BIOL/BOT 297 Final Project

Chris Muir

2020-05-14

This document explains what needs to go into the final project, how it will be graded, and *R* code chunks from an example project.

For the **Project Plan** (due April 24), complete a draft of sections 1–3.

You will submit an *R* script (see example-script.R) and all data files necessary to run your script for your final project. Separately, you will also present your results for a lay audience with a short written description, in the format of a blog post, and a short (5-10 minute) powerpoint-style presentation. I will send instructions on the blog post and presentation separately.

Read all instructions carefully

This rubric therefore sets the minimum requirements and explains how points will be allocated during grading. However, I will allow well justified flexiability if your analysis doesn't exactly meet these requirements. **It is important to discuss with me in advance if your project will be significantly different from this rubric.** With that said, every project should fulfill all the requirements, at least approximately, but may do so in different ways. Some projects may require additional steps not mentioned here, which you will determine with me as your project develops.

The project is due via email by the end of the exam period, Friday May 14

Grading

21 points

- You won't lose points for trying something new, but taking a 'kitchen sink' approach won't help you. Keep it concise!
- For the final, there will be additional points for your written (blog post) and oral presentations. I will send details on those separately.

Import data

You will need to import and process your data prior to analysis. The code below is a bit advanced, so you may take a different approach. You can also process data outside *R* and import the processed data.

```
library(dplyr)
library(ggplot2)
library(scales)

# This imports covid-19 cases by state and summarizes the total number of cases
# per state and creates a new variables with the log-transformed value.
covid19 <- read.csv("states-daily.csv", stringsAsFactors = FALSE) %>%
  group_by(state) %>%
  summarize(positive = max(positive, na.rm = TRUE)) %>%
  mutate(log_positive = log(positive))
```

```r
# Data on state governor political party
state_gov <- read.csv("state-governors.csv", stringsAsFactors = FALSE)

# Join datasets
dat <- full_join(covid19, state_gov, by = "state") %>%
  filter(!is.na(gov_party))

# Extract values for D and R states
D_pos <- dat %>%
  filter(gov_party == "D") %>%
  pull(log_positive)

R_pos <- dat %>%
  filter(gov_party == "R") %>%
  pull(log_positive)
```

1. Question(s) (1 point)

A relatively simple but well-formed biological question about why COVID-19 spreads at different rates.

```r
# Does the political party of the governor affects a state's public health
# response?
```

2. Hypotheses (3 points)

- Scientific hypothesis and prediction. The hypothesis should be a statement or proposition that if true, would generate a testable prediction. I'm just grading for completion since we haven't covered scientific (as opposed to statistical) hypotheses in detail.

  If propsition X is true, I predict outcome Y

```r
# If political party effects public health policy, this could affect the number
# of COVID-19 cases in the state.
```

- Statistical hypotheses (null and alternatives hypotheses, 1 point each). Statistical hypotheses are narrower and more specific than scientific hypotheses. Your null hypothesis (H_0) should state a specific value that would be interesting to reject. The alternative hypothesis (H_A) should follow from your null hypothesis. They should be in the form of:
  - H_0: the value of Y is _____
  - H_A: the value of Y is not _____

```r
# H0: The number hospitalizations due to COVID-19 will on average be the same in
# states with Democratic and Republican governors.

# HA: The number hospitalizations due to COVID-19 will on average NOT be the
# same in states with Democratic and Republican governors.
```

3. Briefly explain what test you will use and why (2 points)

For full credit, you must identify an appropriate test (there may be more than one correct answer, see Interleaf 7 in the text). However, if you chose the wrong test, I will not penalize you in subsequent sections as long as your answers follow logically. For example, if you incorrectly use an ANOVA rather than a linear regression, I will not penalize you again for using the F-distribution to plot the sampling distribution.

```r
# My explanatory variable is categorical (political party) and my response
# variable is numerical. Since I have only two categories, I will use a
# two-sample t-test.
```

4. Plot the sampling distribution under the null hypothesis (3 points)

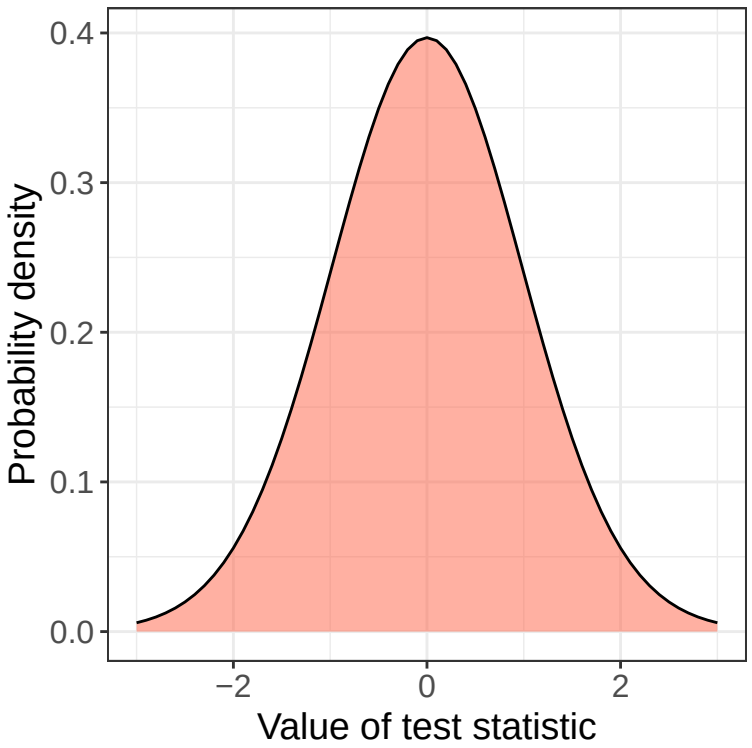
2 points for correct choice and calculation; 1 point for good graphical principles

```r
# The sampling distribution under the null hypothesis is the t-distribution with 4
# 48 degrees of freedom

sample_dist <- data.frame(Y = seq(-3, 3, 0.1))
sample_dist$probability <- dt(sample_dist$Y, df = 48)

sample_dist_plot <- ggplot(sample_dist, aes(Y, probability)) +
  geom_area(fill = "tomato", alpha = 0.5) +
  geom_line() +
  xlab("Value of test statistic") +
  ylab("Probability density") +
  theme_bw() +
  theme(
    axis.title = element_text(size = 14),
    axis.text = element_text(size = 12)
  )

sample_dist_plot
```



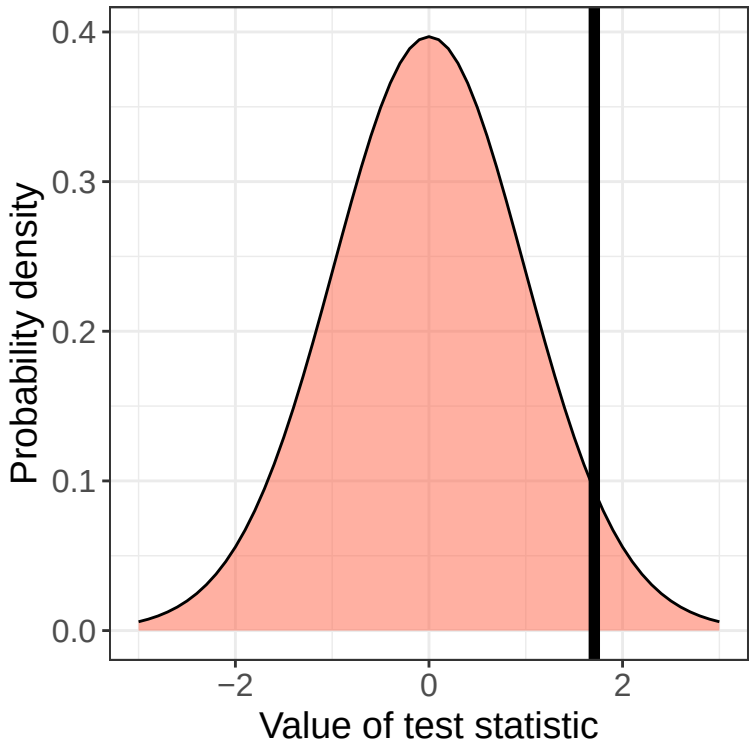
5. Compute the test statistic (3 points)

- Calculate the test statistic "by hand" (i.e. show your calculations in *R* as we did in the code demos). 1 point for choosing the appropriate test statistic, 1 point for calculating correctly. You should be able to check with the appropriate *R* function to see if your calculation is correct.
- Add the test statistic to your plot above (1 point).

```
Ybar_1 <- mean(D_pos)
Ybar_2 <- mean(R_pos)
n_1 <- length(D_pos)
n_2 <- length(R_pos)
s2_1 <- var(D_pos)
s2_2 <- var(R_pos)
df_1 <- n_1 - 1
df_2 <- n_2 - 1

s2_p <- (df_1 * s2_1 + df_2 * s2_2) / (df_1 + df_2)
SE_dY <- sqrt(s2_p * (1 / n_1 + 1 / n_2))
test_stat <- (Ybar_1 - Ybar_2) / SE_dY

sample_dist_plot +
  geom_vline(xintercept = test_stat, size = 2)
```



6. Calculate parameter estimates and confidence intervals (3 points)

As with the test statistic, you should do this "by-hand" and show your *R* calculations, but you can compare your result with the appropriate a *R* function. The exact parameter and confidence interval method will depend on which test you are using. If there are multiple parameters for a given test, you can just choose one to show here.

You need to:

1. Choose an appropriate parameter to estimate and *briefly explain your choice in writing* (1 point).
2. Estimate the parameter correctly (1 point).
3. Calculate a confidence interval correctly (there may be more than one valid method, 1 point).

Partial credit will be awarded if, for example, you misidentify the parameter but estimate it correctly.

```
# I will estimate the difference in mean number of COVID-19 cases (log-scale)
# between states with Democratic and Republican governors.

Ybar_1 - Ybar_2
```

```
## [1] 0.7017495
```

```
# parameter estimate is 0.70
alpha <- 0.05
(Ybar_1 - Ybar_2) + qt(alpha/2, df_1 + df_2) * SE_dY
```

```
## [1] -0.1243187
```

```
(Ybar_1 - Ybar_2) - qt(alpha/2, df_1 + df_2) * SE_dY
```

```
## [1] 1.527818
```

```
# 95% confidence interval is -0.12 - 1.52
```

7. Calculate P-value (1 point)

As with the test statistic and confidence intervals, you should do this "by-hand" and show your calculations in R, but you can easily compare your result to that using an R function.

```
2 * pt(test_stat, df = df_1 + df_2, lower.tail = FALSE)
```

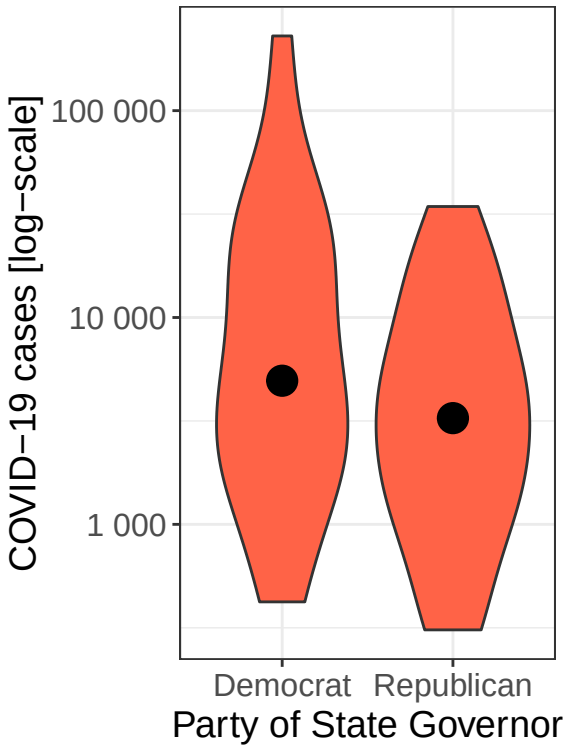
```
## [1] 0.09408847
```

8. Graph your data (3 points)

- Based on your explanatory and response variables, which type of graph did you choose and why? See W&S Chapter 2, 7-9 for guidance. There may be more than one correct answer. *You need to provide a concise written justification* (1 point).
- Make a graphic from your data that follows good graphical principles as described in W&S (2 points).

```
# I chose a violin because the explanatory variable is categorical and the
# response variable is numeric

ggplot(dat, aes(gov_party, positive)) +
  geom_violin(fill = "tomato") +
  stat_summary(fun = "median", geom = "point", size = 5) +
  scale_x_discrete(labels = c("Democrat", "Republican")) +
  scale_y_log10(labels = label_number()) +
  xlab("Party of State Governor") +
  ylab("COVID-19 cases [log-scale]") +
  theme_bw() +
  theme(
    axis.title = element_text(size = 14),
    axis.text = element_text(size = 12)
  )
```



9. Draw the appropriate conclusions (2 points)

Address the following questions *with written responses* for both tests:

1. What is your significance level (α)? (1 point)
2. Based on your P-value and α, do you reject the null hypothesis? (1 point)

```
# Using an alpha of 0.05, I fail to reject the null hypothesis that the party
# affiliation of the state's governor has an effect on the number of COVID-19
# cases in that state.
```