POL501 - Problem Set 2

Answers to Questions

2024-11-07

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 particular problem set has a maximum score of **8 points**.
- **Scoring Breakdown**:
  - Each sub-letter in Question 1 is worth **1 point** (Total of 5 points)
  - Each sub-letter in Question 2 is worth **0.5 points** (Total of 3 points)
- **Grading Guidelines**:
  - Full credit will be awarded for answers that closely match the provided solutions.
  - 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

Let's start by describing the data using summary statistics.

```
knitr::kable(describe(df_clean)[c('vars', 'n', 'mean', 'sd', 'median', 'min',
↪  'max')], 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

Answer - Question (1.a)

Based on the survey data, calculate the probability that a randomly selected respondent identifies as either a Democrat or a Republican.

```
# Step 1: Calculate the total number of valid respondents
n_total_q1 <- nrow(df_clean)

# Step 2: Filter for respondents who identify as either Democrat or Republican
n_rep_OR_dem <- df_clean %>%
  filter(PARTY == 1 | PARTY == 2) %>%  # Logical OR condition to select Democrats
  ↪  (coded as 2) or Republicans (coded as 1)
  nrow()

# Step 2 (Equivalent Alternative): Filter for respondents who identify as either
↪  Democrat or Republican
n_rep_OR_dem <- df_clean %>%
  filter(PARTY %in% c(1,2) ) %>%  # Logical OR condition to select Democrats
  ↪  (coded as 2) or Republicans (coded as 1)
  nrow()

# Step 3: Calculate the probability of identifying as Democrat or Republican
prob_rep_OR_dem <- n_rep_OR_dem / n_total_q1

# Step 4: Display the result
cat(paste("Probability of identifying as Democrat or Republican:",
            round(prob_rep_OR_dem, 2)))
```

```
## Probability of identifying as Democrat or Republican: 0.64
```

Explanation: The probability of identifying as Democrat or Republican is equal to 64 percent. This was computed by counting the respondents who classified as Democrat **OR** Republican. Since Democrat and Republican are mutually exclusive categories, it is valid to add their counts. In other

words, **Probability of Democrat or Republican = (#Democrats + #Republicans)/(Total Respondents)**. This ensures that we are not double-counting any respondents, as they can belong to only one of these categories.

Answer - Question (1.b)

What is the probability that a respondent listens to the radio or uses the internet almost constantly?

```
# Step 1: Calculate the total number of respondents
n_total_q1b <- nrow(df_clean)  # This is the total number of respondents in the
↪  dataset

# Step 2: Count respondents who listen to the radio
n_radio_yes <- df_clean %>%
  filter(RADIO == 1) %>%  # 'RADIO == 1' filters for respondents who answered
  ↪  'Yes' to listening to the radio
  nrow()  # Counts the number of rows that meet the condition (i.e., number of
  ↪  people who listen to the radio)

# Step 3: Count respondents who use the internet almost constantly
n_inter_constantly <- df_clean %>%
  filter(INTFREQ == 1) %>%  # 'INTFREQ == 1' filters for respondents who use the
  ↪  internet 'almost constantly'
  nrow()  # Counts the number of rows that meet the condition (i.e., number of
  ↪  people using internet almost constantly)

# Step 4: Count respondents who both listen to the radio and use the internet
↪  almost constantly
n_radio_yes_AND_inter_constantly <- df_clean %>%
  filter(RADIO == 1 & INTFREQ == 1) %>%  # '&' is used for the logical AND to
  ↪  filter for people who listen to the radio AND use the internet almost
  ↪  constantly
  nrow()  # Counts the number of rows that meet both conditions

# Step 5: Calculate probabilities for individual events and their intersection
prob_event_A <- n_radio_yes / n_total_q1b  # Probability of listening to the
↪  radio (Event A)
prob_event_B <- n_inter_constantly / n_total_q1b  # Probability of using the
↪  internet almost constantly (Event B)
prob_A_and_B <- n_radio_yes_AND_inter_constantly / n_total_q1b  # Probability of
↪  both listening to the radio AND using the internet almost constantly (A and
↪  B)

# Step 6: Calculate the probability of either listening to the radio OR using the
↪  internet almost constantly
# Using the formula P(A or B) = P(A) + P(B) - P(A and B) to avoid double-counting
↪  overlap
```

```r
prob_question_1b <- prob_event_A + prob_event_B - prob_A_and_B

# Step 7: Verify calculation using OR logic (alternative approach)
# This calculates the number of respondents who meet either condition (radio OR
↪  internet almost constantly)
n_A_and_B <- df_clean %>%
  filter(RADIO == 1 | INTFREQ == 1) %>%  # Logical OR condition selects
  ↪  respondents who meet either condition
  nrow()
prob_A_and_B <- n_A_and_B / n_total_q1b  # Calculate probability based on OR
↪  condition

# Step 8: Print results to verify consistency between calculated and alternative
↪  method
print(prob_A_and_B)  # Print probability from OR calculation
```

```
## [1] 0.8907482
```

```r
print(prob_question_1b)  # Print probability from formula calculation
```

```
## [1] 0.8907482
```

Explanation: The probability that a respondent listens to the radio or uses the internet almost constantly is equal to 89 percent. We used the general OR formula:

$$\Pr(A \text{ or } B) = \Pr(A) + \Pr(B) - \Pr(A \text{ and } B)$$

This formula accounts for the overlap between the two groups to avoid double-counting individuals who both listen to the radio and use the internet almost constantly. Additionally, we verified this calculation using an alternative method that directly counts the respondents meeting either condition, which yielded the same result.

Answer - Question (1.c)

What is the probability that a respondent believes there would be more crime if more Americans owned guns and describes their community as somewhat safe or safer?

```r
# Step 1: Calculate the total number of respondents
n_total_q1c <- nrow(df_clean)  # This is the total number of respondents in the
↪  dataset


# Construct binary variable for CRIMESAFE and MOREGUNIMPACT given the question
↪  prompt
df_clean <- df_clean %>%
  mutate(somwhsafe_or_safer = if_else(CRIMESAFE %in% c(1,2,3), 1, 0),
         guns_more_crime = if_else(MOREGUNIMPACT == 1, 1, 0)
         )
```

```r
# Step 2: Count respondents with both somwhsafe_or_safer==1 (event A) AND
↪  guns_more_crime==1 (event B)
n_A_and_B <- df_clean %>%
  filter(somwhsafe_or_safer==1 & guns_more_crime==1) %>%
  nrow()

# Step 3: Compute the probability.
prob_q1c <- n_A_and_B / n_total_q1c

print(prob_q1c)
```

```
## [1] 0.3869975
```

Explanation: Recall the coding of `CRIMESAFE`. 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

Also, recall the coding for `MOREGUNIMPACT`. If more Americans owned guns, do you think there would be...

- 1 = More crime;
- 2 = Less crime;
- 3 = No difference

A key step then is defining the binary variables per question prompt requirements.

Binary Variable Definitions:

1. **Binary Variable for Safety (`somwhsafe_or_safer`)**: This variable is 1 if the respondent describes their community as "somewhat safe" or safer, otherwise 0.

$$\texttt{somwhsafe\_or\_safer} = \begin{cases} 1 & \text{if } CRIMESAFE \in \{1, 2, 3\} \\ 0 & \text{if } CRIMESAFE \in \{4, 5\} \end{cases}$$

2. **Binary Variable for Gun Impact (`guns_more_crime`)**: This variable is 1 if the respondent believes that more Americans owning guns would lead to more crime, otherwise 0.

$$\texttt{guns\_more\_crime} = \begin{cases} 1 & \text{if } MOREGUNIMPACT = 1 \\ 0 & \text{if } MOREGUNIMPACT \in \{2, 3\} \end{cases}$$

These definitions reflect how we classified each respondent into binary categories based on their responses to the survey questions.

The probability that a respondent believes there would be more crime if more Americans owned guns and describes their community as somewhat safe or safer is equal to 39 percent. We used the

classical probability definition:

$$\Pr(A \text{ and } B) = \frac{\text{Cases with both A and B}}{\text{Total Cases in Sample Space}}$$

This calculation shows how we determine the likelihood that both events occur simultaneously.

Answer - Question (1.d)

Given that a respondent describes their community as somewhat safe or safer, what is the probability that they believe there would be more crime if more Americans owned guns?

```
n_total_sample <- nrow(df_clean)

n_total_event_B <- df_clean %>%
  filter(somwhsafe_or_safer==1) %>%
  nrow()

n_A_and_B <- df_clean %>%
  filter(somwhsafe_or_safer==1 & guns_more_crime==1) %>%
  nrow()

prob_A_and_B <- n_A_and_B / n_total_sample
prob_B <- n_total_event_B / n_total_sample

# Conditional Probability of A given B:
prob_A_given_B <- prob_A_and_B / prob_B

# Print.
print(paste0("The probability is: ", round(prob_A_given_B,3)))
```

```
## [1] "The probability is: 0.433"
```

Explanation: Define **Event B** as "respondent describes their community as somewhat safe or safer." Additionally, define **Event A** as "they believe there would be more crime if more Americans owned guns." Then,

$$\Pr(A|B) = \frac{P(A \text{ and } B)}{P(B)}$$

Thus, given that a respondent describes their community as somewhat safe or safer, the probability that they believe there would be more crime if more Americans owned guns is equal to 43 percent. This conditional probability helps us understand the likelihood of one event occurring given the occurrence of another event.

Answer - Question (1.e)

Create a 2x2 table examining the relationship between:

- Perceptions of economic conditions in the community (grouped as "Excellent/Good" vs. "Only fair/Poor")

- Support for increasing government spending on roads and bridges (grouped as "Increase a lot/little" vs. "Stay the same/Decrease")

Using this table, calculate the conditional probability that a respondent supports increased spending, given that they perceive economic conditions as Excellent or Good.

```
# Step 1: Create new binary variables for economic perceptions and spending
↪  support
df_clean <- df_clean %>%
  mutate(
    econ_excell_good = if_else(ECON1MOD %in% c(1, 2), 1, 0),  # 1 for
    ↪  "Excellent/Good", 0 for "Only fair/Poor"
    incrspend_alot_alittle = if_else(INFRASPEND %in% c(1, 2), 1, 0)  # 1 for
    ↪  "Increase a lot/little", 0 for "Stay the same/Decrease"
  )

# Step 2: Use dplyr to create the 2x2 contingency table
contingency_table <- df_clean %>%
  group_by(econ_excell_good, incrspend_alot_alittle) %>%
  summarise(count = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'econ_excell_good'. You can override using
## the `.groups` argument.
```

```
print(contingency_table)
```

```
## # A tibble: 4 x 3
##   econ_excell_good incrspend_alot_alittle count
##              <dbl>                  <dbl> <int>
## 1                0                      0  1125
## 2                0                      1  1892
## 3                1                      0   757
## 4                1                      1  1425
```

```
# We need some changes to get it to the correct 2x2 format
contingency_table <- contingency_table %>%
  mutate(econ_excell_good = if_else(econ_excell_good==1,
                                    'Econ: Excellent/Good', 'Econ: Only
                                    ↪  fair/Poor'),
         incrspend_alot_alittle = if_else(incrspend_alot_alittle==1,
                                          'Spend: Increase a lot/little', 'Spend:
                                          ↪  Stay the same/Decrease')) %>%
  tidyr::spread(econ_excell_good, count, fill = 0)  # Spread into a wide format
↪  table

# Step 3: Print the contingency table
print(contingency_table)
```

```
## # A tibble: 2 x 3
```

```
##   incrspend_alot_alittle        `Econ: Excellent/Good` `Econ: Only fair/Poor`
##   <chr>                                          <dbl>                  <dbl>
## 1 Spend: Increase a lot/little                    1425                   1892
## 2 Spend: Stay the same/Decrease                    757                   1125
```

```
knitr::kable(contingency_table, format='latex')
```

incrspend_alot_alittle	Econ: Excellent/Good	Econ: Only fair/Poor
Spend: Increase a lot/little	1425	1892
Spend: Stay the same/Decrease	757	1125

```
# Step 4: Calculate the conditional probability of supporting increased spending
↪  given Excellent/Good economic conditions
n_tot_econ_good <- sum(contingency_table$`Econ: Excellent/Good`)
n_tot_incr_spend_AND_econ_good <- contingency_table %>%
  filter(incrspend_alot_alittle=='Spend: Increase a lot/little') %>%
  pull(`Econ: Excellent/Good`)

# Calculate the conditional probability
conditional_prob_incrspend_given_econ_good <- n_tot_incr_spend_AND_econ_good /
↪  n_tot_econ_good

# Print the conditional probability as a percentage
cat(paste0("Conditional Probability of supporting increased spending \ngiven
↪  Excellent/Good economic conditions: ",
           round(conditional_prob_incrspend_given_econ_good * 100, 2), "%"))
```

```
## Conditional Probability of supporting increased spending
## given Excellent/Good economic conditions: 65.31%
```

Explanation: Recall:

`INFRASPEND`. Thinking about government spending on roads and bridges in the area where you live, do you think this spending should...

- 1 = Increase a lot;
- 2 = Increase a little;
- 3 = Stay about the same;
- 4 = Decrease a little;
- 5 = Decrease a lot

`ECON1MOD`. How would you rate economic conditions in your community today?

- 1 = Excellent;
- 2 = Good;
- 3 = Only fair;
- 4 = Poor

Then, the probability that a respondent supports increased spending, given that they perceive economic conditions as Excellent or Good is equal to 65.31% percent. By grouping economic perceptions and support for spending, we can better understand how positive economic outlooks influence support for public investment.

Question 2 - Solutions

Answer - Question (2.a)

Calculate the Expected Value of the Number of Participants

```
# Define the possible values of the number of participants (in thousands)
P_values <- c(50000, 75000, 100000, 125000, 150000)

# Define the corresponding probabilities of each value of P
P_probs <- c(0.08, 0.22, 0.31, 0.24, 0.15)

# Calculate the expected value of the number of participants
E_P <- sum(P_values * P_probs)

# Print the expected value
print(paste0("Expected value of the number of participants: ", E_P))
```

```
## [1] "Expected value of the number of participants: 104000"
```

Explanation: The expected value of the number of participants is calculated using the formula for the expected value of a discrete random variable:

$$E[P] = \sum p_i x_i$$

Where p_i represents the probability of each possible outcome, and x_i represents each possible outcome.

Answer - Question (2.b)

Define the Total Monthly Cost as a Linear Combination

```
# Define fixed administrative cost and monthly benefit per participant
fixed_cost <- 6000000
monthly_benefit_per_participant <- 200

# Total monthly cost as a linear combination of P (number of participants)
# C = fixed_cost + (monthly_benefit_per_participant * P)
cat("Total Monthly Cost (C):", "C =", fixed_cost, "+",
↪  monthly_benefit_per_participant, "* P")
```

```
## Total Monthly Cost (C): C = 6e+06 + 200 * P
```

Explanation: The total monthly cost, C, is expressed as a linear combination of the number of participants, P, the monthly benefit per participant, and the fixed administrative cost. The fixed cost remains constant, while the cost associated with participants is proportional to the number of participants.

$$C = 6000000 + 200P$$

Answer - Question (2.c)

Compute the Probability Mass Function of the Total Monthly Cost

```
# Calculate the possible values of the total monthly cost
C_values <- fixed_cost + (monthly_benefit_per_participant * P_values)

# Probability Mass Function of the total monthly cost
C_probs <- P_probs

# Create a data frame to display the probability mass function
pmf_C <- data.frame(Total_Monthly_Cost = C_values, Probability = C_probs)

# Print the probability mass function
print(pmf_C)
```

```
##   Total_Monthly_Cost Probability
## 1            1.6e+07        0.08
## 2            2.1e+07        0.22
## 3            2.6e+07        0.31
## 4            3.1e+07        0.24
## 5            3.6e+07        0.15
```

Explanation: The probability mass function (PMF) of the total monthly cost is calculated by using the possible values of P to determine the corresponding costs. Since the number of participants has a discrete distribution, the total cost also has a corresponding probability for each value.

Total_Monthly_Cost	Probability
1.6e+07	0.08
2.1e+07	0.22
2.6e+07	0.31
3.1e+07	0.24
3.6e+07	0.15

Answer - Question (2.d)

Compute the Expected Value of the Total Monthly Cost

```
# Calculate the expected value of the total monthly cost
E_C <- sum(C_values * C_probs)

# Print the expected value of the total monthly cost
print(paste0("Expected value of the total monthly cost: ", E_C))
```

```
## [1] "Expected value of the total monthly cost: 26800000"
```

Explanation: The expected value of the total monthly cost is calculated by taking the sum of the product of each possible value of the total cost and its corresponding probability. This gives us the long-run average cost of the program.

Answer - Question (2.e)

Compute the Cumulative Distribution Function of the Total Monthly Cost

```
# Calculate the cumulative distribution function (CDF) of the total monthly cost
CDF_C <- cumsum(C_probs)

# Create a data frame to display the CDF
cdf_C <- data.frame(Total_Monthly_Cost = C_values, Cumulative_Probability =
↪  CDF_C)

# Print the cumulative distribution function
print(cdf_C)
```

```
##   Total_Monthly_Cost Cumulative_Probability
## 1            1.6e+07                    0.08
## 2            2.1e+07                    0.30
## 3            2.6e+07                    0.61
## 4            3.1e+07                    0.85
## 5            3.6e+07                    1.00
```

Explanation: The cumulative distribution function (CDF) of the total monthly cost is calculated by taking the cumulative sum of the probabilities. This helps us determine the probability that the total monthly cost will be less than or equal to a certain value.

Total_Monthly_Cost	Cumulative_Probability
1.6e+07	0.08
2.1e+07	0.30
2.6e+07	0.61
3.1e+07	0.85
3.6e+07	1.00

Answer - Question (2.f)

Assess Budget Constraints

```
# Define the maximum monthly budget
max_budget <- 30000000

# Calculate the probability that the cost exceeds the budget
prob_exceeds_budget <- 1 - CDF_C[which(C_values <=
↪  max_budget)][length(which(C_values <= max_budget))]

# Print the probability that the cost exceeds the budget
print(paste0("Probability that the cost exceeds the budget: ",
↪  round(prob_exceeds_budget, 2)))
```

```
## [1] "Probability that the cost exceeds the budget: 0.39"
```

Explanation: The probability that the cost exceeds the allocated monthly budget of $30,000,000 is calculated by finding the complement of the CDF up to that value. This tells us the likelihood

that the costs will surpass the budget constraints. We find that the probability that the maximum budget is exceeded is 39%.