STA303/1002 Portfolio

An exploration of linear mixed models and common misconceptions in statistics

Aliza Aziz Lakho

2022-02-21

Contents

Introduction **3**

Statistical skills sample **4**

Task 1: Setting up libraries and seed value 4

Task 2a: Return to Statdew Valley: exploring sources of variance in a balanced experimental design (teaching and learning world) 4

Task 2b: Applying linear mixed models for the strawberry data (practical world) . . . 7

Task 3a: Building a confidence interval interpreter 8

Task 3b: Building a p value interpreter 9

Task 3c: User instructions and disclaimer 11

Task 4: Creating a reproducible example (reprex) 12

Task 5: Simulating p-values 13

Writing sample **17**

References 17

Reflection **19**

List of Figures

1 A figure caption 5

2 This is a figure caption 14

3 This is a figure caption 15

4 This is a figure caption 16

Introduction

This portfolio highlights the tools and techniques used for various tasks such as Exploring sources of variance in a balanced experimental design, applying linear mixed models, building a confidence interval and p-value interpreter, creating a reproducible example, and simulating p-values. This portfolio is also followed by a writing sample and a reflection.

In the very first task, we set up the necessary libraries for the entirety of the portfolio. In task 2 we focus on exploring sources of variance in a balanced experimental design, demonstrating calculation of sources of variance in a least-squares modelling context, and applying linear mixed models for the strawberry data. In task 3, we wrote a function that helps interpret a frequentist confidence interval and p-values based in the strength of evidence. In task 4, we demonstrate how to best reproduce our problem using reprex package in R. In task 5 we answer the question about the distribution the process whereby we simulate 10,000 sets of 10,000 normally distributed data points N(0,1) and perform one-sample t-test on each data set.

Towards the end of the document we proceed to a writing sample where we explore Motulsky (2014) with the intention of writing for our future self, and then end with a reflection on something specific that I am proud, something new that I have learned and demonstrated in this portfolio, and what is something I would do diffrenlty in the future. I hope you have a wonderful time going through this portfolio as much as I did creating it!

Statistical skills sample

Task 1: Setting up libraries and seed value

```
#Loading the tidyverse library
library(tidyverse)

#Loading the lme4 library
library(lme4)

#Defining last3digplus which is 100 + the last three digits of my student number
last3digplus <- 100 + 448
```

Task 2a: Return to Statdew Valley: exploring sources of variance in a balanced experimental design (teaching and learning world)

Growinng your (grandmother's) strawberry patch

```
# Sourcing it makes a function available
source("grow_my_strawberries.R")


# Altering the my_patch data so that treatment is a factor variable with the
#levels ordered as follows: "No netting," "Netting," "Scarecrow."

my_patch <- grow_my_strawberries(seed = last3digplus)

my_patch <- my_patch %>%
  mutate(treatment = fct_relevel(treatment, "No netting", after = 0))
```

Plotting the strawberry patch

```
#Plotting the strawberry patch
ggplot(my_patch, aes(x = patch, y = yield, fill= treatment, colour = treatment,
                     group = treatment)) +
  geom_point(pch = 25) +
```

```
scale_fill_manual(values = c("#78BC61", "#E03400", "#520048"))+
scale_color_manual(values = c("#78BC61", "#E03400", "#520048"))+
labs(caption = "Created by Aliza Aziz Lakho in STA303/1002, Winter 2022",
     tag = "Fig 2a: Exploring sources of variance")+
theme_minimal()+
theme(plot.tag.position = "bottom")
```

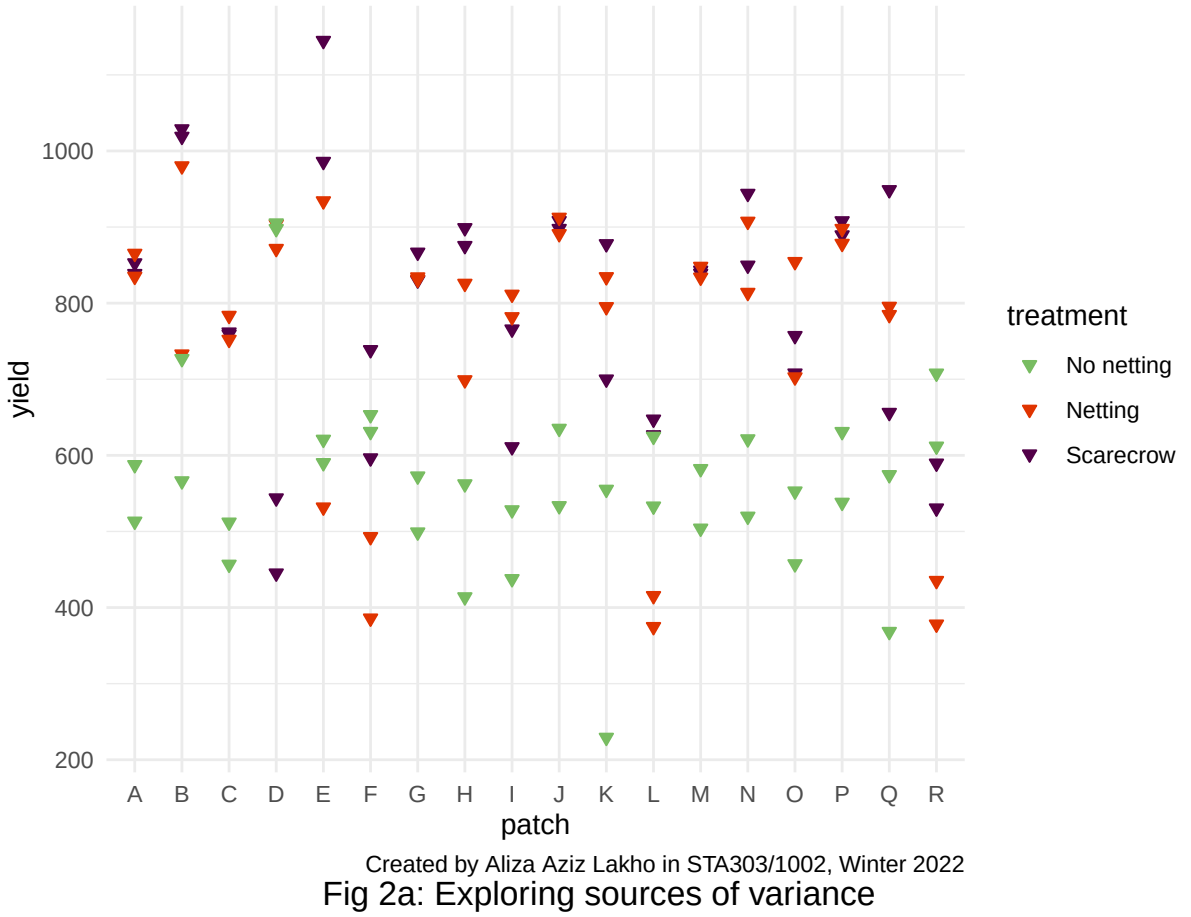

Figure 1: A figure caption

Demonstrating calculation of sources of variance in a least-squares modelling context

Model formula

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}$$

where: - y_{ijk} is the amount of yield in the k^{th} harvest at the j^th patch while receiving treatment i - μ is the mean of yield - α_i are the fixed effects of treatments, - β_j is the random effects for

patch j is a random effect of the interaction between patch and treatment ($\beta_j \sim N(0, \sigma^2_\beta)$) - $(\alpha\beta)_{ij}$, $(\alpha\beta)_{ij} \sim N(0, \sigma^2_{\alpha\beta})$), and - ϵ_{ijk} is the error term, ($\epsilon_{ijk} \sim N(0, \sigma^2)$)

```
#Wrangling data

#We want to find the average strawberry yield for each patch and treatment
#combination.
agg_int <- my_patch %>%
  group_by(patch, treatment) %>%
  summarize(yield_avg_int = mean(yield), .groups = "drop")


#We want to find the average strawberry yield for each patch
agg_patch <- my_patch %>%
  group_by(patch) %>%
  summarize(yield_avg_patch = mean(yield))
```

```
#Fitting the models

#We want to fit an interaction model to predict yield based on treatment and
#patch. We will call this int_mod
int_mod <- lm(yield ~ treatment + patch, data = my_patch)
#Next we want to find residual variance after fitting this linear model
var_int <- summary(int_mod)$sigma^2

#We will aslo create a main effects model, where yield_avg_int is the
#response and patch and treatment are the predictors.
agg_mod <- lm(yield_avg_int ~ treatment + patch, data = agg_int)
#Where Patches = 18 and Treatments= 3
K <- nrow(my_patch)/(18*3)
#Calculating variance in yield explained by the interaction of patch & treatment
var_ab <- summary(agg_mod)$sigma^2 - var_int / K

#Now we need one final aggregation and model which will give is patch-patch var
#So we create an intercept only model, where yield_avg_patch is the
#response. We will call this patch_mod
patch_mod <- lm(yield_avg_patch ~ 1, data = agg_patch)
#Calculating variance in average yield patch-to-patch,
#Note: No. of treatment = 3, hence $I = 3$
var_patch <- summary(patch_mod)$sigma^2 - (summary(agg_mod)$sigma^2)/3
```

```
tibble(`Source of variation` = c("Patch:Treatment",
                                 "Patch",
                                 "Residual variance"),
       Variance = c(var_ab, var_patch, var_int),
       Proportion = c(round(var_ab/(var_ab+var_patch+var_int), 2),
                      round(var_patch/(var_ab+var_patch+var_int), 2),
                      round(var_int/(var_ab+var_patch+var_int), 2) )) %>%
  knitr::kable(caption = "Summary of all our calculated error variances")
```

Table 1: Summary of all our calculated error variances

Source of variation	Variance	Proportion
Patch:Treatment	9293.332	0.31
Patch	1528.654	0.05
Residual variance	19022.238	0.64

Task 2b: Applying linear mixed models for the strawberry data (practical world)

```
#Creating the three models
mod0 <- lm(yield ~ treatment,  data = my_patch)

mod1 <- lmer(yield ~ treatment + (1 | patch),  data = my_patch)

mod2 <- lmer(yield ~ treatment + (1 | patch) + (1 | patch:treatment),  data =
↪  my_patch)

#Performing the likelihood test
lmtest::lrtest(mod0, mod1, mod2)
```

```
## Warning in modelUpdate(objects[[i - 1]], objects[[i]]): original model was of
## class "lm", updated model is of class "lmerMod"

## Warning in modelUpdate(objects[[i - 1]], objects[[i]]): original model was of
## class "lm", updated model is of class "lmerMod"

## Likelihood ratio test
```

```
##
## Model 1: yield ~ treatment
## Model 2: yield ~ treatment + (1 | patch)
## Model 3: yield ~ treatment + (1 | patch) + (1 | patch:treatment)
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   4 -695.26
## 2   5 -679.31  1 31.889  1.632e-08 ***
## 3   6 -665.12  1 28.398  9.877e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We believe that mod2 is the most appropriate final model.

Justification and interpreation

We have strong evidence against the hypothesis that the simpler model fits the data just as well.

Task 3a: Building a confidence interval interpreter

```
#A function that interprets confidence intervals
interpret_ci <- function(lower, upper, ci_level, stat){
  if(!is.character(stat)) {
    # produce a warning if the statement of the parameter isn't a character string
    # the spacing is a little weird looking so that it prints nicely in your pdf
    warning("
    Warning:
    stat should be a character string that describes the statistics of
    interest.")
  } else if(!is.numeric(lower)) {
    # produce a warning if lower isn't numeric
    warning("
    Warning:
    lower should be of a numeric data type that describes the lower bound
    pf the confidence interval.")
  } else if(!is.numeric(upper)) {
    # produce a warning if upper isn't numeric
    warning("
    Warning:
```

```
    upper should be of a numeric data type that describes the upper bound
    of the confidence interval.")
  } else if(!is.numeric(ci_level) | ci_level < 0 | ci_level > 100) {
    # produce a warning if ci_level isn't appropriate
    warning("
    Warning:
    ci_level represents the confidence interval level and it cannot be
    less than 0 or greater than 100.")
  } else{
    # print interpretation
  str_c("We are ", ci_level, "% confident that the ", stat, " is
        between ", lower, " and ", upper, "." )
  }
}

# Test 1
ci_test1 <- interpret_ci(10, 20, 99, "mean number of shoes owned by students")

# Test 2
ci_test2 <- interpret_ci(10, 20, -1, "mean number of shoes owned by students")

# Test 3
ci_test3 <- interpret_ci(10, 20, 95, 99)
```

CI function test 1: We are 99% confident that the mean number of shoes owned by students is between 10 and 20.

CI function test 2: Warning: ci_level represents the confidence interval level and it cannot be less than 0 or greater than 100.

CI function test 3: Warning: stat should be a character string that describes the statistics of interest.

Task 3b: Building a p value interpreter

```
# message=FALSE means we will not get the warnings
# A function that interprets p-values
interpret_pval <- function(pval, nullhyp){
  if(!is.character(nullhyp)) {
    warning("
```

```
    Warning:
    nullhyp should be a character string that describes the null
    hypothesis.")
  } else if(!is.numeric(pval)) {
    warning("
    Warning:
    pval should be of a numeric datatype that stores the p-value.")
  }  else if(pval > 1) {
    warning("
    Warning:
    pval cannot be greater than 1 since it represents the p-value.")
  } else if(pval < 0){
    warning("
    Warning:
    pval cannot be negative since it represents the p-value.")
  } else if(pval >= 0.1){
    str_c("The p value is ", round(pval, 3),
                ", and so we have no evidence against the null hypothesis that is ",
                ↪  nullhyp, ".")
  } else if(pval >= 0.05 & pval < 0.1){
    str_c("The p value is ", round(pval, 3),
                ", and so we have weak evidence against the null hypothesis thar is ",
                ↪  nullhyp, ".")
  } else if(pval >= 0.01 & pval < 0.05){
    str_c("The p value is ", round(pval, 3),
                ", and so we have moderate or some evidence against the null
                ↪  hypothesis that is ", nullhyp, ".")
  } else if(pval >= 0.001 & pval < 0.01){
    str_c("The p value is ", round(pval, 3),
                ", and so we have strong evidence against the null hypothesis that is
                ↪  ", nullhyp, ".")
  } else if(pval < 0.001){
    str_c("The p value is <.001",
                ", and so we have very strong evidence against the null hypothesis
                ↪  that is ", nullhyp, ".")
  }
}

pval_test1 <- interpret_pval(0.0000000003,
                             "the mean grade for statistics students is the same as
                             ↪  for non-stats students")
```

```
pval_test2 <- interpret_pval(0.0499999,
                             "the mean grade for statistics students is the same as
                             ↪  for non-stats students")

pval_test3 <- interpret_pval(0.050001,
                             "the mean grade for statistics students is the same as
                             ↪  for non-stats students")

pval_test4 <- interpret_pval("0.05", 7)
```

p value function test 1: The p value is <.001, and so we have very strong evidence against the null hypothesis that is the mean grade for statistics students is the same as for non-stats students.

p value function test 2: The p value is 0.05, and so we have moderate or some evidence against the null hypothesis that is the mean grade for statistics students is the same as for non-stats students.

p value function test 3: The p value is 0.05, and so we have weak evidence against the null hypothesis thar is the mean grade for statistics students is the same as for non-stats students.

p value function test 4: Warning: nullhyp should be a character string that describes the null hypothesis.

Task 3c: User instructions and disclaimer

Instructions

For the confidence interval interpreter make sure to hvae your statistics of interest "stat" as a string, upper and lower bounds should be numeric, and the ci_levels provided should be between 0 and 100. This interpreter takes stat, upper, lower, and ci_interavl to give you an interpretation. The upper and lower bounds give you the range an interval that has a certain confidence level (which is the probability) that the statistic of interest lies in that interval. For example take the ubsample of n=10 participants in the 7th examination of the Framingham Offspring Study. We get the interval for 113.3, 129.1 for 95% confidence level where our statistic is finding the true systilic blood pressure in the population. Hence we can say that we are 95% confident that the true systolic blood pressure in the given population is between 113.3 and 129.1,

For the p-value interpreter ensure that the pval provided is numeric and strictly between 0 and 1. Also note that the nullhyp provided should be a character string. This interpreter provides the strength of evidence, meaning that how weak of strong evidence we have against the null

hypothesis. Normally, the stronger the evidence against the null hypothesis, we reject it otherwise we fail to reject the null hypothesis.For instance, if the p-value of seeing a kangaroo at University of Toronto-St George is 0.01, and if you do see it than it makes this event significant but rare.

Disclaimer

Use the p-interpreter with caution since it takes raw p-values to determine the strength. For instance the difference between 0.0499999 and 0.050001 is not much but the strength of evidence for both are very different. This interpreter should be used to give a general idea about the strength of the evidence and not as a final answer.

Task 4: Creating a reproducible example (reprex)

Reprex is an example that when provided to someone else, they can reproduce it without any difficulty. In the reprex you provide the necesssary data, libraries, and just about anything that would help someone else reproduce the problem.

```
library(tidyverse)
my_data <- tibble(group = rep(1:10, each=10),
                  value = c(16, 18, 19, 15, 15, 23, 16, 8, 18, 18, 16, 17, 17,
                            16, 37, 23, 22, 13, 8, 35, 20, 19, 21, 18, 18, 18,
                            17, 14, 18, 22, 15, 27, 20, 15, 12, 18, 15, 24, 18,
                            21, 28, 22, 15, 18, 21, 18, 24, 21, 12, 20, 15, 21,
                            33, 15, 15, 22, 23, 27, 20, 23, 14, 20, 21, 19, 20,
                            18, 16, 8, 7, 23, 24, 30, 19, 21, 25, 15, 22, 12,
                            18, 18, 24, 23, 32, 22, 11, 24, 11, 23, 22, 26, 5,
                            16, 23, 26, 20, 25, 34, 27, 22, 28))

my_summary <- my_data %>%
  summarize(group_by = group, mean_val = mean(value))

glimpse(my_summary)
#> Rows: 100
#> Columns: 2
#> $ group_by <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3...
#> $ mean_val <dbl> 19.67, 19.67, 19.67, 19.67, 19.67, 19.67, 19.67, 19.67, 19.67...
```

Task 5: Simulating p-values

Setting up simulated data

```
# Setting the seed
set.seed(last3digplus)

#Creating the four simulated datasets with their values generated from N(0, 1), N(0.2,
↪  1), N(1,1), and Pois(5)
sim1 <- tibble(group = rep(1:1000, each = 100), val = rnorm(100000))

sim2 <- tibble(group = rep(1:1000, each = 100), val = rnorm(100000, mean=0.2, sd=1))

sim3 <- tibble(group = rep(1:1000, each = 100), val = rnorm(100000, mean=1, sd=1))

sim4 <- tibble(group = rep(1:1000, each = 100), val = rpois(100000, lambda=5))

#Combining all the datasets together
all_sim <- bind_rows(sim1, sim2, sim3, sim4, .id = "sim")

#Creating the labels for each simulation
sim_description <- tibble(sim = 1:4,
                          desc = c("N(0, 1)",
                                   "N(0.2, 1)",
                                   "N(1, 1)",
                                   "Pois(5)"))
#Changing the data type of sim to numeric from string
all_sim <- all_sim %>%
  mutate(sim = as.numeric(sim))

#Adding the labels for each simulation
all_sim <- left_join(all_sim, sim_description)
```

```
## Joining, by = "sim"
```

```
# Visualizing the first three groups
all_sim %>%
  filter(group <= 3) %>%
  filter(sim <= 3) %>%
  ggplot(aes(x = val)) +
```

```
  geom_histogram(bins = 40) +
  facet_wrap(desc~group, nrow = 3) +
  theme_minimal() +
  labs(caption = "Created by Aliza Aziz Lakho in STA303/1002, Winter 2022",
       tag = "Fig 5a: Exploring simulated datasets")+
  theme(plot.tag.position = "bottom")
```

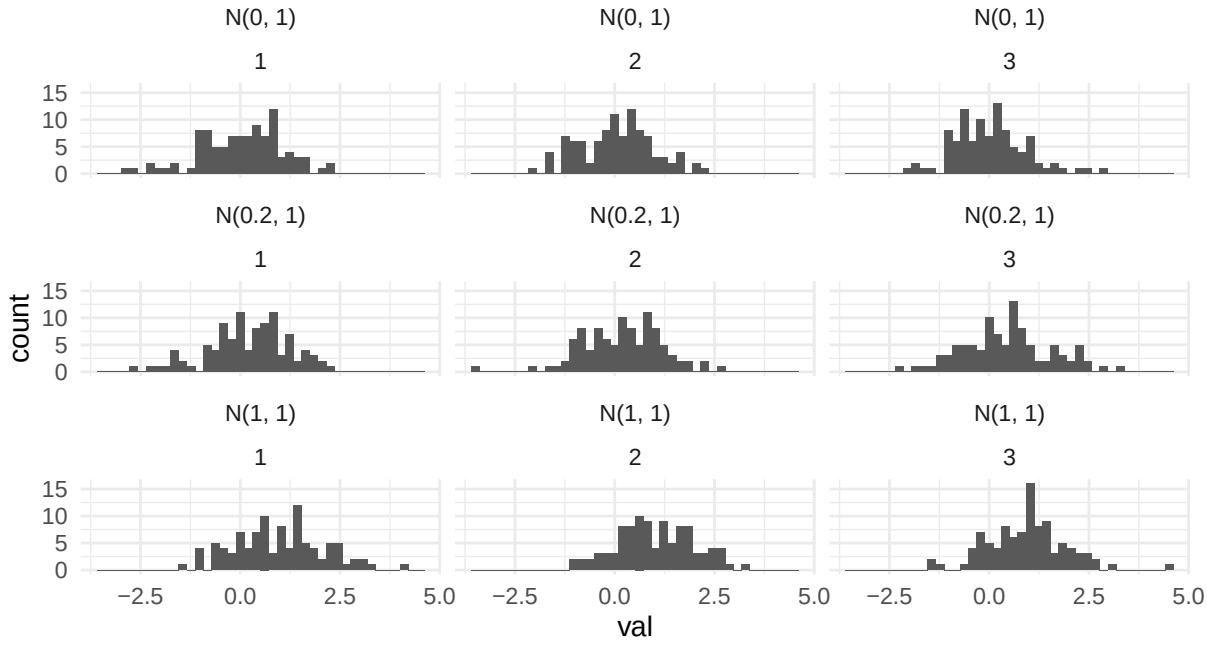

Fig 5a: Exploring simulated datasets

Figure 2: This is a figure caption

Calculating *p* values

```
#Generate a dataset that contains p_values for groups in all_sim
pvals <- all_sim %>%
  group_by(desc, group) %>%
  summarize(pval= t.test(val, mu=0)$p.value, .groups="drop")
```

```
#Visualize the histograms of p-values
pvals %>%
```

```
  filter(desc != "Pois(5)") %>%
  ggplot(aes(x = pval)) +
  geom_histogram(boundary = 0, binwidth = 0.05, fill = "grey", color = "black") +
  xlim(0,1)+
  facet_wrap(~desc, scales = "free_y") +
  theme_minimal() +
  labs(caption = "Created by Aliza Aziz Lakho in STA303/1002, Winter 2022",
       tag = "Fig 5b: Exploring p-value histograms")+
  theme(plot.tag.position = "bottom")
```

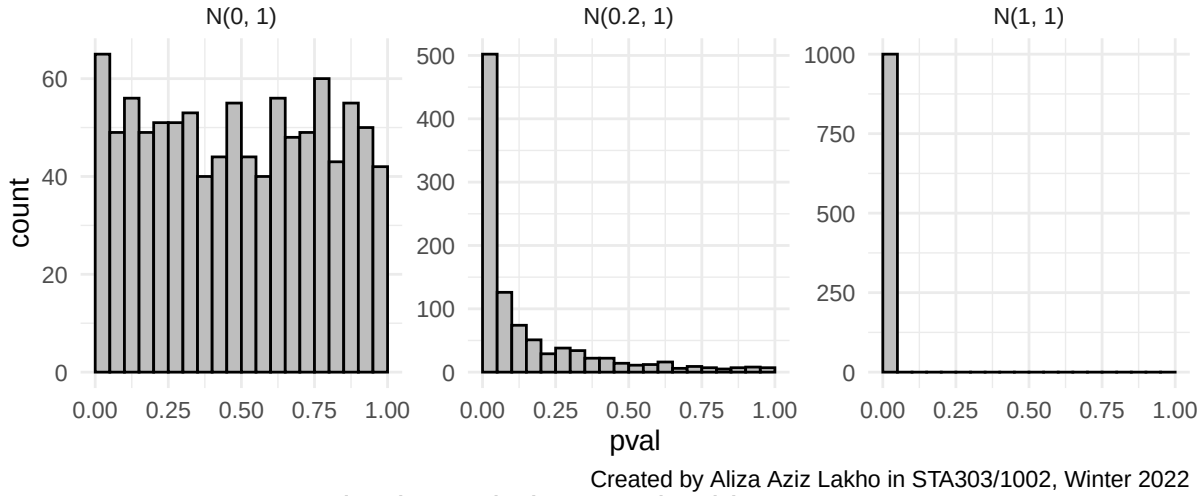

Fig 5b: Exploring p−value histograms

Figure 3: This is a figure caption

Drawing Q-Q plots

```
#Drawing QQ Plots
pvals %>%
  filter(desc != "Pois(5)") %>%
  ggplot(aes(sample = pval)) +
  geom_qq(distribution = stats::qunif) +
  geom_abline(intercept = 0, slope = 1) +
  facet_wrap(~desc) +
  theme_minimal() +
  labs(caption = "Created by Aliza Aziz Lakho in STA303/1002, Winter 2022",
```

```
      tag = "Fig 5c: QQ Plots")+
theme(plot.tag.position = "bottom")
```

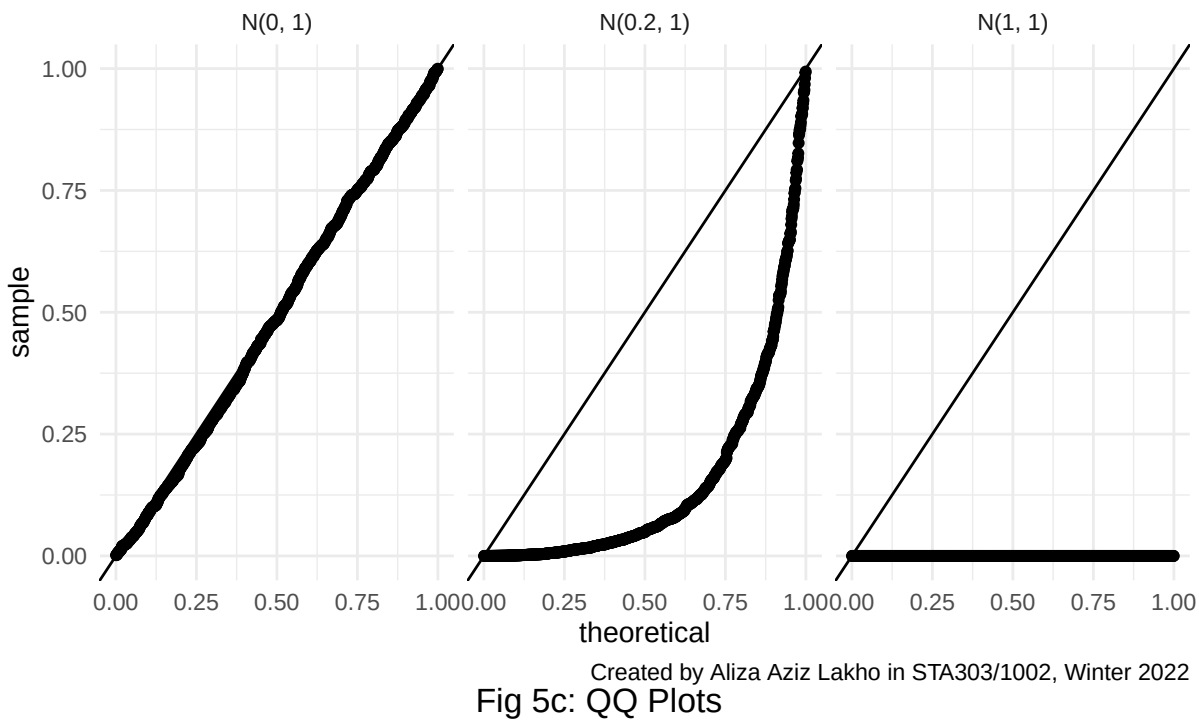

Figure 4: This is a figure caption

Conclusion and summary

This brings us to the summary that when we simulate 10,000 sets of 10,000 normally distributed data points N(0,1) and perform one-sample t-test on each data set, we get a uniform distribution. This is represented best by the QQ plots. Hence we can conclude that Approximately 10% of the p-values will be between 0.9 and 1.

Writing sample

As Motulsky (2014) states that in the case that you have to statistically analyse data, you should be clear with the processes and results. That is to say that every step of data collection, wrangling, analysis, presentation should be done with the intention of presenting complete findings seriously and without any bias. Motulsky writes a brilliant piece on five different misconceptions associated with statistics and analysis and how to avoid them. I find it very interesting how the diagrams and animations have been incorporated to show that P-hacking is not okay, P values do not convey information about effect size, the importance of statistical significance, the need to report the details, and that the standard error of mean does not quantify variability.

I, myself, am aware with how P-hacking is not okay. I agree that to satisfy one's bias, we can force our biased results by reanalyzing. This is problematic since once we reach our desired result, we do not look at anything else and just report the results. I am also aware that as someone who does statistical analysis with the intention of publishing the results, the details should be reported completely. This further helps others to reproduce our results and shows transparency in our work. I also agree that getting a statistical significance is not necessary for experimental research. I believe that forcing one self to produce a statistical significance result is equivalent to P-hacking.

It was surprising for me to learn that the standard error of the mean does not quantify variability. Motulsky best demonstrates this by mentioning that sometimes with large samples where we have a lot of variability, the standard error mean can be very small. Motulsky suggests the best way to demonstrate variability is making available the raw data in some graphical form. The other piece of information that I did not realise is how the P values do not convey information about the effect size. This, as I understand, is because the P values represent the probability of seeing an effect's size if the null hypothesis is true but it gives no information about the size of the effect itself.

This article has helped me reinforce the misconceptions associated with statistics and has helped me to understand further on how to avoid them. Lastly, I firmly believe that, as Motulsky states, "any experienced investigator with the right tools should be able to reproduce a finding published in a peer-reviewed biomedical science journal."

Word count: 409 words

References

Motulsky, H. J. (2014). Common misconceptions about data analysis and statistics. *Naunyn-Schmiedeberg's Archives of Pharmacology*, *387*(11), 1017–1023. https://doi.org/10.1007/s0

> 0210-014-1037-6

Reflection

What is something specific that I am proud of in this portfolio?

I am very proud of working on this portfolio with very limited help due to the circumstance that required me to take an extensions. I am very glad that I was able to go through it all by myself. This has made me more confident in my experience in R and in general with my understanding of statistic. Even though, limited help meant getting frustrated at the code that did not work for longer, I am very proud that just by going through piazza I was able to solve all the issues that I had with the code.

How might I apply what I've learned and demonstrated in this portfolio in future work and study, after STA303/1002?

I have learned a great deal about following instructions to produce desired results. I believe this is very important to study and then work in the future. This can be easily applied in the my studies here since each assignment comes with a rubric highlighting the expectations of the assignment. With understanding how to follow instructions, I can work on my future assignments knowing how to do my level best. This can also has applications in the work industry as at the end of the day one has to meet certain expectations set by the employers.

What is something I'd do differently next time?

For next time, I would perhaps try to skim the entire assignment before hand and to prepare the kind of questions that I might have and ask them before all the office hours end. This is important as this can save me so much of time which I can spend perfecting some other part of the assignment. Even though, there was a time management section at the top of the portfolio, I believe if I would have taken that into more consideration, I would not have had so much problem completing the assignment.