - 1 - 1 = 2$. The approximate p-value of 0 tells us the proportion of test statistics $T$ that are as or more extreme than the observed test statistics (70.91) is roughly 0, indicating that we reject $H_0$ at a significance level $\alpha = 0.05$. Such a result also matches Graph 3.2.3 that depicts a clear difference between observed and expected frequencies for each group.

Hence, we can conclude that the observed data does not follow a Poisson distribution. Nonetheless, as mentioned earlier, further studies are required with a larger-sized random sample data because our data is not randomly selected, and hence, is not representative of the entire population, the whole group of Australian students.

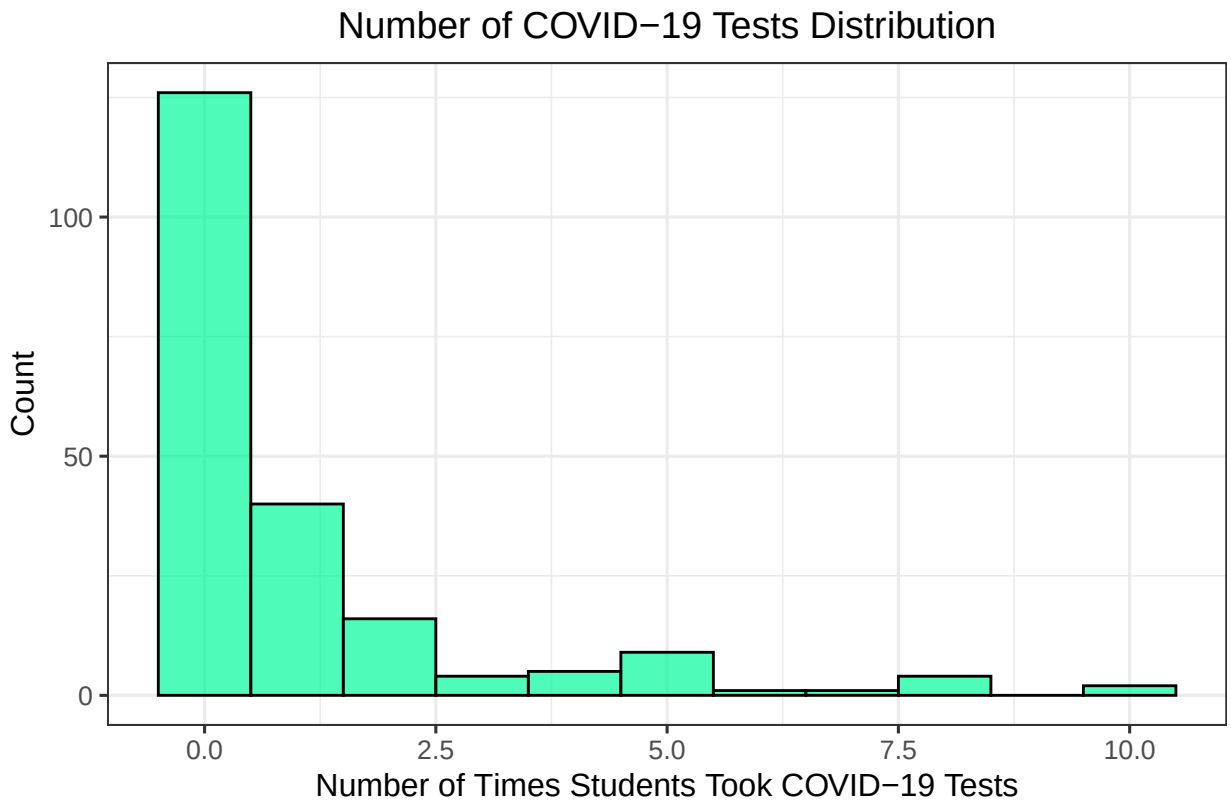
```
# explore missing data
covid_tests_df = tibble(survey$covid_tests)
vis_miss(covid_tests_df) +
  labs(title = "Missingness of 'covid_tests' Data",
       caption = "Graph 3.2.1") +
  theme_bw(base_size = 10) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10))
```




Graph 3.2.1

```
# exclude missing 'covid_tests' data
covid_tests = survey %>%
  filter(!is.na(covid_tests)) %>%
  select(covid_tests)

# visualize the number of times that students have taken COVID tests (without missing data)
covid_tests %>% ggplot() +
  aes(x = covid_tests) +
  geom_histogram(binwidth = 1, color = "black", fill = "mediumspringgreen", alpha = 0.7) +
  labs(x = "Number of Times Students Took COVID-19 Tests",
       y = "Count", title = "Number of COVID-19 Tests Distribution",
       caption = "Graph 3.2.2") +
  theme_bw(base_size = 12) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10))
```




Graph 3.2.2

```
# explore 'covid_tests' data without missing data
covid_tests = survey %>%
  group_by(covid_tests) %>%
  select(covid_tests) %>%
  count()

# vectorize 'covid_tests' data
y = covid_tests$n # number of observations for corresponding number of times students have taken COVID
x = covid_tests$covid_tests # unique number of times students have taken COVID tests

# since none of the students have taken COVID tests 9 times,
# add 0 to y, and 9 to x
y = c(y[1:9], 0, y[10])
x = c(x[1:9], 9, x[10])

## statistics required for a chi-squared test
n = sum(y) # total number of observations
k = length(y) # number of groups
estimated_lam = sum(y * x)/n # estimate lambda
p = dpois(x, estimated_lam) # obtain probabilities for each x value from Poisson pmf

# round up probabilities (4 decimal places)
p = round(p, 4)
# the probabilities sum to 1
sum(p)
```

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

```r
# compute expected counts
e = n * p

# check if all the expected counts are greater than or equal to 5 - a chi-squared test assumption
expected_counts_df = tibble(
  COVID_Tests = x,
  Expected_Counts = e)
knitr::kable(expected_counts_df, digits = 1, caption = "Table 3.2.1: Original Expected Counts")
```

Table 1: Table 3.2.1: Original Expected Counts

| COVID_Tests | Expected_Counts |
|---|---|
| 0 | 74.3 |
| 1 | 76.5 |
| 2 | 39.4 |
| 3 | 13.5 |
| 4 | 3.5 |
| 5 | 0.7 |
| 6 | 0.1 |
| 7 | 0.0 |
| 8 | 0.0 |
| 9 | 0.0 |
| 10 | 0.0 |

```r
# as some of the expected counts are not greater than or equal to 5,
# combine those expected counts (corresponding x = 3:10)
new_y = c(y[1:3], sum(y[4:11]))
new_p = c(p[1:3], sum(p[4:11]))
new_e = c(e[1:3], sum(e[4:11]))
new_k = length(new_y)
new_x = c("0", "1", "2", "3+")

# statistics required for a chi-squared test (a summary)
stats = tibble(
  COVID_Tests = new_x,
  Observed_Counts = new_y,
  Probabilities = new_p,
  Expected_Counts = round(new_e, 1),
)
knitr::kable(stats, caption = "Table 3.2.2: Final Statistics for a Chi-squared goodness-of-fit Test")
```

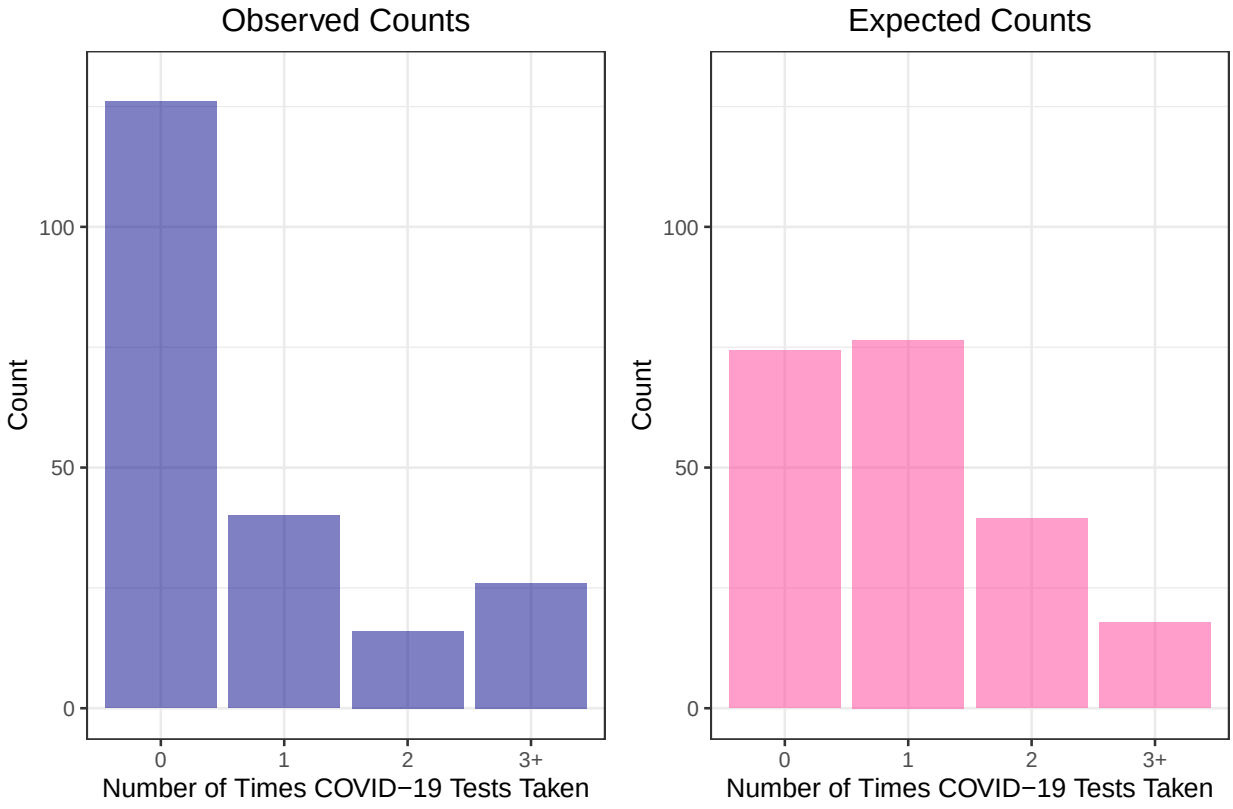
Table 2: Table 3.2.2: Final Statistics for a Chi-squared goodness-of-fit Test

| COVID_Tests | Observed_Counts | Probabilities | Expected_Counts |
|---|---|---|---|
| 0 | 126 | 0.3574 | 74.3 |
| 1 | 40 | 0.3677 | 76.5 |
| 2 | 16 | 0.1892 | 39.4 |
| 3+ | 26 | 0.0857 | 17.8 |

```r
# compare observed counts and expected counts
obs_counts = stats %>% ggplot() +
  aes(x = COVID_Tests, y = Observed_Counts) +
  geom_bar(stat = "identity", alpha = 0.5, fill = "darkblue") +
  labs(x = "Number of Times COVID-19 Tests Taken", y = "Count", title = "Observed Counts") +
  coord_cartesian(ylim = c(0, 130)) +
  theme_bw(base_size = 10) +
  theme(plot.title = element_text(hjust = 0.5))

exp_counts =stats %>% ggplot() +
  aes(x = COVID_Tests, y = Expected_Counts) +
  geom_bar(stat = "identity", alpha = 0.5, fill = "violetred1") +
  labs(x = "Number of Times COVID-19 Tests Taken", y = "Count", title = "Expected Counts") +
  coord_cartesian(ylim = c(0, 130)) +
  theme_bw(base_size = 10) +
  theme(plot.title = element_text(hjust = 0.5))

grid.arrange(obs_counts, exp_counts, nrow = 1,
             bottom = textGrob("Graph 3.2.3", hjust = 0.5))
```




Graph 3.2.3

```r
# compute the observed test statistic
t0 = sum((new_y - new_e)^2/new_e)

# compute a p-value (df = k - 1 - 1)
p_val = pchisq(t0, df = new_k - 1 - 1, lower.tail = FALSE)
```

```r
# chi-squared goodness-of-fit test result
chisq.result = matrix(data = c(t0, new_k - 1 - 1, p_val),
                      ncol = 3, byrow = TRUE)
rownames(chisq.result) = "Value"
colnames(chisq.result) = c("Observed Test Statistic", "df", "p-value")
knitr::kable(chisq.result, caption = "Table 3.2.3: Chi-squared Goodness-of-fit Test Result")
```

Table 3: Table 3.2.3: Chi-squared Goodness-of-fit Test Result

| | Observed Test Statistic | df | p-value |
|---|---|---|---|
| Value | 70.90979 | 2 | 0 |

## Stress Analysis: Are first-year university students more stressed out than second or third-year students?

Since the learning process at the university is different from studying at secondary school, one of the most challenging moments for first-year university students will be an adaptation to the university environment. In this analysis, we analyzed the existence of a difference in the level of stress across three different year-of-study groups: first-year, second-year, and third-year student groups. As the original quantified stress data varies from 0 to 10, we categorized these different scales into three groups: low (a stress score of $0 \sim 3$), moderate (a stress score of $4 \sim 6$), and high level of stress (a stress score of $7 \sim 10$) groups. Therefore, we will be comparing this categorized stress data based on different students' year-of-study status. Also, as presented in Graph 3.3.1, there was a small amount of missing data in both data (less than 1%), we omitted those missing data for accuracy purposes.

According to Graph 3.3.2, it appears that approximately 80 students are identified as low and high-stress groups, whereas only a half of students (approximate 40 students) are specified as a moderate stress group. To investigate how the year-of-study status affects the level of stress students feel, we visualized the proportions of different year-of-study statuses across those three stress categories. Graph 3.3.3 shows a clear trend that none of the first-year students are identified as a low-stress group, while most of those first-year students are under a high level of stress.

However, as Graph 3.3.2 does not show the exact number of students in each category, we also tabulated the observed counts for each category as shown in Table 3.3.1. With the same workflow as Section 3.2 above, the expected counts for each category are also tabulated (please refer to Table 3.3.2). Since the number of first-year students is small, we will also expect to see a small number of students categorized in each of those three stress groups.

In general, a chi-squared test for homogeneity is used to detect the existence of a difference in a certain outcome/feature status across different groups. However, since our data does not satisfy the assumption for chi-squared tests, we instead performed a Fisher's exact test. The basic idea about Fisher's exact test is computing the probability (p-value) that we observe contingency tables that are against $H_0$, when we enumerated all the possible permutations of contingency tables. In this example, as we are trying to detect whether first-year students are more stressed than second-year and third-year students, the null and alternative hypotheses of Fisher's exact test are the following:

$$H_0 : \theta = 1 \qquad VS \qquad H_1 : \theta > 1$$

where $\theta = Odds\ Ratio$, which measures the odds of first-year students being more stressed is greater than the odds that non-first-year students being more stressed.

According to Fisher's exact test result below, the p-value of 0.01733 indicates that we reject $H_0$, and thus, we have strong evidence that first-year students are more likely to feel stressed than other students. Clinciu (2013) has found a high negative correlation between stress and an adaptation to the university environment

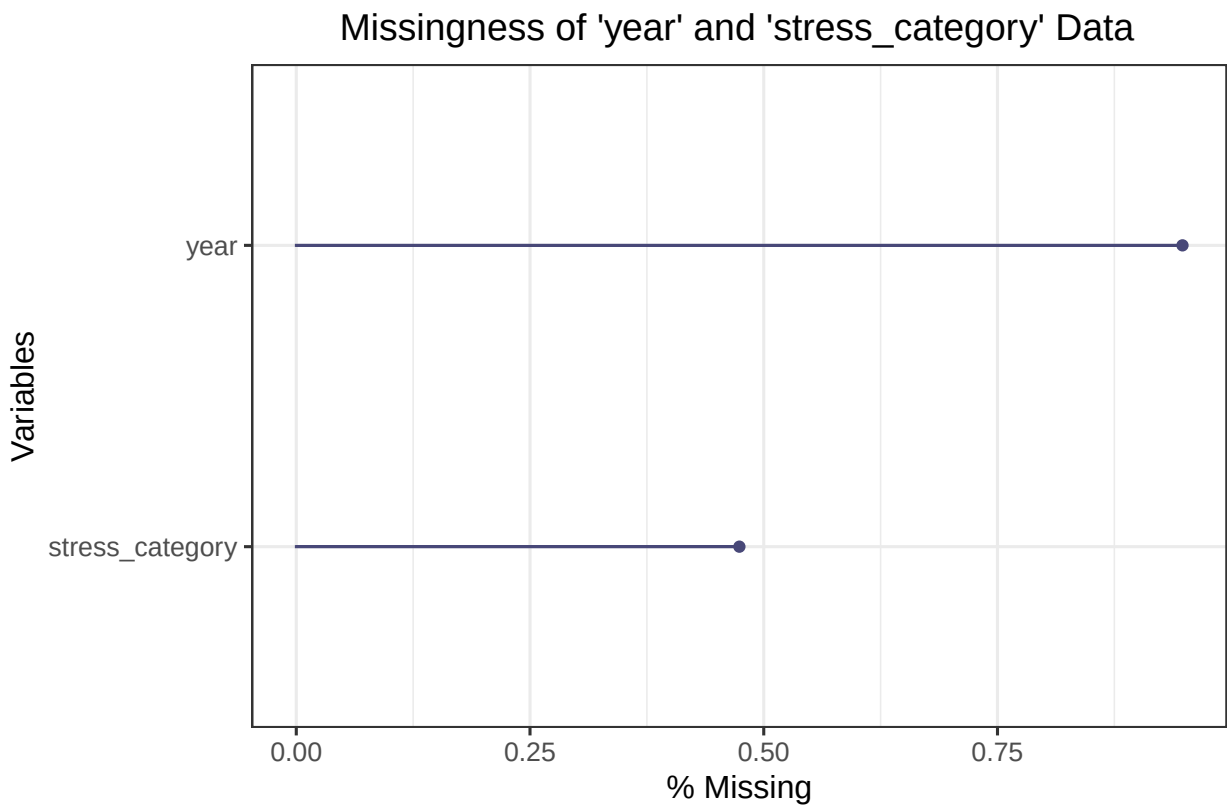
based on data obtained from 157 Transilvania University of Brasov students. In other words, as first-year students have trouble adapting to a new and more complicated university system, they are more likely to be stressed out.

Nevertheless, our analysis also has several pitfalls. First, it is critical to note that the courses DATA2002 and DATA2902 are designed for second-year students, and thus, the general proportions of first-year and third-year students in these two courses are relatively small. For this reason, further studies will yield a more reliable result if we conduct separate analyses over these three different year-of-status categories. Likewise, the result may differ if first-year students are categorized into in-depth groups. For instance, first-year arts students may feel a different level of stress from science or medical students, depending on their fields of study. Other factors also yield conflicting results. Hence, larger-sized random sample student data from various university faculties will enable us to provide more detailed and remarkable findings.

```
# since we are frequently using 'categorized' stress column data,
# create a new column named 'stress_category' in the original data
# 0 ~ 3: Low / 4 ~ 6: Moderate / 7 ~ 10: High
survey = survey %>%
  mutate(stress_category = case_when(
    is.na(stress) ~ NA_character_,
    stress >= 0 & stress <= 5 ~ "Low",
    stress >= 4 & stress <= 6 ~ "Moderate",
    stress >= 7 & stress <= 10 ~ "High"
  ))

survey$stress_category = factor(survey$stress_category, levels = c("Low", "Moderate", "High"))

# explore missing data for 'year' and 'stress' columns
year_stress = survey %>% select(year, stress_category)
gg_miss_var(year_stress, show_pct = TRUE) +
  labs(title = "Missingness of 'year' and 'stress_category' Data",
       caption = "Graph 3.3.1") +
  theme_bw(base_size = 12) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10))
```



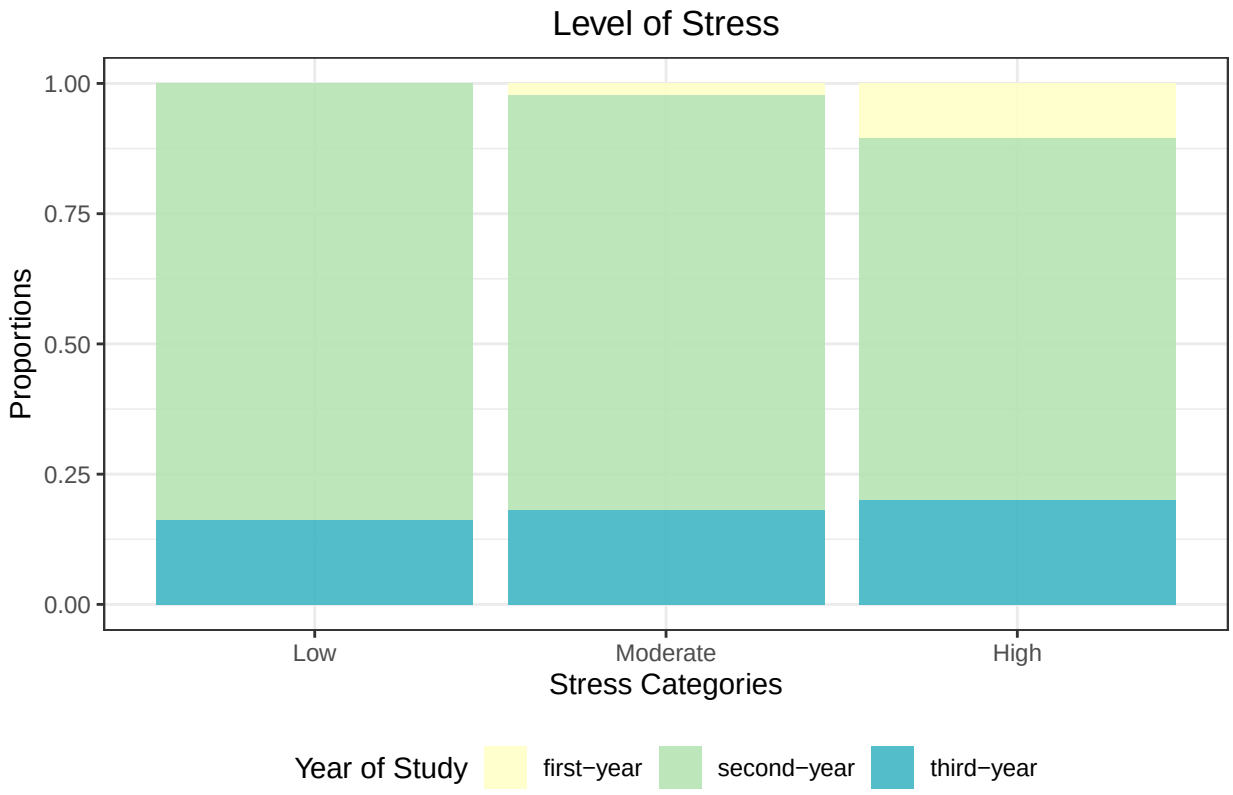

Graph 3.3.1

```
# take 'year' and 'stress_category' data out of the original data (excluding all the missing data from
year_stress = year_stress %>% filter(!is.na(year), !is.na(stress_category)) %>%
  select(year, stress_category)

# visualization of general 'stress_category' data
year_stress %>% ggplot() +
  aes(x = stress_category, fill = stress_category) +
  geom_bar(alpha = 0.7) +
  labs(x = "Stress Categories", y = "Counts",
       title = "Level of Stress", caption = "Graph 3.3.2") +
  scale_fill_brewer(palette = "YlOrRd",) +
  theme_bw(base_size = 11) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10),
        legend.position = "None")
```


Graph 3.3.2

```
# visualization of 'stress_category' data based on the students' 'year' status
year_stress %>% ggplot() +
  aes(x = stress_category, fill = year) +
  geom_bar(alpha = 0.9, position = "fill") +
  labs(x = "Stress Categories", y = "Proportions", title = "Level of Stress",
       caption = "Graph 3.3.3", fill = "Year of Study") +
  scale_fill_manual(values = c("#FFFFC8", "#B6E4B3", "#41B7C4")) +
  theme_bw(base_size = 11) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10),
        legend.position = "bottom")
```




Graph 3.3.3

```
# count the number of students in each corresponding category
year_stress_table = year_stress %>%
  group_by(year, stress_category) %>%
  count() %>%
  tidyr::pivot_wider(id_cols = year,
                     names_from = stress_category,
                     values_from = n) %>%
  mutate(Low = replace_na(Low, 0)) # replace NA with 0

# create a 3 × 3 contingency table (also reorder columns)
year_stress_mat = as.matrix(year_stress_table[, c(4, 2, 3)])
rownames(year_stress_mat) = c("first-year", "second-year", "third-year")

knitr::kable(year_stress_mat, caption = "Table 3.3.1: Contingency Table")
```

Table 4: Table 3.3.1: Contingency Table

| | Low | Moderate | High |
|---|---|---|---|
| first-year | 0 | 1 | 9 |
| second-year | 67 | 35 | 59 |
| third-year | 13 | 8 | 17 |

```
# check the expected counts of each cell
exp_year_stress = chisq.test(year_stress_mat, correct = FALSE)$expected
knitr::kable(round(exp_year_stress, 1), caption = "Table 3.3.2: Expected Contingency table")
```

Table 5: Table 3.3.2: Expected Contingency table

| | Low | Moderate | High |
|---|---|---|---|
| first-year | 3.8 | 2.1 | 4.1 |
| second-year | 61.6 | 33.9 | 65.5 |
| third-year | 14.5 | 8.0 | 15.5 |

```
# since some of the expected counts are less than 5 (first-year students)
# perform a Fisher's exact test
fisher.test(year_stress_mat, alternative = "greater")
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  year_stress_mat
## p-value = 0.01733
## alternative hypothesis: greater
```

## Stress Analysis: Does not staying in Australia affect how much Australian students are stressed out?

According to the Department of Education, Skills and Employment (2021), 342,656 international students are enrolled in Australian higher education programs in July 2021. However, the Australian government has closed its borders due to the current COVID-19 pandemic, and as a result, many international students are studying remotely in their home countries. Such a situation may affect the level of stress Australian university students may feel. Thus, we will analyze the relationship between students' study locations and the level of stress international students may feel. As in the previous analyses, the small amount of missing data (less than 1%) have been excluded in this analysis. (please refer to Graph 3.4.1)

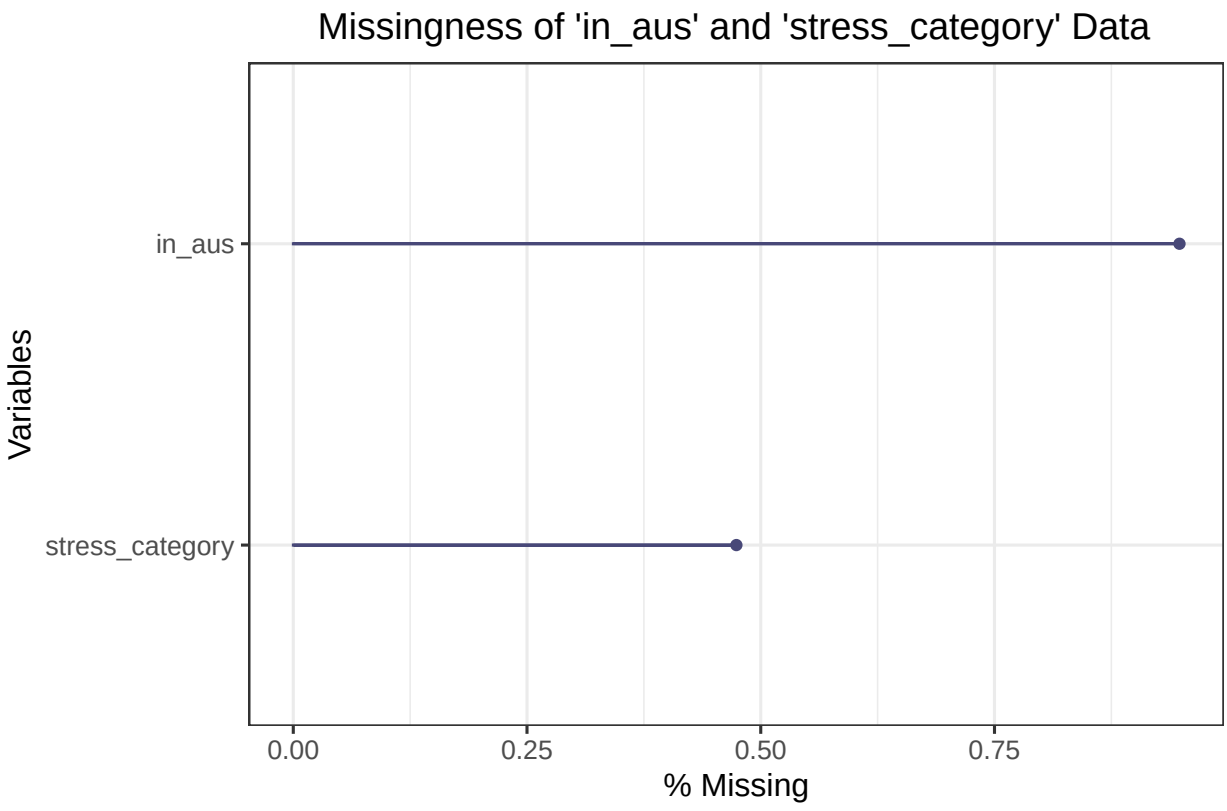
Graph 3.4.2 depicts that students who are currently staying outside Australia seem to be more stressed out, and such a clear trend also matches Table 3.4.1. Among 60 students staying outside Australia, 41 students are identified as a high-stress group, while 19 and 10 students are identified as low-stress and moderate-stress group, respectively. From Table 3.4.2, we can also find a similar pattern for these three different stress groups, based on students' staying-in-Australia status. Since there are more than five expected frequencies in each cell of Table 3.4.2, we conducted a chi-squared test for homogeneity, unlike the previous analysis. The hypotheses are the following:

$$H_0 : p_{11} = p_{21} \ \& \ p_{12} = p_{22} \ \& \ p_{13} = p_{23} \qquad VS \qquad H_1 : p_{11} \neq p_{21} \ \& \ p_{12} \neq p_{22} \ \& \ p_{13} \neq p_{23}$$

$p_{11}$ represents the proportion of the first row and first column cell in Table 3.4.1, while $p_{21}$ measures the proportion of the second row and first column cell, and the same logic applies to the rest notations, $p_{22}$, $p_{22}$, $p_{13}$ and $p_{23}$. This null hypothesis claims that there is no difference in the level of stress students feel regarding their current study locations. Thus, the alternative states that students' study locations affect the level of stress they feel. With a large observed test statistic $t_0 = 14.799$ we obtained a very small p-value of 0.0006116, indicating that we reject $H_0$, and thus, we have strong evidence that there is a clear difference in the level of stress students feel, depending on their staying-in-Australia status.

A recent survey conducted at Jordan University of Science and Technology found that approximately 32% of 1019 respondents reported that they are more stressed out with remote learning due to various factors such as online exams, or other technical issues. (Elsalem et.al., 2020). Nevertheless, as this sample data does not completely represent the whole group of international students studying in Australia, further studies based on randomly selected student data from different universities in Australia will yield a more insightful result.

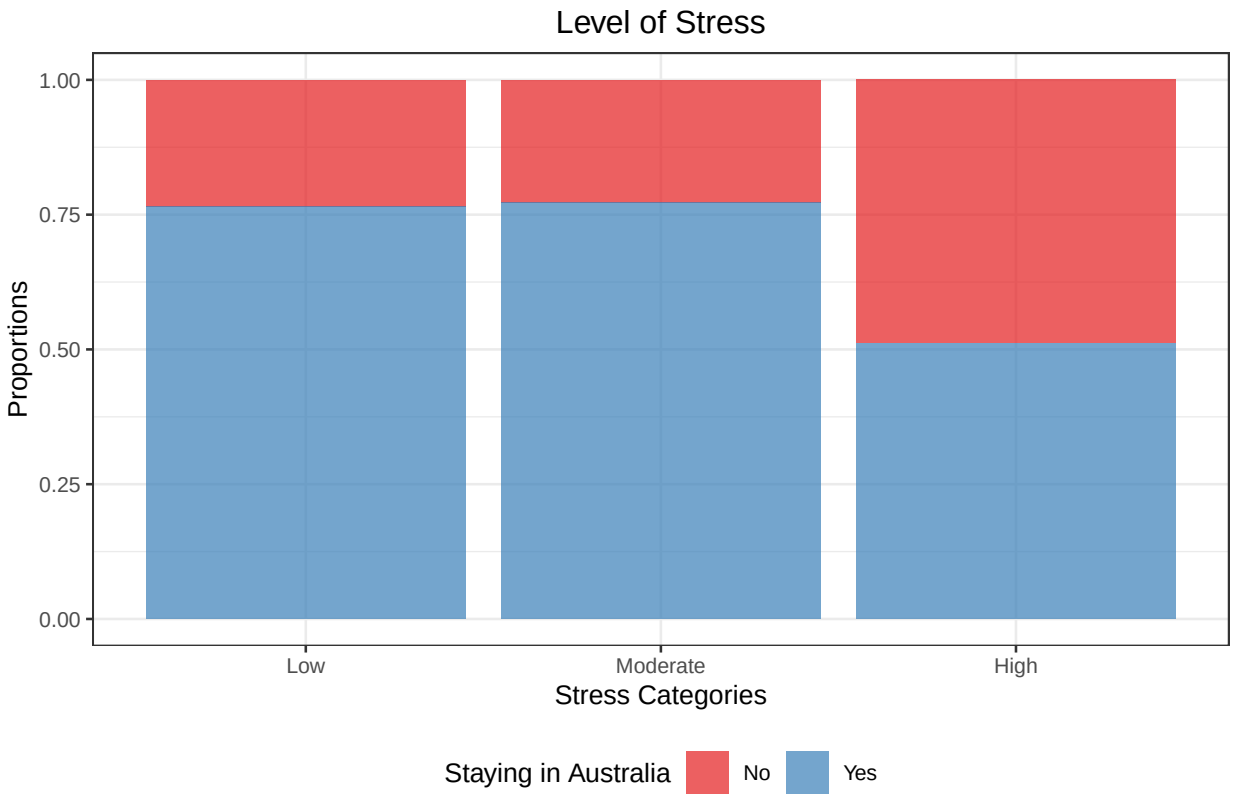
```r
# explore missing data
aus_stress = survey %>% select(in_aus, stress_category)
gg_miss_var(aus_stress, show_pct = TRUE) +
  labs(title = "Missingness of 'in_aus' and 'stress_category' Data",
       caption = "Graph 3.4.1") +
  theme_bw(base_size = 12) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10))
```




Graph 3.4.1

```r
# exclude all the missing data from both columns
aus_stress = aus_stress %>%
  filter(!is.na(in_aus), !is.na(stress_category))

# visualize stress_category data based on the staying status (in Aus vs not in Aus)
aus_stress %>% ggplot() +
  aes(x = stress_category, fill = in_aus) +
  geom_bar(alpha = 0.7, position = "fill") +
  labs(x = "Stress Categories", y = "Proportions", title = "Level of Stress",
       caption = "Graph 3.4.2", fill = "Staying in Australia") +
  scale_fill_brewer(palette = "Set1",) +
  theme_bw(base_size = 10) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10),
        legend.position = "bottom")
```




Graph 3.4.2

```
# count the number of students in each corresponding category
aus_stress_table = aus_stress %>%
  group_by(in_aus, stress_category) %>%
  count() %>%
  tidyr::pivot_wider(id_cols = in_aus,
                     names_from = stress_category,
                     values_from = n)

# create a 3 × 2 contingency table (also reorder columns)
aus_stress_mat = as.matrix(aus_stress_table[, c(2, 3, 4)])
rownames(aus_stress_mat) = c("No", "Yes")

knitr::kable(aus_stress_mat, caption = "Table 3.4.1: Contingency Table")
```

Table 6: Table 3.4.1: Contingency Table

| | Low | Moderate | High |
|---|---|---|---|
| No | 19 | 10 | 41 |
| Yes | 62 | 34 | 43 |

```
# check the expected counts of each cell
exp_aus_stress = chisq.test(aus_stress_mat)$expected
knitr::kable(round(exp_aus_stress, 1), caption = "Table 3.4.2: Expected Contingency Table")
```

Table 7: Table 3.4.2: Expected Contingency Table

| | Low | Moderate | High |
|---|---|---|---|
| No | 27.1 | 14.7 | 28.1 |
| Yes | 53.9 | 29.3 | 55.9 |

```
# as the contingency table above meets the chi-squared test assumption,
# perform a chi-squared test for independence
chisq.test(aus_stress_mat)
```

```
##
##  Pearson's Chi-squared test
##
## data:  aus_stress_mat
## X-squared = 14.799, df = 2, p-value = 0.0006116
```

### Stress Analysis: Are female students more stressed out than male students?

A gender difference in the level of stress has been long discussed. This analysis aims to analyze the impacts of gender on the level of stress university students feel. In addition to missing data shown in Graph 3.5.1, there is non-standard data values, "non-binary". Since there was only one "non-binary" response in the original data, we have also excluded this "non-binary" data values. Unlike the previous analyses conducted above, we did not categorize the original quantified stress data, as we aim to analyze how the magnitude of the difference in the quantified stress scale is between two gender groups: male and female students. Hence, the hypothesis test we performed in this analysis is a two-sample t-test.

A two-sample t-test is a hypothesis test used to measure the difference between two certain population parameters, (e.g., an average WAM of male and female students), where two sample groups are independent of each other. In this analysis, we are trying to see whether female students are more stressed out than male students, the hypotheses are the following:

$$H_0 : \mu_F = \mu_M \qquad VS \qquad H_1 : \mu_F > \mu_M$$

where $\mu_F$ and $\mu_M$ represent the average stress score of female and male students, respectively. Before performing a two-sample t-test, we must check an assumption for t-tests: independence and normality. Since the stress scores of female students does not affect the stress scores of male students (and vice versa), these two sample groups are independent of each other.

In terms of normality of data, as t-tests are conducted assuming that the underlying distribution of sample data are normally distributed (i.e. the distribution is bell-shaped and centered about its mean) we must check whether our sample data of both gender groups are normally distributed. One way of checking normality is to drawing box plots (Graph 3.5.2) and QQ plots (Graph 3.5.3). In Graph 3.5.2, each dot point represent a single observation in each group, while the horizontal line inside each box represents the average stress score for each gender group. Thus, we can see the dot points (single observations) are symmetric about the mean stress scores in both gender groups. On the other hand, in Graph 3.5.3, the x-axis represents theoretical quantiles, while the y-axis represents sample quantiles. If these sample quantiles match the theoretical quantiles, the dot points (single observations) will be roungly close to the straight line and we can say the data is normally distributed. As shown in Graph 3.5.3, it is also clear that the dot points are roughly close to the straight line in both groups. On top of that, if the sample size ($n$) of sample data is reasonably large enough, we can assure that the data is normally distributed relying on the *CentralLimitTheorem*, which states:

*As sample size increases the sample data will be approximately normal.*

Given the large sample size of each group in Table 3.5.2, as the sample size $n$, we can our sample data will be approximately normal, and hence, can perform a two-sample t-test.

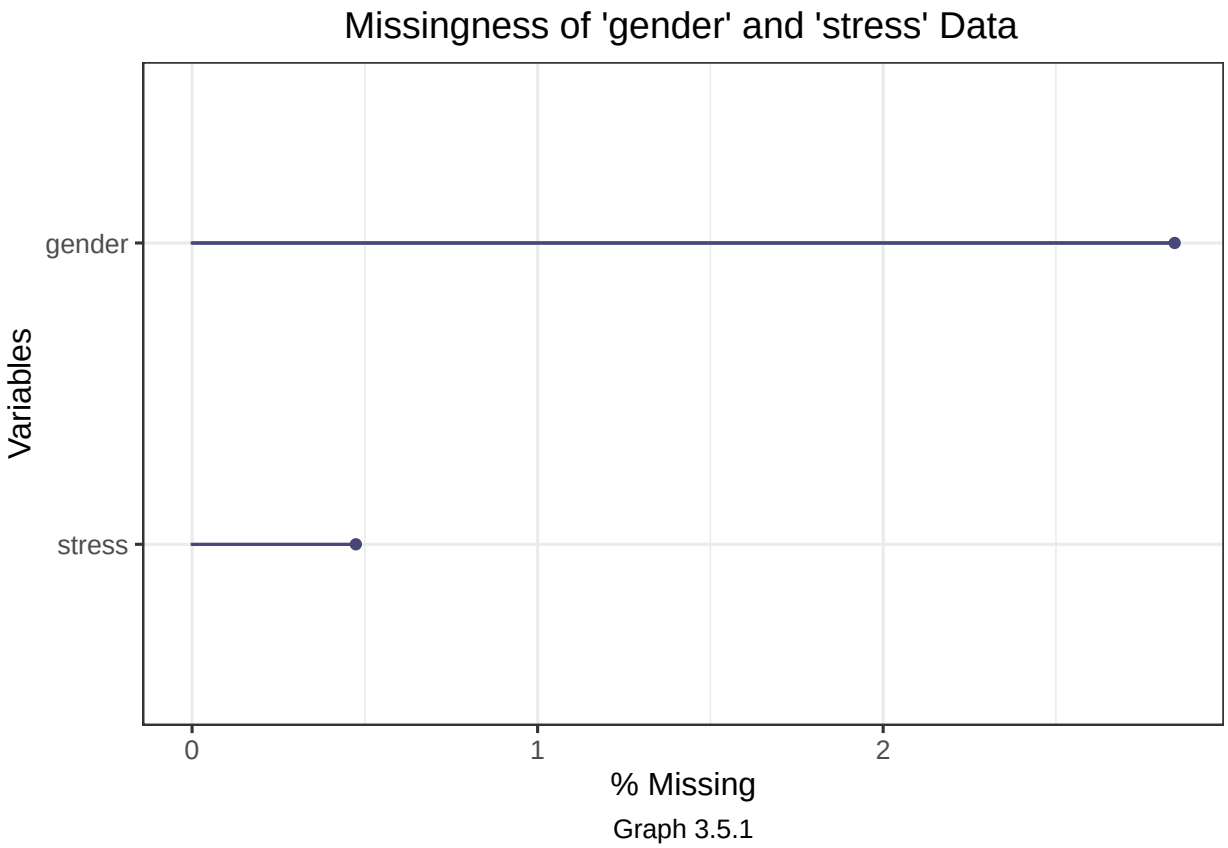
```
# explore missing data
gender_stress = survey %>%
  select(gender, stress)

knitr::kable(tabyl(survey, gender), caption = "Table 3.5.1: Gender Table")
```

Table 8: Table 3.5.1: Gender Table

| gender | n | percent | valid_percent |
|---|---|---|---|
| female | 75 | 0.3554502 | 0.3658537 |
| male | 129 | 0.6113744 | 0.6292683 |
| non-binary | 1 | 0.0047393 | 0.0048780 |
| NA | 6 | 0.0284360 | NA |

```
gg_miss_var(gender_stress, show_pct = TRUE) +
  labs(title = "Missingness of 'gender' and 'stress' Data",
       caption = "Graph 3.5.1") +
  theme_bw(base_size = 12) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10))
```



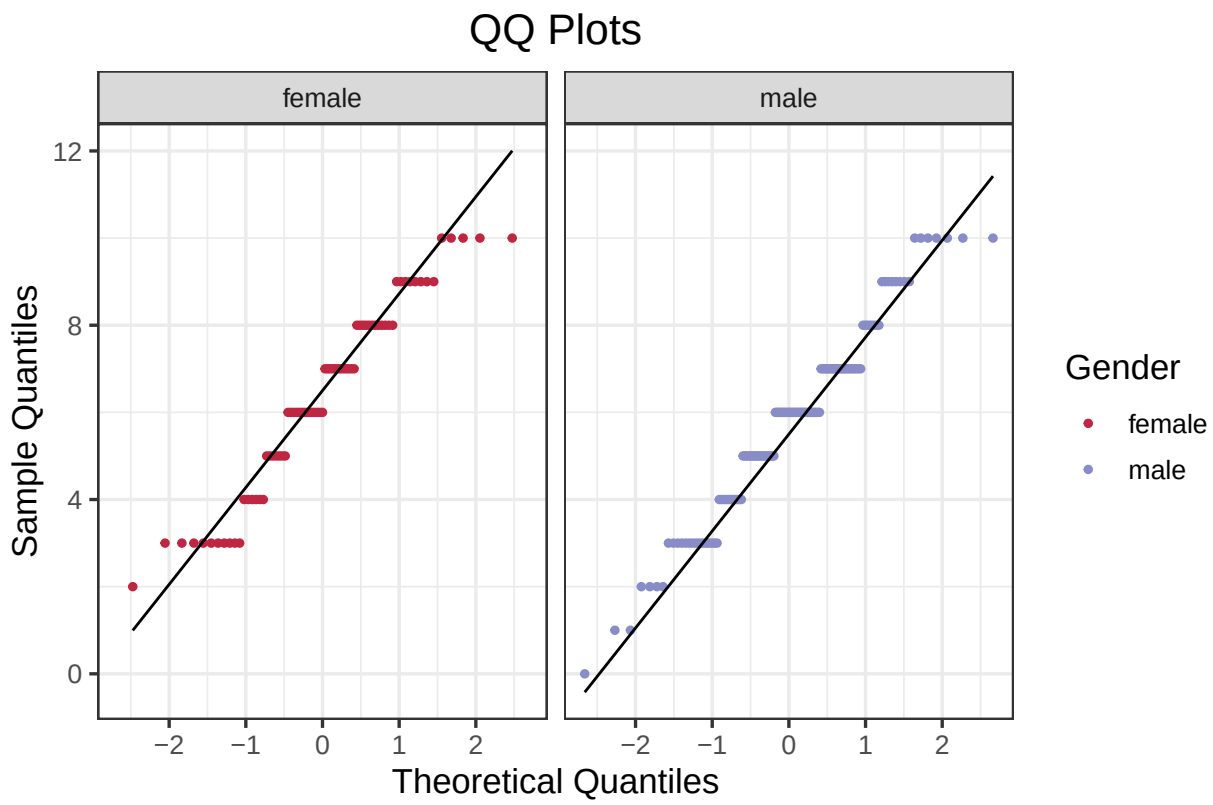
```r
# exclude missing data from both columns
# also exclude "non-binary" gender data
gender_stress = gender_stress %>%
  filter(!is.na(gender), !is.na(stress), gender != "non-binary")

# visualize stress data for two different gender category
gender_stress %>% ggplot() +
  aes(x = gender, y = stress, fill = gender) +
  geom_boxplot(alpha = 0.5, coef = 10) +
  geom_jitter(width = 0.15, size = 1, color = "black") +
  labs(x = "Gender", y = "Stress Scores", title = "Distribution of Stress Scores by Gender",
       caption = "Graph 3.5.2", fill = "Gender") +
  scale_fill_manual(values = c("#BE2641", "#888CC7")) +
  theme_bw(base_size = 12) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10))
```


Graph 3.5.2

```r
# normality check
gender_stress %>% ggplot() +
  aes(sample = stress, color = gender) +
  geom_qq(size = 1) +
  geom_qq_line(color = "black") +
  labs(x = "Theoretical Quantiles", y = "Sample Quantiles", title = "QQ Plots",
       caption = "Graph 3.5.3", color = "Gender") +
  scale_color_manual(values = c("#BE2641", "#888CC7")) +
  theme_bw(base_size = 13) +
```

```
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5, size = 10)) +
  facet_wrap(~gender)
```




Graph 3.5.3

```
# calculate average exercising hours per weekend across different stress category groups
stress_summary = gender_stress %>%
  group_by(gender) %>%
  summarize(Mean = mean(stress),
            SD = sd(stress),
            n = n())
knitr::kable(stress_summary, digits = 2, caption = "Table 3.5.2: Stress Score Statistics by Gender")
```

Table 9: Table 3.5.2: Stress Score Statistics by Gender

| gender | Mean | SD | n |
|---|---|---|---|
| female | 6.36 | 2.14 | 75 |
| male | 5.72 | 2.14 | 129 |

```
# given that the sample sizes of 76 and 129, the stress data for both genders are normally distributed
# hence, perform a two-sample t-test (one-sided)
female_stress = (gender_stress %>%
  filter(gender == "female"))$stress

male_stress = (gender_stress %>%
```

```
  filter(gender == "male"))$stress

t.test(female_stress, male_stress, alternative = "greater", var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  female_stress and male_stress
## t = 2.0564, df = 202, p-value = 0.02051
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.1255429       Inf
## sample estimates:
## mean of x mean of y
##   6.36000   5.72093
```

## Conclusion

## References