ProbSet 6, February 20

Gideon Gordon gideongordon2029@u.northwestern.edu

Due: February 20, 2026

1 Setup

```
library(estimatr)
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.5.2
```

2 Problem 1

2.1 1a. Using the democracy and internet access data from the lecture:

```
# Load data and run models
library(rqog)
qogts <- read_qog(which_data = "standard", data_type = "time-series")
```

```
Local file not found.
Downloading QoG qog_std_ts_jan23.csv data
from http://www.qogdata.pol.gu.se/data/qog_std_ts_jan23.csv
```

```
in file: /var/folders/qq/1hygc6fn77544gc6rc0j9q600000gn/T//RtmpM2HZGc/rqog/qog_std_ts_ja

Reading cache file /var/folders/qq/1hygc6fn77544gc6rc0j9q600000gn/T//RtmpM2HZGc/rqog/qog
```

```r
# Model 1: Basic model with homoskedastic errors
model1 <- lm(vdem_libdem ~ wdi_broadb + I(log(wdi_gdpcappppcon2017)),
             data = qogts)

# Model 2: Robust standard errors
model2 <- lm_robust(vdem_libdem ~ wdi_broadb + I(log(wdi_gdpcappppcon2017)),
             data = qogts)

summary(model1)
```

```
Call:
lm(formula = vdem_libdem ~ wdi_broadb + I(log(wdi_gdpcappppcon2017)),
    data = qogts)

Residuals:
     Min       1Q   Median       3Q      Max
-0.70660 -0.15630  0.02562  0.16594  0.46510

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                  -0.2152498  0.0408603  -5.268 1.48e-07 ***
wdi_broadb                    0.0076919  0.0004347  17.694  < 2e-16 ***
I(log(wdi_gdpcappppcon2017))  0.0636083  0.0046176  13.775  < 2e-16 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2188 on 3035 degrees of freedom
  (12328 observations deleted due to missingness)
Multiple R-squared:  0.3295,    Adjusted R-squared:  0.3291
F-statistic: 745.9 on 2 and 3035 DF,  p-value: < 2.2e-16
```

```
summary(model2)
```

```
Call:
lm_robust(formula = vdem_libdem ~ wdi_broadb + I(log(wdi_gdpcappppcon2017)),
    data = qogts)

Standard error type:  HC2

Coefficients:
                              Estimate Std. Error t value  Pr(>|t|)  CI Lower
(Intercept)                  -0.215250  0.0470851  -4.572 5.036e-06 -0.307572
wdi_broadb                    0.007692  0.0004827  15.935 5.775e-55  0.006745
I(log(wdi_gdpcappppcon2017))  0.063608  0.0055782  11.403 1.597e-29  0.052671
                              CI Upper   DF
(Intercept)                  -0.122928 3035
wdi_broadb                    0.008638 3035
I(log(wdi_gdpcappppcon2017))  0.074546 3035

Multiple R-squared:  0.3295 ,   Adjusted R-squared:  0.3291
F-statistic:  1150 on 2 and 3035 DF,  p-value: < 2.2e-16
```

Questions: 1. Extract and compare the standard errors. Which are larger?

```r
coef(summary(model1))[, "Std. Error"]
```

```
                (Intercept)                   wdi_broadb
               0.0408602509                 0.0004347115
I(log(wdi_gdpcappppcon2017))
               0.0046176007
```

```r
coef(summary(model2))[, "Std. Error"]
```

```
                (Intercept)                   wdi_broadb
               0.0470851200                 0.0004827035
I(log(wdi_gdpcappppcon2017))
               0.0055781698
```

```r
se_broadb1 = coef(summary(model1))[, "Std. Error"][2]
se_log_gdp_per_cap1 = coef(summary(model1))[, "Std. Error"][3]
se_broadb2 = coef(summary(model2))[, "Std. Error"][2]
se_log_gdp_per_cap2 = coef(summary(model2))[, "Std. Error"][3]
```

The standard errors for the latter, the model with robust standard errors, are substantially larger.

2. Calculate the t-statistics manually (i.e., using a calculator or direct math in R and not just summary):

I'm doing a t-test here. $t = \ _0/se()/n$

Beta-0 is our null hypothesis that the coefficient is zero, so this boils down to $/se()$.

```r
beta_wdi_broadb1 = coef(summary(model1))[2,1]
beta_log_gdp_per_cap1 = coef(summary(model1))[3,1]
```

```
beta_wdi_broadb2 = coef(summary(model2))[2,1]
beta_log_gdp_per_cap2 = coef(summary(model2))[3,1]


t_broadband_1 = beta_wdi_broadb1 / se_broadb1
t_broadband_2 = beta_wdi_broadb2 / se_broadb2


t_log_gdp_per_cap1 = beta_log_gdp_per_cap1 / se_log_gdp_per_cap1
t_log_gdp_per_cap2 = beta_log_gdp_per_cap2 / se_log_gdp_per_cap2
```

So what I get out of this is a metric of how many standard errors our values are from zero. That's handy.

3. Compute p-values for both sets of standard errors.

```
p_broadband_1 = 2*pt(t_broadband_2, 3035, lower.tail = FALSE)
p_broadband_2 = 2*pt(t_broadband_2, 3035, lower.tail = FALSE)
p_log_gdp_per_cap1 = 2*pt(t_log_gdp_per_cap1, 3035, lower.tail = FALSE)
p_log_gdp_per_cap2 = 2*pt(t_log_gdp_per_cap2, 3035, lower.tail = FALSE)
print(p_broadband_1)
```

```
  wdi_broadb
5.774558e-55
```

```
print(p_broadband_2)
```

```
  wdi_broadb
5.774558e-55
```

```
print(p_log_gdp_per_cap1)
```

```
I(log(wdi_gdpcappppcon2017))
                6.392551e-42
```

```
print(p_log_gdp_per_cap2)
```

```
I(log(wdi_gdpcappppcon2017))
                1.596556e-29
```

On an initial try, my p-values were all 1, which seems wrong. I tried again to fix the earlier problems with my t-statistics. After confirming I had done that part right (as best I could), I realized my p-values had been calculated with `lower.tail = TRUE`, which is incorrect. Ultimately I got the values listed above.

4. At = 0.05, would you reject the null hypothesis for each coefficient in both models?

   Yes, I would reject the null hypothesis. All the p-values are extremely small (likely as small as R can show).

2.2 1b. The lecture notes that with small samples and normal errors, we use the t-distribution.

What is the degrees of freedom for the t-distribution in your model?

In our model here, we had 3,035 degrees of freedom.

How do t-critical values compare to z-critical values for your sample size? How much difference does it make to choose the t versus normal distribution at this sample size?

We did not discuss t-critical and z-critical values in class or in the readings, so I'm not sure precisely, but I think at this sample size there's probably not much difference. We have a sample of about 3,000 and only 3 parameters, which is enough that the differences between the t-distribution and the normal distribution should be more or less washed out.

3 Problem 2

3.1 2a.

1. In your own words, explain the multiple comparisons problem.

Let's say I want to prove that I can control the weather with my Rainstorminator machine. After enough experiments, I finally get a statistically significant relationship between the weather I evoked on Day N using the Rainstorminator and the weather that came on Day N+30. My confidence buoyed by this statistically significant result, I go and sell my services to drought-stricken nations worldwide.

The problem is that if you do enough tests, you will eventually get a statistically significant result. With $= 0.05$, you have a one in twenty shot, every time, of getting statistical significance purely at random. So if the weather is truly random, I will still be able to find a statistically significant relationship at least once if I run my experiment enough times.

This also applies to testing many hypotheses at once, not just the same hypothesis many times in sequence. I have a regression testing a complicated model of civil war outbreak including thirty variables — GDP per capita, weather patterns on the day shooting began, history of coups, colonial history, colors on the national flag, and the height of the President (which I believe to be inversely correlated with the President's likelihood of surviving a coup attempt, because they'll be a bigger target that way). With so many variables, even if I have enough observations to test them all, I will also be very likely to find one or two correlations by pure chance, since again, there's a 1/20 chance of randomly getting a significant result using the current normal threshold of significance. Similarly, if I run many models with slightly different specifications to check for robustness, I may get significance on one model rather than another by pure chance.

I would argue that this problem is not so much inherent in the math as part of the procedure for interpreting and presenting data. A p-value does not "know" that it is being tested alongside twenty other p-values necessarily. So I can say, from the outset, “I controlled for all these things, but really all I care about is testing the President’s height.” In this case, I’m only running one hypothesis test, and the multiple comparisons problem is not an issue, because I don’t care about the significance of anything but the main variable. Still, if I test multiple models, I would want to be cautious.

2. Define:

- Family-Wise Error Rate (FWER) The FWER is the chance that *any* significant correlation you find is spurious.

- False Discovery Rate (FDR) The FDR is the share of false discoveries to discoveries overall. So it tells me what share of my findings are probably spurious, but it’s more forgiving than calculating the FWER, for which even a single spurious correlation is too many.

3.2 2b. Simulate the multiple comparisons problem:

```r
set.seed(123)
n_tests <- 20
n_obs <- 100
alpha <- 0.05

# Create matrix of 20 independent tests (all null true)
p_values <- matrix(NA, nrow = 1000, ncol = n_tests)

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs)
    y <- rnorm(n_obs)  # No relationship

    # Run regression and extract p-value for slope
    p_values[i, j] <- summary(lm(y ~ x))$coefficients[2, 4]
  }
}

# Analyze results
false_positives <- rowSums(p_values < alpha)
mean_fp <- mean(false_positives)
prop_at_least_one <- mean(false_positives > 0)
```

Questions: 1. What **proportion of simulations have at least one false positive**? This is the Family-Wise Error Rate.

The proportion of simulations with at least one false positive is given by `prop_at_least_one`, which is 0.637.

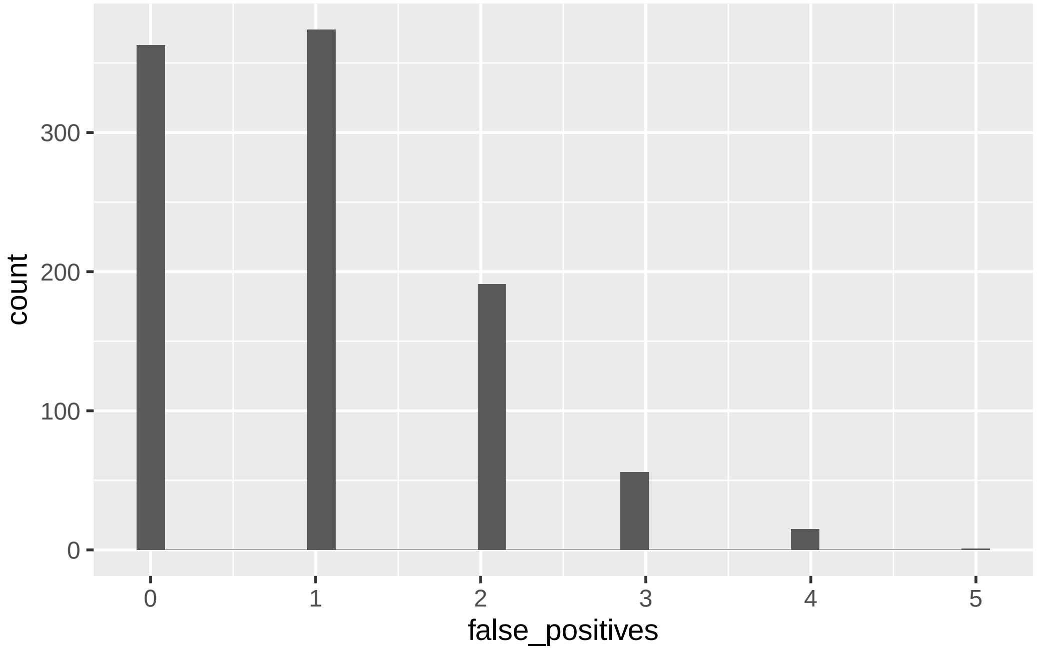
2. What is the **average number of false positives** per simulation? This is the False Discovery Rate.

To be clear the false discovery rate is equal to false positives divided by the total number of significant results. Since *every* significant result in this simulation is a false positive, the false discovery rate is expected to be basically 100%. And so yes, in this case, we see that the false discovery rate is 0.989, which is close enough to 100%.

3. **Create a histogram** of the number of false positives across simulations.

```r
false_positives = data.frame(false_positives)
ggplot(data=false_positives, mapping = aes(x = false_positives)) + geom_histogram()
```

```
`stat_bin()` using `bins = 30`. Pick better value `binwidth`.
```



Gonna graph this in terms of how many tests have how many false positives. What we see is that out of the twenty tests which *ought to* be failed, most simulations show at least one test failing.

3.3 2c. Now simulate correlated tests:

```
# Create correlated predictors
set.seed(123)
n_tests <- 20
n_obs <- 100

# Generate correlated X matrix
library(MASS)
mu <- rep(0, n_tests)
Sigma <- diag(n_tests)
for (i in 1:n_tests) {
```

```r
  for (j in 1:n_tests) {
    Sigma[i, j] <- 0.7^abs(i-j)  # AR(1) correlation
  }
}

p_values_cor <- matrix(NA, nrow = 1000, ncol = n_tests)

for (i in 1:1000) {
  # Generate correlated predictors
  X <- mvrnorm(n_obs, mu, Sigma)
  # Generate Y with no relationship to any X
  y <- rnorm(n_obs)

  # Run all regressions
  for (j in 1:n_tests) {
    p_values_cor[i, j] <- summary(lm(y ~ X[, j]))$coefficients[2, 4]
  }
}

# Compare with independent case
false_positives_cor <- rowSums(p_values_cor < alpha)
mean_fp_cor <- mean(false_positives_cor)
prop_cor <- mean(false_positives_cor > 0)
```

Questions: 1. How does correlation affect the multiple comparisons problem?

```r
prop_at_least_one
```

```
[1] 0.637
```

```r
prop_cor
```

```
[1] 0.51
```

The number of false positives goes down slightly. Correlations between the variables somewhat reduce the issue because there are effectively multiple checks for each test; if one p-value was randomly high, its correlation with other predictors would pull it back towards the expected value of those predictors, which is zero. (I don't know if that makes sense, please advise).

4 Problem 3

4.1 3a. Find a published political science article that reports multiple hypothesis tests (e.g., a table with many coefficients and stars).

A paper I evaluated was Haer et al. 2013, "Analyzing the microfoundations of human violence in the DRC - intrinsic and extrinsic rewards and the prediction of appetitive aggression" from *Conflict and Health*. This paper claims that the causes of the psychological consequences of participation in war (specifically, an ex-combatant's degree of "appetitive aggression", their non-instrumental enjoyment of perpetrating violence) vary depending on how a fighter was recruited. For abductees, being offered extrinsic rewards for their participation (mostly the opportunity to loot) correlated with higher appetitive aggression (as measured post-war). For volunteers, social closeness to other former soldiers correlated with higher appetitive aggression. I have my qualms about this study because they don't really present much of a causal story to explain the results they find.

1. Count how many hypothesis tests are reported.

Initially two hypotheses were tested (the effect of intragroup closeness and the effect of monetary rewards), and these hypotheses were tested across two models for each individual. There were three models total – the first one was with all combatants, and the other two models were for mutually exclusive subsets of combatants, abductees and volunteers. I'm not precisely certain how much tests this amounts to, whether changing the population means reducing the number of tests. I *think* we're looking at a total of six tests.

Then in a second trial, they tested interaction effects as well, in a model with five terms (money, closeness, recruitment, money * recruitment, closeness * recruitment). I'll focus on the trial they did, without interaction effects.

2. Calculate the expected number of false positives if all nulls were true.

They have an n of 78 for the first model, n of 29 for the second model, and n of 49 for the third model. For each, two hypotheses are tested.

```r
set.seed(123)
n_tests <- 2
n_obs_1 <- 78
n_obs_2 <- 29
n_obs_3 <- 49
alpha <- 0.05

# Create matrix of 20 independent tests (all null true)
p_values_1 <- matrix(NA, nrow = 1000, ncol = n_tests)
p_values_2 <- matrix(NA, nrow = 1000, ncol = n_tests)
p_values_3 <- matrix(NA, nrow = 1000, ncol = n_tests)

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs_1)
    AAS <- rnorm(n_obs_1)  # No relationship
    # Run regression and extract p-value for slope
    p_values_1[i, j] <- summary(lm(AAS ~ x))$coefficients[2, 4]
  }
}

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs_2)
    AAS <- rnorm(n_obs_2)  # No relationship
    # Run regression and extract p-value for slope
    p_values_2[i, j] <- summary(lm(AAS ~ x))$coefficients[2, 4]
  }
}

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs_3)
    AAS <- rnorm(n_obs_3)  # No relationship
    # Run regression and extract p-value for slope
    p_values_3[i, j] <- summary(lm(AAS ~ x))$coefficients[2, 4]
  }
}
```

```
# Analyze results
false_positives_1 <- rowSums(p_values_1 < alpha)
false_positives_2 <- rowSums(p_values_2 < alpha)
false_positives_3 <- rowSums(p_values_3 < alpha)
prop_at_least_one_1 <- mean(false_positives_1 > 0)
prop_at_least_one_2 <- mean(false_positives_2 > 0)
prop_at_least_one_3 <- mean(false_positives_3 > 0)
```

```
prop_at_least_one_1
```

```
[1] 0.101
```

```
prop_at_least_one_2
```

```
[1] 0.094
```

```
prop_at_least_one_3
```

```
[1] 0.09
```

In the study, the main findings were significant at a higher level than 0.05 (0.01). Given that their reported findings were secure at this level or higher, their conclusions seem much less likely to be results of random chance. A second test applied with $= 0.01$ indicates that their findings are even safer.

```
set.seed(123)
n_tests <- 2
n_obs_1 <- 78
n_obs_2 <- 29
n_obs_3 <- 49
alpha <- 0.01
```

```r
# Create matrix of 20 independent tests (all null true)
p_values_1a <- matrix(NA, nrow = 1000, ncol = n_tests)
p_values_2a <- matrix(NA, nrow = 1000, ncol = n_tests)
p_values_3a <- matrix(NA, nrow = 1000, ncol = n_tests)

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs_1)
    AAS <- rnorm(n_obs_1)  # No relationship
    # Run regression and extract p-value for slope
    p_values_1a[i, j] <- summary(lm(AAS ~ x))$coefficients[2, 4]
  }
}

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs_2)
    AAS <- rnorm(n_obs_2)  # No relationship
    # Run regression and extract p-value for slope
    p_values_2a[i, j] <- summary(lm(AAS ~ x))$coefficients[2, 4]
  }
}

for (i in 1:1000) {
  for (j in 1:n_tests) {
    # Generate independent data
    x <- rnorm(n_obs_3)
    AAS <- rnorm(n_obs_3)  # No relationship
    # Run regression and extract p-value for slope
    p_values_3a[i, j] <- summary(lm(AAS ~ x))$coefficients[2, 4]
  }
}

# Analyze results
false_positives_1a <- rowSums(p_values_1a < alpha)
false_positives_2a <- rowSums(p_values_2a < alpha)
false_positives_3a <- rowSums(p_values_3a < alpha)
prop_at_least_one_1a <- mean(false_positives_1a > 0)
prop_at_least_one_2a <- mean(false_positives_2a > 0)
prop_at_least_one_3a <- mean(false_positives_3a > 0)
```

```r
prop_at_least_one_1a
```

```
[1] 0.017
```

```r
prop_at_least_one_2a
```

```
[1] 0.011
```

```r
prop_at_least_one_3a
```

```
[1] 0.02
```

The risks that one of their findings is spirious at this significance level are very low, all 2% or less. And of the significant findings, only one was at 0.01; the others were actually at 0.001.

3. Would multiple testing corrections change their conclusions?

They do not report their exact p-values; they just tell us whether they get a p-value greater or less than the threshold. But given that the chance of their results being a result of random chance are very low. My skepticism about their findings remains, but more on the grounds of weak theorization than on the basis of the multiple comparisons problem. Because of the small number of tests, a Bonferroni correction would reduce the required p-value only by a factor of three, and most of the findings were significant at a p-value significantly lower than (0.05/3).

4.2 3b. You're designing a study to test 15 different hypotheses about voter behavior.

1. How would you adjust your analysis plan to account for multiple comparisons?

First of all, if I can at all combine hypotheses into an index, I would do that, as Coppock suggests in his article on multiple comparisons.

Second, if these are really all completely different hypotheses, I would add a correction (probably Benjamini-Hochberg, see below).

Third, every additional modification of model parameters and re-run of the regression means an increase in the number of tests by fifteen, which is sort of a lot. Especially since the increase in the effect of multiple comparisons is exponential. I'll specify in advance what modifications I will test and what expectations I have for how the modifications to parameters will affect the outcomes to reassure readers that I'm not just fishing.

Fourth, to get a significant result, because of the multiple comparisons problem, I would effectively need to meet much more demanding significance tests. This raises issues of statistical power. I would want to, if possible, increase the expected effect size (for example, making cues more powerful, assuming this is an experimental design). I would also want to

2. Would you use FWER or FDR control? Justify your choice.

If I have fifteen hypotheses tested all at once, it seems like I'm at an exploratory stage. I don't have much more information in this scenario, so I think I'd lean towards FDR controls to make sure I catch any real relationships worth following up on. Bonferroni would be far too conservative in this context (since I'd essentially be multiplying the p-values by *fifteen*!), and I would also expect some of the hypotheses to be correlated with one another.

3. How would sample size affect your decision?

If I had an absurdly large sample size, I would worry a bit less; the chance of a wide departure from the true value would be lower. Ideally, I would give myself as much statistical power as possible by increasing the sample size, since having the power to test a very small p-value (as this survey would have).

4. What would you report in the methods section about multiple testing?

I would explain that I'm only doing an exploratory survey, that as a result I corrected to keep my FDR low but was less concerned about my FWER, and that my pre-analysis plan is registered at (some link).