Nathan Tomlin

MATH 3070

December 1st, 2024

R-Project 2 Report

1. Set Up:

For the setup portion of this project, we must create a population of 500,000,000 people where 52% are in support of a presidential candidate and 48% are not in support of the candidate. We can accomplish this by using the following R code:

```
PopulationSize <- 500000000
Population <- c(rep("support", 0.52 * PopulationSize), rep("not", 0.48 * Popu
lationSize))
```

2. Sampling Distribution

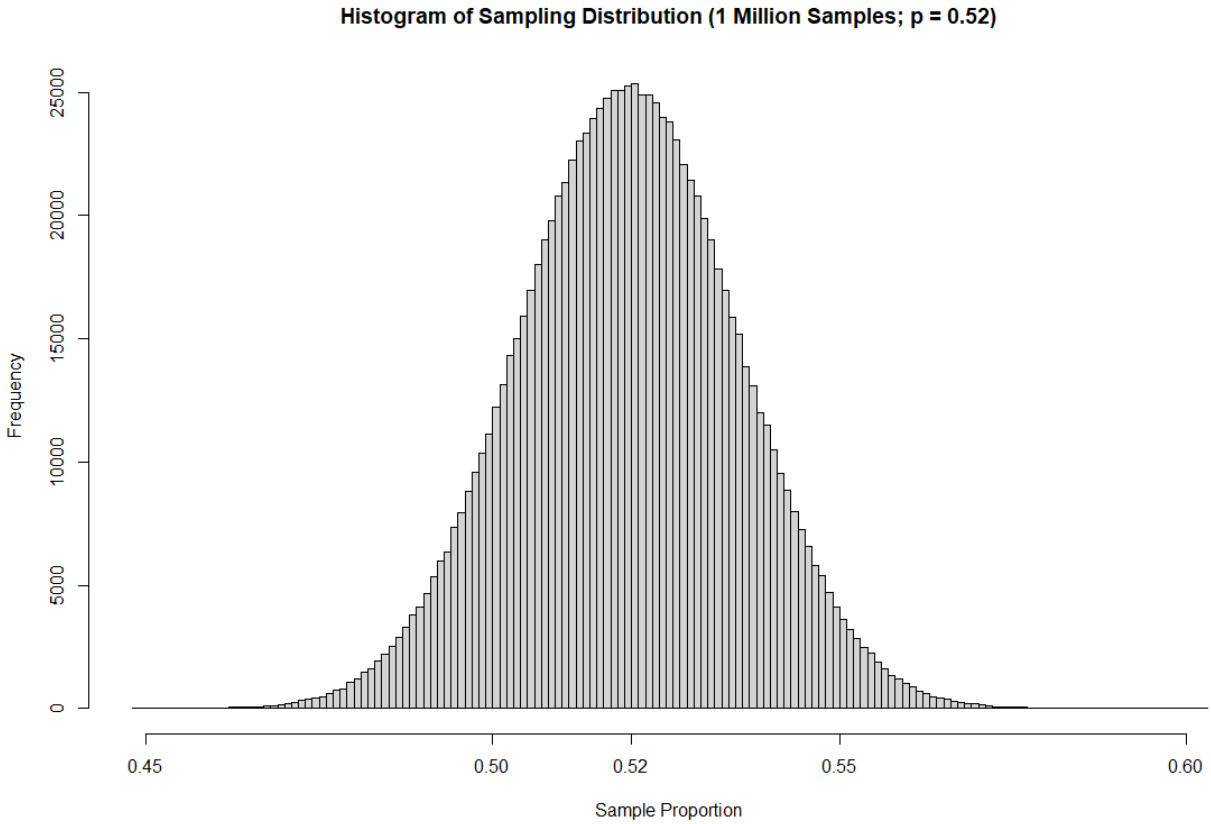
To obtain the sampling distribution, we must define a sample size of 1000, and then randomly sample this set sample size 1,000,000 times, making sure to calculate the sample proportion for every random sample. We can do this by using the following R code:

```
MillionSamplesOfSize1000_SampleProportions <- rep(NA, 1000000)
for(i in 1:1000000){
   samp <- sample(Population, size = 1000)
   MillionSamplesOfSize1000_SampleProportions[i] <- sum(samp == "support") /
1000
}
```

Below is the histogram of my sampling distribution, as well as the mean and standard deviation. These can be calculated by using the following R code:

```
hist(MillionSamplesOfSize1000_SampleProportions, breaks = 200,
      main = "Histogram of Sampling Distribution (1 Million Samples; p = 0.52
)",
      xlab = "Sample Proportion")
 axis(side = 1, at = c(0.52)) # Add a tick-mark at the true proportion
 mean(MillionSamplesOfSize1000_SampleProportions)
 sd(MillionSamplesOfSize1000_SampleProportions)
```

Note: The mean and sd will be printed to the console when the code is ran. They will not have labels, but will appear underneath their respective variables



Mean: 0.5200226

Standard Deviation: 0.01581167

3. Confidence Interval:

In order to create a 92% confidence interval for each of the 1,000,000 random samples, we can use the following R code:

```
ConfidenceIntervalLowerBounds <- rep(NA, 1000000)
ConfidenceIntervalUpperBounds <- rep(NA, 1000000)
z_score <- qnorm(0.96) # For 92% confidence level

for(i in 1:1000000){
   p_hat <- MillionSamplesOfSize1000_SampleProportions[i]
   ConfidenceIntervalLowerBounds[i] <- p_hat - z_score * sqrt(p_hat * (1 - p_
hat) / 1000)
   ConfidenceIntervalUpperBounds[i] <- p_hat + z_score * sqrt(p_hat * (1 - p_
hat) / 1000)
}

# Count confidence intervals that contain the true proportion
CountContainTrueProportion <- sum((0.52 >= ConfidenceIntervalLowerBounds) & (
0.52 <= ConfidenceIntervalUpperBounds))
```

```
PercentageContainTrueProportion <- CountContainTrueProportion / 1000000
PercentageContainTrueProportion
```

The number of intervals that contain the true proportion: 917,971

The percentage of intervals that contain the true proportion: 0.917971

➔ (91.7971% = about 92%)

Interpretation: From the 1,000,000 random samples we drew, we applied a 92% confidence interval to each of them. This means that, if correct, 92% of the intervals should contain the true population proportion. After applying the intervals to each random sample, we can see that the percentage of intervals that contained the true population proportion is around 91.7971, which is consistent with our expectations.

4. Significance Level:

For this section, we must conduct a hypothesis test on each of the 1,000,000 random samples. Each test must have a significance level (alpha) of 0.03, and a null hypothesis of p = 0.52. To do this, we can use the following R code:

```
# Set significance level and null hypothesis proportion
alpha <- 0.03
p_null <- 0.52
SE <- sqrt(p_null * (1 - p_null) / 1000)

# Initialize vectors for storing results
phatVector <- rep(NA, 1000000)   # Sample proportions
ZVector <- rep(NA, 1000000)      # Z-scores
pvalueVector <- rep(NA, 1000000) # p-values

# Perform hypothesis tests for 1,000,000 samples
for(i in 1:1000000){
  phatVector[i] <- MillionSamplesOfSize1000_SampleProportions[i]
  ZVector[i] <- (phatVector[i] - p_null) / SE
  pvalueVector[i] <- 2 * (1 - pnorm(abs(ZVector[i])))
}

# Calculate the number of rejected null hypotheses
Rejections <- sum(pvalueVector < alpha)

# Calculate the rejection rate
```

```
RejectionRate <- Rejections / 1000000
RejectionRate
```

From this code, we can also see:

- The number of hypotheses tests where the null value was rejected = 28,982
- The percentage hypothesis tests where the null value was rejected = 0.028982 = about 0.03 = 3%.

Interpretation: When the significance level (alpha) is equal to 0.03, we expect to incorrectly reject the null hypothesis in 3% of the tests. When we ran the hypothesis test on each of these 1,000,000 random samples, we found that we incorrectly rejected the null hypothesis in 2.89% of the tests, which is consistent with our expectations.

5. Hypothesis Test

For the final section, we must conduct a hypothesis test on each of the 1,000,000 random samples, in which the significance level (alpha) is 0.05 and the null hypothesis = 0.5 (not having a majority of support or not support in the population). We can do so by using the following R code:

```
Set significance level and null hypothesis proportion
alpha <- 0.05
p_null <- 0.5
SE <- sqrt(p_null * (1 - p_null) / 1000)

# Initialize vectors for storing results
phatVector <- rep(NA, 1000000)   # Sample proportions
ZVector <- rep(NA, 1000000)      # Z-scores
pvalueVector <- rep(NA, 1000000) # p-values

 Perform hypothesis tests for 1,000,000 samples
for(i in 1:1000000){
  phatVector[i] <- MillionSamplesOfSize1000_SampleProportions[i]
  ZVector[i] <- (phatVector[i] - p_null) / SE
  pvalueVector[i] <- 2 * (1 - pnorm(abs(ZVector[i])))
}

mean(phatvector)
sd(phatvector)
```

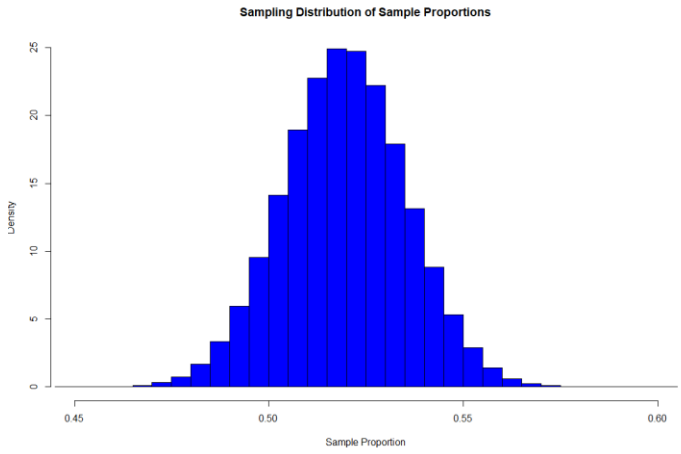
```
# Calculate the number of non-rejected null hypotheses
NonRejections <- sum(pvalueVector >= alpha)

# Calculate the non-rejection rate
NonRejectionRate <- NonRejections / 1000000
NonRejectionRate
```

We also need to get the sampling distribution as well as the null distribution, which we can do by using the following R code:

For the sampling distribution:

```
hist(phatVector, breaks = 50, freq = FALSE,
     main = "Sampling Distribution of Sample Proportions",
     xlab = "Sample Proportion",
     col = "blue",
     xlim = c(0.45, 0.6))
```

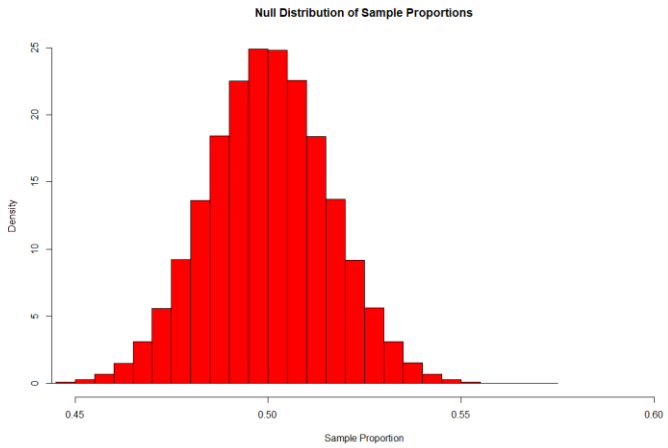


Sampling Mean: 0.5199864

Sampling SD: 0.01580013

For the null distribution:

```
hist(NullDistribution, breaks = 50, freq = FALSE,
     main = "Null Distribution of Sample Proportions",
     xlab = "Sample Proportion",
     col = "red",
     xlim = c(0.45, 0.6))
```



Null Mean: 0.50
Null SD: 0.0158

There is a difference between the two distributions. As you can see above, the standard deviation of both the null distribution and the sampling distribution are the same, however the means are different. The mean for the sampling distribution is 0.5199, while the mean for the null distribution is 0.50. This is because the sample distribution has a true proportion of 0.52, while the null hypothesis distribution has a value of 0.5.

Number of non-rejections: 746713

Percentage of non-rejections: 0.746713

For this specific section of the project, I did account for the number of type II errors made. This is because a type II error occurs when we fail to reject a null hypothesis which happens to be false, and we are recording the number of times that happens.