POL501 - Problem Set 3

Solutions

2024-12-04

Contents

Grading Criteria

- There will be **four problem sets** throughout the semester, which together account for **25% of the final course grade**.
- The total possible score for these problem sets is **25 out of 100 points** (with 100 points being the maximum course score).
- This problem set has a maximum score of **6.5 points**.
- **Scoring Breakdown**:
    - Each sub-letter in Question 1 is worth **0.5 point** (Total of 2.5 points)
    - Each sub-letter in Question 2 is worth **0.5 points** (Total of 1.5 points)
    - Each sub-letter in Question 3 is worth **0.5 points** (Total of 2.5 points)
- **Grading Guidelines**:
    - Full credit will be awarded for fully correct answers that closely match the provided solutions included in this document.
    - Partial credit will be given for incomplete or partially incorrect answers/justifications.
    - **0 points** will be awarded for missing answers, answers with no justification, or entirely incorrect responses.
    - Submissions without the RMD file will have point deductions.

Question 1 - Solutions

Preliminary Steps

Important:

- You must place this RMD file in the same folder in which you have the data.
- Then, you will be able to load the dataset into memory. The name of the dataset as an R object is `df_clean`.

As a preliminary recommended step, and to make sure we understand how the variables are coded and their distribution, let's start by describing the data using summary statistics.

```r
sstats_list <- c('vars', 'n', 'mean', 'sd', 'median', 'min', 'max')
kable(describe(df_clean)[sstats_list], format='latex')
```

	vars	n	mean	sd	median	min	max
RESPID	1	5199	9457.661281	5483.2245412	9464	2	18838
PARTY	2	5199	2.177150	0.9685161	2	1	4
INTFREQ	3	5199	1.810156	0.8919744	2	1	5
RADIO	4	5199	1.219465	0.4139242	1	1	2
ECON1MOD	5	5199	2.730140	0.8352547	3	1	4
INFRASPEND	6	5199	2.151568	0.9420048	2	1	5
MOREGUNIMPACT	7	5199	1.878631	0.8552866	2	1	3
CRIMESAFE	8	5199	2.575303	0.8459974	3	1	5

Then, we define a user-defined function to compute confidence intervals. The benefit of defining this function, is that we can then reuse it multiple times just by providing new inputs.

```r
# Function to calculate confidence interval
confidence_interval <- function(data, var_name, confidence_level = 0.95, method =
↪  "t") {

  # Extract the data column
  data_column <- data[[var_name]]

  # Validate inputs (method, var_name)
  if (!(method %in% c("t", "z"))) {
    stop("Method must be either 't' or 'z'.")
  }
  if (!(var_name %in% colnames(data))) {
    stop("var_name must be the name of a column in the data frame.")
  }

  # Calculate sample statistics
  sample_mean <- mean(data_column, na.rm = TRUE)
  sample_sd <- sd(data_column, na.rm = TRUE)
  n <- length(na.omit(data_column))
  standard_error <- sample_sd / sqrt(n)

  # Set alpha for confidence level
  alpha <- 1 - confidence_level

  # Calculate the critical value based on the selected method
  if (method == "t") {
    critical_value <- qt(1 - alpha / 2, df = n - 1)  # t-distribution critical
↪   value
  } else {
    critical_value <- qnorm(1 - alpha / 2)  # z-distribution critical value
  }

  # Calculate margin of error (MOE)
  margin_of_error <- critical_value * standard_error

  # Calculate confidence interval bounds
  lower_bound <- sample_mean - margin_of_error
  upper_bound <- sample_mean + margin_of_error

  # Prepare output as a named vector
  output <- c(
    `Sample Mean of` = var_name,
    Estimate = round(sample_mean, 3),
    MOE = round(margin_of_error, 3),
    `Lower CI Bound` = round(lower_bound, 3),
    `Upper CI Bound` = round(upper_bound, 3)
  )

  # Return the result
  return(output)
}
```

Answer to (1.a)

First, recall the coding of `CRIMESAFE` responses. The question asked: ***How would you describe the area where you live, in terms of crime?***$

- 1 = Extremely safe;
- 2 = Very safe;
- 3 = Somewhat safe;
- 4 = Not too safe;
- 5 = Not at all safe

```
# Question 1.a: Use the function `confidence_interval` to compute the mean of
↪   `CRIMESAFE` with a 95% confidence level. Interpret the confidence interval
↪   appropriately.
crimesafe_CI_z_95 <- confidence_interval(data=df_clean,
```

Table 1: 95% CI for the mean of ‘CRIMESAFE‘

	x
Sample Mean of	CRIMESAFE
Estimate	2.575
MOE	0.023
Lower CI Bound	2.552
Upper CI Bound	2.598

```
                                         var_name='CRIMESAFE',
                                         confidence_level = 0.95, method = "t")
print(crimesafe_CI_z_95)
```

```
## Sample Mean of       Estimate            MOE Lower CI Bound Upper CI Bound
##    "CRIMESAFE"        "2.575"        "0.023"        "2.552"        "2.598"
```

```
kable(crimesafe_CI_z_95, format='latex', caption = "95\\% CI for the mean of
↪  `CRIMESAFE`")
```

Explanation/Justification: The average score for CRIMESAFE is 2.575, with a 95% confidence interval ranging from 2.552 to 2.598. This means we can be 95% confident that the true average perception of safety in the area falls within this range. Since the scale ranges from 1 (Extremely Safe) to 5 (Not at All Safe), a score of 2.575 indicates that most people feel their area is somewhere between “Very Safe” and “Somewhat Safe.” The narrow margin of error (0.023) suggests the estimate is precise, giving us strong confidence in this result.

Answer to (1.b)

```
# Question 1.b: Use the function `confidence_interval` to compute the mean of
↪  `CRIMESAFE` with a 99% confidence level
crimesafe_CI_z_99 <- confidence_interval(data=df_clean,
                                         var_name='CRIMESAFE',
                                         confidence_level = 0.99, method = "t")
print(crimesafe_CI_z_99)
```

```
## Sample Mean of       Estimate            MOE Lower CI Bound Upper CI Bound
##    "CRIMESAFE"        "2.575"         "0.03"        "2.545"        "2.606"
```

```
kable(crimesafe_CI_z_99, format='latex',caption = "99\\% CI for the mean of
↪  `CRIMESAFE`")
```

Explanation/Justification: The average score for CRIMESAFE is 2.575, with a 99% confidence interval ranging from 2.545 to 2.606. This means we can be 99% confident that the true average perception of safety in the area lies within this range. Similar to the 95% confidence interval, the score of 2.575 reflects that most people perceive their neighborhood as between “Very Safe” and “Somewhat Safe.” However, the wider margin of error (0.03) in the 99% confidence interval compared to the 95% interval shows that increasing the confidence level requires a broader range to ensure

Table 2: 99% CI for the mean of ‘CRIMESAFE‘

	x
Sample Mean of	CRIMESAFE
Estimate	2.575
MOE	0.03
Lower CI Bound	2.545
Upper CI Bound	2.606

higher certainty. This still indicates a generally positive perception of safety in the community with a high degree of confidence.

Answer to (1.c)

```
# Question 1.c: Use the `dplyr` command 'filter' to create a data frame for the
↪  responses of Democrats and Republicans
# Create one data frame for Democrats and another for Republicans

# Let's inspect the values of the variable `PARTY`
table(df_clean$PARTY)
```

```
##
##    1    2    3    4
## 1514 1795 1345  545
```

```
# Filter rows and create subsamples
df_dems <- df_clean %>%
  filter(PARTY==2)

df_rep <- df_clean %>%
  filter(PARTY==1)
```

Explanation/Justification:

First, we recall that the coding for the responses for the `PARTY` variable are:

`PARTY`. In politics today, do you consider yourself a…

- 1 = Republican;
- 2 = Democrat;
- 3 = Independent;
- 4 = Something else.

Thus, using `filter()` we select the rows matching Republicans and Democrats. The data frame `df_dems` includes rows where $PARTY = 2$ which matches respondents who selected $2 = Democrat$, and the data frame `df_reps` includes rows where $PARTY = 1$, which corresponds to respondents who selected $1 = Republican$.

Table 3: 95% CI for the mean of ‘CRIMESAFE‘ among Democrats

	x
Sample Mean of	CRIMESAFE
Estimate	2.526
MOE	0.04
Lower CI Bound	2.486
Upper CI Bound	2.566

Answer to (1.d)

```
# Question 1.d: Compute the sample mean for `CRIMESAFE` and 95% confidence
↪  intervals for Democrats and Republicans
crimesafe_CI_z_95_dems <- confidence_interval(data=df_dems,
                                              var_name='CRIMESAFE',
                                              confidence_level = 0.95, method = "t")
cat('\nResults for Democrats:\n')
```

```
##
## Results for Democrats:
```

```
print(crimesafe_CI_z_95_dems)
```

```
## Sample Mean of       Estimate            MOE Lower CI Bound Upper CI Bound
##    "CRIMESAFE"        "2.526"         "0.04"        "2.486"        "2.566"
```

```
kable(crimesafe_CI_z_95_dems, format='latex', caption = "95\\% CI for the mean of
↪  `CRIMESAFE` among Democrats")
```

```
crimesafe_CI_z_95_reps <- confidence_interval(data=df_rep,
                                              var_name='CRIMESAFE',
                                              confidence_level = 0.95, method = "t")
cat('\nResults for Republicans:\n')
```

```
##
## Results for Republicans:
```

```
print(crimesafe_CI_z_95_reps)
```

```
## Sample Mean of       Estimate            MOE Lower CI Bound Upper CI Bound
##    "CRIMESAFE"        "2.596"        "0.041"        "2.556"        "2.637"
```

```
kable(crimesafe_CI_z_95_reps, format='latex', caption = "95\\% CI for the mean of
↪  `CRIMESAFE` among Republicans")
```

Explanation/Justification: The results show that Democrats have an average CRIMESAFE score of 2.526, with a 95% confidence interval of [2.486, 2.566], indicating they perceive their neighborhoods as leaning towards “Very Safe” to “Somewhat Safe.” Republicans, on the other hand, have a slightly higher average score of 2.596, with a 95% confidence interval of [2.556, 2.637],

Table 4: 95% CI for the mean of ‘CRIMESAFE‘ among Republicans

	x
Sample Mean of	CRIMESAFE
Estimate	2.596
MOE	0.041
Lower CI Bound	2.556
Upper CI Bound	2.637

suggesting they also perceive their neighborhoods as “Very Safe” to “Somewhat Safe,” but slightly less safe compared to Democrats.

Answer to (1.e)

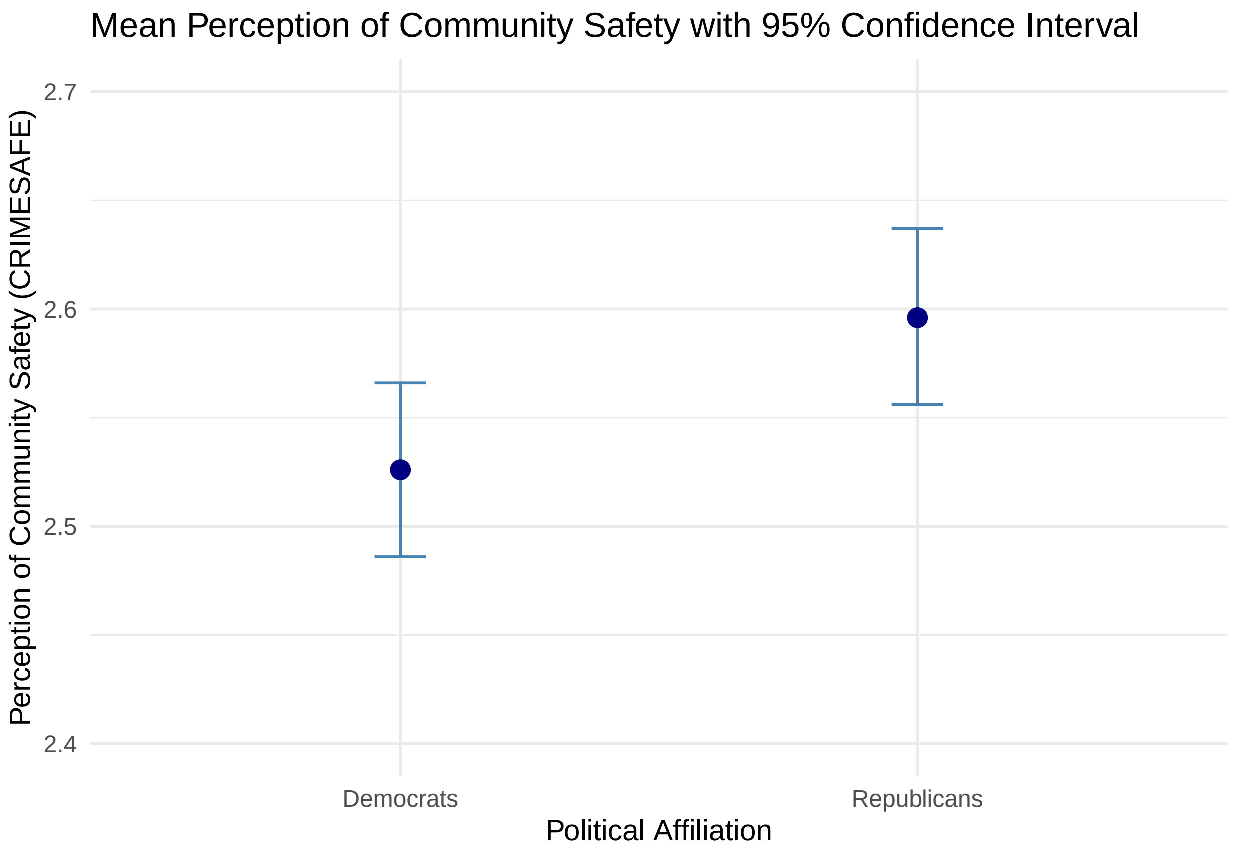
```
# Question 1.e: Create a data frame with the results of (1.d) and plot the
↪  results
# Extract the values and combine them into a data frame for plotting
combined_crimesafe_CI_95 <- cbind(c(Group = 'Democrats', crimesafe_CI_z_95_dems),
                                  c(Group = 'Republicans',
                                   ↪  crimesafe_CI_z_95_reps))

# Convert the combined matrix into a data frame with proper column names
df_combined_CI <- as.data.frame(t(combined_crimesafe_CI_95), stringsAsFactors =
↪  FALSE)

# Rename columns for clarity
colnames(df_combined_CI) <- c("Group", "Sample_Mean", "Estimate", "MOE", "Lower",
↪  "Upper")

# Convert relevant columns to numeric
df_combined_CI <- df_combined_CI %>%
  mutate(Estimate = as.numeric(Estimate),
         Lower = as.numeric(Lower),
         Upper = as.numeric(Upper))

# Plot using ggplot2 with a dot and whiskers for confidence interval
ggplot(df_combined_CI, aes(x = Group, y = Estimate)) +
  geom_errorbar(aes(ymin = Lower, ymax = Upper), width = 0.1, color =
  ↪  "steelblue") +  # Whiskers for CI
  geom_point(size = 3, color = "navyblue") +  # Dot for the sample mean
  labs(
    title = "Mean Perception of Community Safety with 95% Confidence Interval",
    x = "Political Affiliation",
    y = "Perception of Community Safety (CRIMESAFE)"
  ) +
  theme_minimal() +
ylim(2.4, 2.7)  # Set the y-axis limits here
```



Explanation/Justification: The plot shows that the 95% confidence intervals for Democrats ([2.486, 2.566]) and Republicans ([2.556, 2.637]) represent the range within which their true mean perceptions of neighborhood safety likely fall. While these intervals overlap slightly, this does not automatically imply that the difference in their true means is not statistically significant. Intuitively, a smaller overlap does indicate a lower likelihood that the true means of the two groups are the same. Yet, overlap of confidence intervals is not a definitive test of significance and can lead to misinterpretation. To properly assess whether the difference in perceptions is statistically meaningful, a confidence interval for the difference in means between the two groups should be calculated.

Question 2 - Solutions

This question follows the previous one and now focuses on inference for proportions.

Recall the coding of `CRIMESAFE` responses. The question asked: ***How would you describe the area where you live, in terms of crime?***$

- 1 = Extremely safe;
- 2 = Very safe;
- 3 = Somewhat safe;
- 4 = Not too safe;

- 5 = Not at all safe

Answer to (2.a)

```r
# Question 2.a: Create a new variable in the dataframe `df_clean` that equals 1
↪  if the respondent selected
# "Not too safe" (=4) or "Not at all safe" (=5), and zero otherwise. Then,
↪  recreate the subsample data frames for Democrats and Republicans.
# Finally, interpret the mean of this variable.

df_clean <- df_clean %>%
  mutate(
    notsafe_binary = if_else(CRIMESAFE %in% c(4, 5), 1, 0)
  )

# Re-Create subsamples for Democrats and Republicans (Now the subsamples include
↪  the new variable)
df_dems <- df_clean %>%
  filter(PARTY==2)

df_rep <- df_clean %>%
  filter(PARTY==1)

# Describe the mean of the new variable - entire sample
print("Proportion of Respondents Who Selected `Not too safe` OR `Not at all safe`
↪  (entire sample):")
```

```
## [1] "Proportion of Respondents Who Selected `Not too safe` OR `Not at all safe` (entire samp
```

```r
mean(df_clean$notsafe_binary)
```

```
## [1] 0.107136
```

```r
# Describe the mean of the new variable for Democrats
print("Proportion of Democrats Who Selected `Not too safe` OR `Not at all
↪  safe`:")
```

```
## [1] "Proportion of Democrats Who Selected `Not too safe` OR `Not at all safe`:"
```

```r
mean(df_dems$notsafe_binary)
```

```
## [1] 0.1097493
```

```r
# Describe the mean of the new variable for Republicans
print("Proportion of Republicans Who Selected `Not too safe` OR `Not at all
↪  safe`:")
```

```
## [1] "Proportion of Republicans Who Selected `Not too safe` OR `Not at all safe`:"
```

Table 5: Proportion of Respondents Who Selected ‘Not too safe‘ OR ‘Not at all safe‘

Group	Proportion
Entire Sample	0.107
Democrats	0.110
Republicans	0.098

```r
mean(df_rep$notsafe_binary)
```

```
## [1] 0.0984148
```

```r
# Create a summary dataframe
summary_df <- data.frame(
  Group = c("Entire Sample", "Democrats", "Republicans"),
  Proportion = c('mean_fullsample'=mean(df_clean$notsafe_binary),
                 'mean_dems'=mean(df_dems$notsafe_binary),
                 'mean_reps'=mean(df_rep$notsafe_binary))
)
rownames(summary_df) <- NULL # Remove row "ID names"
summary_df$Proportion <- round(summary_df$Proportion,3) # round values

# Use kable to display the dataframe
print(summary_df)
```

```
##           Group Proportion
## 1 Entire Sample      0.107
## 2     Democrats      0.110
## 3   Republicans      0.098
```

```r
kable(summary_df, caption = "Proportion of Respondents Who Selected `Not too
↪  safe` OR `Not at all safe`", format='latex')
```

Explanation/Justification: The sample mean of the `notsafe_binary` variable represents the proportion of respondents who perceive their neighborhoods as “Not too safe” or “Not at all safe.” For the entire sample, the proportion is **0.107**, indicating that approximately 10.7% of all respondents feel their neighborhoods are unsafe. Among Democrats, the proportion is slightly higher at **0.110**, meaning about 11.0% of Democrats perceive their neighborhoods as unsafe. For Republicans, the proportion is slightly lower at **0.098**, with only 9.8% expressing similar concerns. These results suggest that Democrats are slightly more likely than Republicans to feel unsafe in their neighborhoods, but overall, the proportions are relatively small in both groups, indicating that the majority of respondents from both political affiliations perceive their neighborhoods as relatively safe.

Answer to (2.b)

```
# Question 2.b: Using the subsample for Democrats, compute the p-value with the
↪  null hypothesis
# that the true proportion for Democrats differs from the sample proportion for
↪  Republicans.
# EXPLAIN AND INTERPRET THE RESULTS

prop_null_hypothesis <- mean(df_rep$notsafe_binary) # hypothesis = the sample
↪  prop for reps
n_sample_size <- nrow(df_dems)

# Calculate the standard DEVIATION for the null hypothesis (replace with the
↪  correct formula)
SD_null_hypot <- sqrt(prop_null_hypothesis*(1-prop_null_hypothesis))

# Run the z-test to test if the mean is significantly different from the sample
↪  proportion for Republicans
#   The function will compute in the background the correct Standard Error.
z_test_result <- z.test(
  x = df_dems$notsafe_binary, # Here we declare the variable for our sample,
  ↪  i.e., democrats
  mu = prop_null_hypothesis,  # Hypothesized population mean
  sigma.x = SD_null_hypot  # Population standard DEVIATION (given H0)
)

# Print the z-test result
print(z_test_result)
```

```
##
##  One-sample z-Test
##
## data:  df_dems$notsafe_binary
## z = 1.6121, p-value = 0.1069
## alternative hypothesis: true mean is not equal to 0.0984148
## 95 percent confidence interval:
##  0.0959693 0.1235293
## sample estimates:
## mean of x
## 0.1097493
```

Explanation/Justification: For the Democratic subsample, the z-test results in a z-value of **1.6121** and a p-value of **0.1069**. Since the p-value is greater than both the 5% (0.05) and 1% (0.01) significance levels, we fail to reject the null hypothesis. This means there is insufficient evidence to conclude that the true proportion of Democrats who feel unsafe is significantly different from the sample proportion observed for Republicans.

Answer to (2.c)

```r
# Question 2.c: Using the REPUBLICANS subsample, compute the p-value with the
↪  null hypothesis
# that the true proportion differs from the estimated sample proportion for
↪  Democrats.
# EXPLAIN AND INTERPRET THE RESULTS

prop_null_hypothesis <- mean(df_dems$notsafe_binary)
n_sample_size <- nrow(df_rep) # UPDATED CODE LINE

# Calculate the standard DEVIATION for the null hypothesis (replace with the
↪  correct formula)
SD_null_hypot <- sqrt(prop_null_hypothesis*(1-prop_null_hypothesis))

# Run the z-test to test if the mean is significantly different from the sample
↪  proportion for Democrats
z_test_result <- z.test(
  x = df_rep$notsafe_binary, # Here we declare the variable for our sample
  mu = prop_null_hypothesis,  # Hypothesized population mean
  sigma.x = SD_null_hypot  # Population standard DEVIATION (assuming known)
)

# Print the z-test result
print(z_test_result)
```

```
##
##  One-sample z-Test
##
## data:  df_rep$notsafe_binary
## z = -1.4109, p-value = 0.1583
## alternative hypothesis: true mean is not equal to 0.1097493
## 95 percent confidence interval:
##  0.08266981 0.11415978
## sample estimates:
## mean of x
## 0.0984148
```

Explanation/Justification: For the Republican subsample, the z-test yields a z-value of **-1.4109** and a p-value of **0.1583**. Since the p-value is greater than both the 5% (0.05) and 1% (0.01) significance levels, we fail to reject the null hypothesis. This indicates there is insufficient evidence to conclude that the true proportion of Republicans who feel unsafe is significantly different from the sample proportion observed for Democrats.

Question 3 - Solutions

This question introduces hypothesis testing using a mock dataset for proportions without relying on any specialized packages. In this question you will test the null hypothesis that the true population proportion is 0.43 against the alternative that it is different, using a 1% significance level.

Preliminary Steps

We will generate a new mock dataset for this question. We set a seed equal to 12345 to ensure reproducibility and create the mock dataset. Also, we will set up the simulated data using a bernoulli random variable generator with a true proportion of success equal to 0.44. We will use a sample size of 35 observations.

Don't modify this chunk:

```
# Set seed for reproducibility
set.seed(12345)

# Create a mock dataset with 35 respondents where each respondent either supports
↪  (1) or does not support (0) a specific policy
respondents <- 35
true_prop <- 0.44
mock_data_q3 <- data.frame(support_policy = rbinom(n=respondents, size=1,
↪  true_prop))  # size=1 means one trial, hence we are drawing 'n' bernoulli
↪  random variables.

# Visualize the first few rows of the dataset
head(mock_data_q3)
```

```
##   support_policy
## 1              1
## 2              1
## 3              1
## 4              1
## 5              0
## 6              0
```

Answer to (3.a)

First, calculate the sample proportion and the standard error.

```
# Calculate the sample proportion
p_hat <- mean(mock_data_q3$support_policy) # Observed Sample Proportion

# Define the null hypothesis proportion
p_0 <- 0.43

# Compute the sample size
n <- nrow(mock_data_q3)
```

```r
# Calculate the standard ERROR under the null (Appropiate SE assuming H0 is true)
standard_error_null_hypothesis <- sqrt( ( p_0*(1 - p_0) )/n )  # SE assuming the
↪  null hypothesis

# Print the results
cat("Sample Proportion: ", round(p_hat, 3), "\n")
```

```
## Sample Proportion:  0.4
```

```r
cat("Standard Error: ", round(standard_error_null_hypothesis, 3), "\n")
```

```
## Standard Error:  0.084
```

Explanation/Justification: The sample proportion ($\hat{p}$) is **0.4**, meaning that 40% of respondents in the sample support the policy. The standard error under the null hypothesis (SE_{H_0}) is **0.084**, which measures the expected variability of the sample proportion if the true proportion in the population is equal to the null hypothesis proportion ($p_0 = 0.43$).

Answer to (3.b)

- **Null Hypothesis (H_0):**
  The true population proportion of respondents who support the policy is equal to 0.43.
  Mathematically: $H_0 : p = 0.43$.
- **Alternative Hypothesis (H_a):**
  The true population proportion of respondents who support the policy is not equal to 0.43 (i.e., it could be higher or lower).
  Mathematically: $H_a : p \neq 0.43$.

This is a two-tailed test because we are interested in whether the true proportion differs from 0.43 in either direction, not just whether it is greater or less than 0.43. The significance level (α) is set at 1%, which corresponds to a 99% confidence level for the test.

Answer to (3.c)

```r
# Define the null hypothesis proportion
p_0 <- 0.43

# Define sample proportion (copy and paste code above)
p_hat <- mean(mock_data_q3$support_policy) # Observed Sample Proportion

# Define Standard Error Under the Null
SE_H0 <- sqrt( ( p_0*(1 - p_0) )/n )  # SE assuming the null hypothesis

# Calculate the z-test statistic
z_statistic <- (p_hat - p_0)/SE_H0  # Use: `p_0`, `p_hat`, `SE_H0`

# Print the test statistic
cat("Z-test Statistic: ", round(z_statistic, 3), "\n")
```

```
## Z-test Statistic:  -0.358
```

Explanation/Justification: The z-test statistic is **-0.358**, which measures how many standard errors the observed sample proportion ($\hat{p} = 0.4$) is away from the null hypothesis proportion ($p_0 = 0.43$). A negative z-value indicates that the observed proportion is below the null hypothesis proportion, but the magnitude of -0.358 is relatively small.

Answer to (3.d)

```r
# Critical value for 1% significance level (two-tailed)
alpha <- 0.01
critical_value <- qnorm( 1 - (alpha / 2) )  # You can learn more about the
↪  function `qnorm` by running in the console: ?qnorm

# Calculate the z-test statistic
z_statistic <- (p_hat - p_0)/SE_H0   # Replace with the code used above

# Calculate the p-value. HINT: Use `pnorm(abs(z_statistic), lower.tail = FALSE)`
↪  to compute Pr(Z > |Z_obs|).
p_value <- 2*(pnorm(abs(z_statistic), lower.tail = FALSE))

# Print the critical value and p-value
cat("Critical Value: ", round(critical_value, 3), "\n")
```

```
## Critical Value:  2.576
```

```r
cat("P-value: ", round(p_value, 3), "\n")
```

```
## P-value:  0.72
```

Explanation/Justification: The **critical value** at a 1% significance level ($\alpha = 0.01$) for a two-tailed test is **2.576**. This means that for the null hypothesis to be rejected, the z-test statistic would need to fall outside the range of $[-2.576, 2.576]$. However, the calculated z-test statistic is **-0.358**, which lies well within this range, indicating that the observed sample proportion is not significantly different from the hypothesized proportion.

The p-value of **0.72** means that, if the true population proportion were 0.43, there is a 72% probability of observing a result as extreme as (or more extreme than) the one in the sample due to random chance alone. Since the p-value is much larger than the 1% significance level ($p > 0.01$), we fail to reject the null hypothesis.

Answer to (3.e)

```r
# Print the z-statistic, critical value and p-value
cat("Z-test Statistic: ", round(z_statistic, 3), "\n")
```

```
## Z-test Statistic:  -0.358
```

```r
cat("Critical Value: ", round(critical_value, 3), "\n")
```

```
## Critical Value:  2.576
```

```
cat("P-value: ", round(p_value, 3), "\n")
```

```
## P-value:  0.72
```

Explanation/Justification:

The z-test statistic is **-0.358**, which is well within the range of $[-2.576, 2.576]$, defined by the critical values at the 1% significance level. Additionally, the p-value is **0.72**, which is far greater than the significance level of 0.01. Both results indicate that the observed sample proportion ($\hat{p} = 0.4$) is not significantly different from the hypothesized population proportion ($p_0 = 0.43$).

Conclusion

At the 1% significance level, we fail to reject the null hypothesis. This means there is no sufficient evidence to conclude that the true proportion of respondents who support the policy is different from 0.43. The observed difference is likely due to random sampling variability.

Additional Remarks:

While the sample proportion ($\hat{p} = 0.4$) differs slightly from the hypothesized proportion ($H_0 : p = 0.43$), and the true proportion used to generate the data is $p = 0.44$, we still fail to reject the null hypothesis. This occurs because the sample size ($n = 35$) is relatively small, and the null hypothesis value ($p_0 = 0.43$) lies very close to the true proportion ($p = 0.44$). With a small sample size, the standard error is larger, making it harder to detect small differences between the observed proportion and the null hypothesis. If the sample size were larger, the standard error would decrease, improving our ability to distinguish small differences between the population parameters.

(Optional) Verify your previous result running the following chunk:

```
prop_null_hypothesis <- 0.43
n_sample_size <- respondents

# Calculate the standard DEVIATION for the null hypothesis (replace with the
↪  correct formula)
SD_null_hypot <- sqrt( prop_null_hypothesis*(1-prop_null_hypothesis) )   # The
↪  formula for SD does not include the sample size

# Run the z-test to test if the mean is significantly different from the sample
↪  proportion for Democrats
z_test_result <- z.test(
  x = mock_data_q3$support_policy,
  mu = prop_null_hypothesis,  # Hypothesized population mean
  sigma.x = SD_null_hypot  # Population standard DEVIATION (assuming known)
)

# Print results
print(z_test_result)
```

```
##
##  One-sample z-Test
##
## data:  mock_data_q3$support_policy
## z = -0.3585, p-value = 0.72
## alternative hypothesis: true mean is not equal to 0.43
## 95 percent confidence interval:
##  0.2359842 0.5640158
## sample estimates:
## mean of x
##       0.4
```

```
# Print p-value
print(z_test_result$p.value)
```

```
## [1] 0.7199726
```