ProbSet 7, February 27

Gideon Gordon gideongordon2029@u.northwestern.edu

Due: February 27, 2026

Submission: https://canvas.northwestern.edu/courses/245562/assignments/1687752

```r
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```r
library(ggplot2)
```

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

---

1 Problem 1

1. Define statistical power in your own words. Statistical power is the likelihood that, given a certain experimental setup, a certain acceptable alpha-level, and a certain actual effect size, the researcher will *correctly* reject the null hypothesis. That is, the experimental setup can correctly determine that the effect is statistically different from zero.

2. Explain the relationship between Type I error (), Type II error (), and power. A Type I error is where we incorrectly believe that a relationship is real when it is not (we incorrectly reject the null hypothesis). A Type II error is where we incorrectly believe that a relationship is not real when it totally is (we incorrectly fail to reject the null hypothesis).

---

2 Problem 2

2.1 2a.

Simulate power for different scenarios:

```
# Complete this code to simulate power for different sample sizes
  # 1. Simulate n_sim datasets with given parameters
  # 2. For each dataset, run linear regression
  # 3. Calculate proportion of simulations where p < alpha
  # 4. Return power estimate
set.seed(144)


simulate_power <- function(true_effect, sample_size, sigma = 1, alpha = 0.05, n_sim = 10
```

```r
  significant_count = 0

  for (i in 1:n_sim) {
  x = c(rnorm(n = sample_size))
  y = c(true_effect*x + rnorm(n = sample_size, sd = sigma))
  simulated_data = data.frame(x = x, y = y)
  simulated_model = lm(y ~ x, data = simulated_data)
  p_value_i = summary(simulated_model)$coefficients[2, 4]
  if (p_value_i < alpha) {significant_count = significant_count + 1}
  }
  significant_count / n_sim
}

simulate_power(0.2, 100, sigma = 1, alpha = 0.05, n_sim = 1000)
```

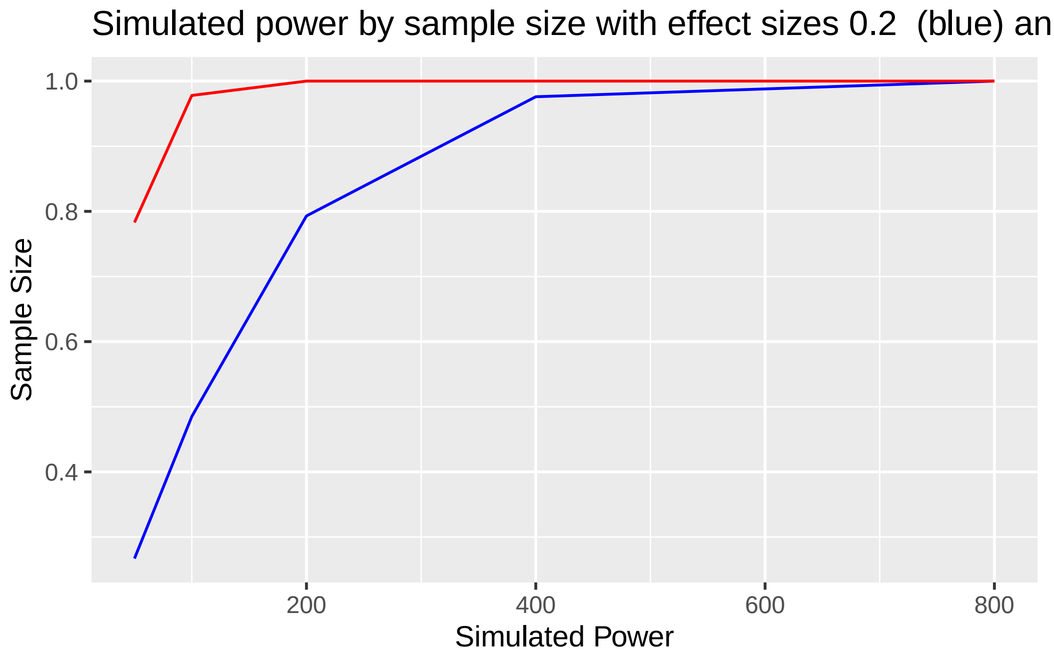
```
[1] 0.51
```

```r
# Test for different sample sizes
sample_sizes <- c(50, 100, 200, 400, 800)
true_effects <- c(0.2, 0.4)
sigmas = c(1, 2)
n_sim = 1000
```

```r
# Create a data frame with power estimates for each sample size
simulated_power_sample_size = data.frame(sample_sizes)

count = 0
powers_0.2 = c()
for (i in sample_sizes){
```

```
  count = count + 1
  power = simulate_power(true_effect = 0.2, sample_size = sample_sizes[count], sigma = 1
  powers_0.2 = c(powers_0.2, power)
}
simulated_power_sample_size_0.2 = data.frame(sample_sizes, powers_0.2)

count = 0
powers_0.4 = c()
for (i in sample_sizes){
  count = count + 1
  power = simulate_power(true_effect = 0.4, sample_size = sample_sizes[count], sigma = 1
  powers_0.4 = c(powers_0.4, power)
}
simulated_power_sample_size_0.4 = data.frame(sample_sizes, powers_0.4)
```

```
# Create visualization
library(ggplot2)
# Plot power vs sample size
graph_simulated_power_sample_size = ggplot() +
  geom_line(data = simulated_power_sample_size_0.2, aes(x = sample_sizes, y = powers_0.2
  geom_line(data = simulated_power_sample_size_0.4, aes(x = sample_sizes, y = powers_0.4
  labs(x = "Simulated Power", y = "Sample Size", title = "Simulated power by sample size

graph_simulated_power_sample_size
```



(I do not go for creativity in labeling this graph).

Questions: 1. What sample size is needed to achieve 80% power for detecting an effect of 0.2?

You need at least 400 observations.

2. How does changing the true effect size to 0.4 affect the required sample size?

The required sample size goes way down, from 400 to 100; doubling the effect reduces the required sample size by a factor of four.

3. What happens to power if you double the variance (sigma)?

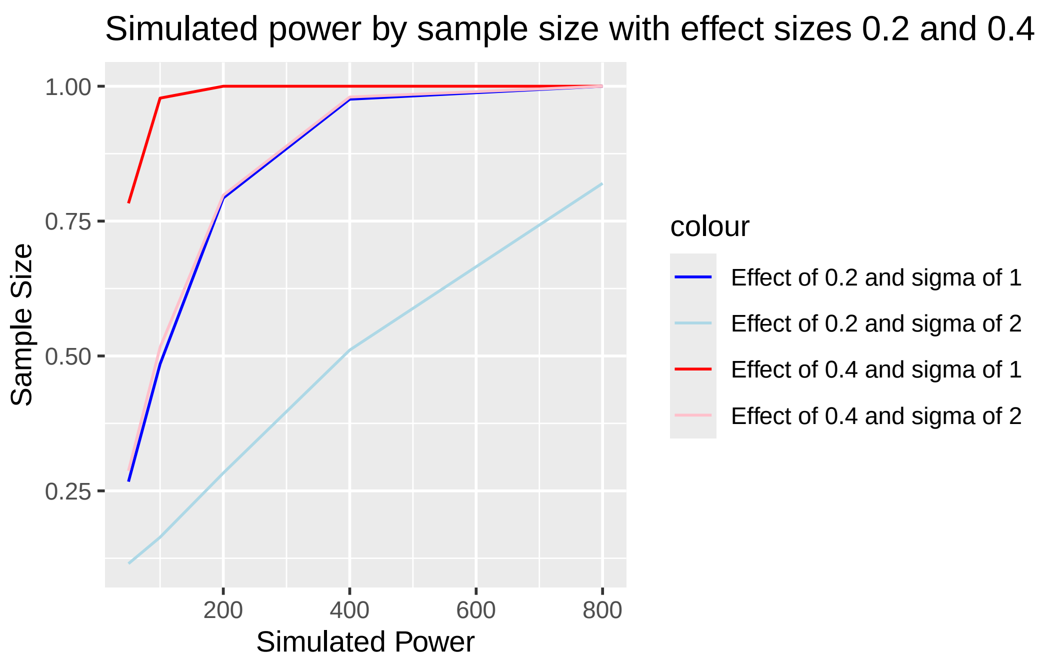
Let's find out!

```
count = 0

powers_0.2_sigma2 = c()

for (i in sample_sizes){

  count = count + 1

  power = simulate_power(true_effect = 0.2, sample_size = sample_sizes[count], sigma = 2
```

```r
  powers_0.2_sigma2 = c(powers_0.2_sigma2, power)
}
simulated_power_sample_size_0.2_sigma2 = data.frame(sample_sizes, powers_0.2_sigma2)

count = 0
powers_0.4_sigma2 = c()
for (i in sample_sizes){
  count = count + 1
  power = simulate_power(true_effect = 0.4, sample_size = sample_sizes[count], sigma = 2
  powers_0.4_sigma2 = c(powers_0.4_sigma2, power)
}
simulated_power_sample_size_0.4_sigma2 = data.frame(sample_sizes, powers_0.4_sigma2)
```

```r
# Plot power vs sample size
colors = c("powers_0.2" = "blue", "powers_0.2_sigma2" = "lightblue", "powers_0.4" = "red

graph_simulated_power_sample_size_and_variance = ggplot() +
  scale_color_manual(values = colors, labels = c("Effect of 0.2 and sigma of 1", "Effect
  geom_line(data = simulated_power_sample_size_0.2, aes(x = sample_sizes, y = powers_0.2
  geom_line(data = simulated_power_sample_size_0.2_sigma2, aes(x = sample_sizes, y = pow
  geom_line(data = simulated_power_sample_size_0.4, aes(x = sample_sizes, y = powers_0.4
    geom_line(data = simulated_power_sample_size_0.4_sigma2, aes(x = sample_sizes, y = p
  labs(
    x = "Simulated Power",
    y = "Sample Size",
    title = "Simulated power by sample size with effect sizes 0.2 and 0.4"
    )
```

```
graph_simulated_power_sample_size_and_variance
```



So what we can see from this graph is that doubling the sigma basically cancels out the effect of doubling the effect. A sample size four times larger is required when variance is doubled.

2.2 2b.

Questions: 1. What is the "winner's curse" and why does it occur?

A problem that comes with insufficient power is not just that we will make Type II errors and fail to distinguish a real effect from zero. We also find that when our confidence intervals are very wide (because of a small sample size or large variance) and our predicted effect fairly small, any "successes" we get even in the right direction will always be massive overestimates. With large confidence intervals, small true effects are indistinguishable from zero.

2. How does sample size affect the magnitude of the winner's curse?

If you can increase your sample size, you reduce the magnitude of the problem, since you become better able to identify smaller real effects as your confidence intervals shrink.