Problem set 5

Your name here

Due 11/1/2022 at 5pm

NOTE1: Start with the file `ps5_2022.Rmd` (available from the github repository at https://github.com/UChicago-pol-methods/IntroQSS-F22/tree/main/assignments). Modify that file to include your answers. Make sure you can "knit" the file (e.g. in RStudio by clicking on the `Knit` button). Submit both the Rmd file and the knitted PDF via Canvas

Question 1

(1a) **Write code to draw one sample of size 5 from a normal distribution with mean 3 and variance of 2.**

```
# your code here
# you can make the rest of the code chunks yourself
```

(1b) **Draw $m = 2000$ such samples, compute the sample mean for each sample, and plot a histogram of the result.**

(1c) **Compute the variance across your sample means. Compare it to the theoretical sampling variance of the sample mean. Explain any difference you find.**

(1d) **Repeat (1b) and (1c) but set the sample size to 500.**

(1e) **Show the histogram of Democratic vote share by county in the US in the 2020 presidential election. (You used this data in problem set 2. You might use the `pull()` function to extract a single column.) Does the distribution look symmetric?**

(1f) **Obtain 2000 random samples of 5 counties (without replacement), compute the mean of Democratic vote share for each sample, and plot the distribution of sample means. Does the distribution look symmetric?**

(1g) **Compute the variance across your sample means. Compare it to the theoretical variance of the sample mean. Explain any difference you find.**

(1h) **Repeat (1f) and (1g) but set the sample size to 500.**

Question 2

(2a) **Using the plug-in sample variance (Definition 3.2.18 in Aronow & Miller) as a guide, write the formula for plug-in sample covariance.**

Your math = here

(2b) **Pretending for a moment that U.S. county election results in 2020 are a sample from a super-population, use the formula you wrote above to estimate the covariance between the total votes cast in a county in 2020 and the proportion of votes cast for the Democrat in 2020. Compare it to the unbiased sample covariance, which you can compute using `R`'s `cov()` function.**