Luc Mekouar 3617696

Problem Set 1+2 (15% + 15%)

Due: 2023-12-3 23:59 (HKT)

General Introduction

In this Problem Set, you will apply data science skills to wrangle and visualize the replication data of the following research article:

Cantú, F. (2019). The fingerprints of fraud: Evidence from Mexico's 1988 presidential election. *American Political Science Review*, *113*(3), 710-726.

Requirements and Reminders

- You are required to use **RMarkdown** to compile your answer to this Problem Set.
- Two submissions are required (via Moodle)
  - A `.pdf` file rendered by `Rmarkdown` that contains all your answer.
  - A compressed (in `.zip` format) R project repo. The expectation is that the instructor can unzip, open the project file, knitr your `.Rmd` file, and obtain the exact same output as the submitted `.pdf` document.
- The Problem Set is worth 30 points in total, allocated across 7 tasks. The point distribution across tasks is specified in the title line of each task. Within each task, the points are evenly distributed across sub-tasks. Bonus points (+5% max.) will be awarded to recognize exceptional performance.
- Grading rubrics: Overall, your answer will be evaluated based on its quality in three dimensions
  - Correctness and beauty of your outputs
  - Style of your code
  - Insightfulness of your interpretation or discussion
- Unless otherwise specified, you are required to use functions from the `tidyverse` package to complete this assignments.
- Fo some tasks, they may be multiple ways to achieve the same desired outcomes. You are encouraged to explore multiple methods. If you perform a task using multiple methods, do show it in your submission. You may earn bonus points for it.
- You are encouraged to use Generative AI such as ChatGPT to assist with your work. However, you will need to acknowledge it properly and validate AI's outputs. You may attach selected chat history with the AI you use and describe how it helps you get the work done. Extra credit may be rewarded to recognize creative use of Generative AI.
- This Problem Set is an individual assignment. You are expected to complete it independently. Clarification questions are welcome. Discussions on concepts and techniques related to the Problem Set among peers is encouraged. However, without the instructor's consent, sharing (sending and requesting) code and text that complete the entirety of a task is prohibited. You are strongly encouraged to use *CampusWire* for clarification questions and discussions.

Background

In 1998, Mexico had a close presidential election. Irregularities were detected around the country during the voting process. For example, when 2% of the vote tallies had been counted, the preliminary results showed the PRI's imminent defeat in Mexico City metropolitan area and a very narrow vote margin between PRI and FDN. A few minutes later, the screens at the Ministry of Interior went blank, an event that electoral authorities justified as a technical problem caused by an overload on telephone lines. The vote count was therefore suspended for three days, despite the fact that opposition representatives found a computer in the basement that continued to receive electoral results. Three days later, the vote count resumed, and soon the official announced PRI's winning with 50.4% of the vote.

What happened on that night and the following days? Were there electoral fraud during the election? A political scientist, Francisco Cantú, unearths a promising dataset that could provide some clues. At the National Archive in Mexico City, Cantú discovered about 53,000 vote tally sheets. Using machine learning methods, he detected that a significant number of tally sheets were *altered*! In addition, he found evidence that the altered tally sheets were biased in favor of the incumbent party. In this Problem Set, you will use Cantú's replication dossier to replicate and extend his data work.

Please read Cantú (2019) for the full story. And see Figure 1 for a few examples of altered (fraudulent) tallies.

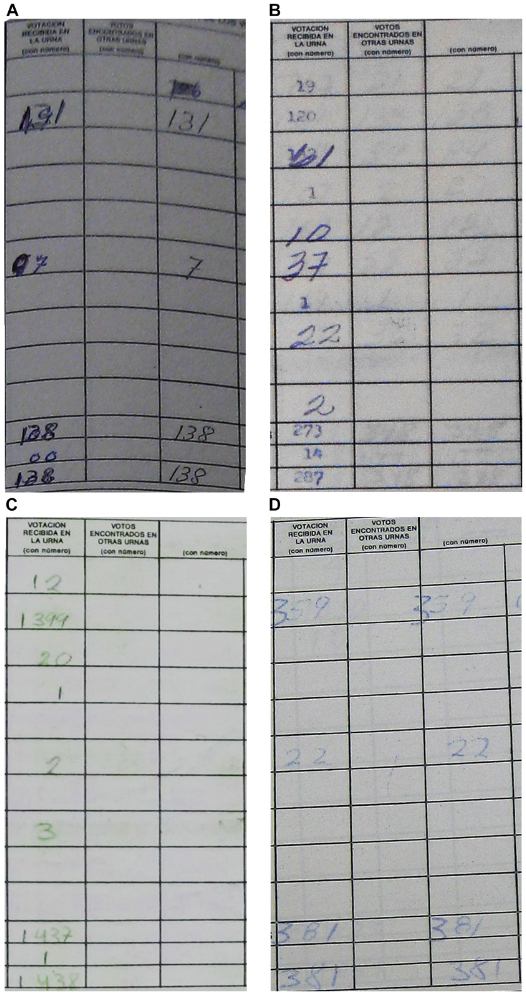

A
B
C
D
VOTACION RECIBIDA EN LA URNA (con número)
VOTOS ENCONTRADOS EN OTRAS URNAS (con número)
(con número)

Figure 1: Examples of altered tally sheets (reproducing Figure 1 of Cantú 2018)

Task 0. Loading required packages (3pt)

For Better organization, it is a good habit to load all required packages up front at the start of your document. Please load the all packages you use throughout the whole Problem Set here.

```r
library(tidyverse)
```

Task 1. Clean machine classification results (3pt)

Cantú applys machine learning models to 55,334 images of tally sheets to detect signs of fraud (i.e., alteration). The machine learning model returns results recorded in a table. The information in this table is messy and requires data wrangling before we can use them.

Task 1.1. Load classified images of tally sheets

The path of the classified images of tally sheets is `data/classification.txt`. Your first task is loading these data onto R using a `tidyverse` function. Name it `d_tally`.

Note:

- Although the file extension of this dataset is `.txt`, you are recommended to use the `tidyverse` function we use for `.csv` files to read it.
- Unlike the data files we have read in class, this table has *no column names*. Look up the documentation and find a way to handle it.
- There will be three columns in this dataset, name them `name_image`, `label`, and `probability`.

Print your table to show your output.

```r
d_tally <- read_csv("data/classification.txt", col_names = FALSE) |>
  rename("name_image"="X1","label"="X2","probability"="X3")
```

Note 1. What are in this dataset?

Before you proceed, let me explain the meaning of the three variables.

- `name_image` contains the names of of the tallies' image files (as you may infer from the `.jpg` file extensions. They contain information about the locations where each of the tally sheets are produced.
- `label` is a machine-predicted label indicating whether a tally is fraudulent or not. `label = 1` means the machine learning model has detected signs of fraud in the tally sheet. `label = 0` means the machine detects no sign of fraud in the tally sheet. In short, `label = 1` means fraud; `label = 0` means no fraud.
- `probability` indicates the machine's certainty about its predicted `label` (explained above). It ranges from 0 to 1, where higher values mean higher level of certainty.

Interpret `label` and `probability` carefully. Two examples can hopefully give you clues about their correct interpretation. In the first row, `label = 0` and `probability = 0.9991`. That means the machine thinks this tally sheet is NOT FRAUDULENT with a probability of 0.9991. Then, the probability that this tally sheet is fraudulent is `1 - 0.9991 = 0.0009`. Take another example, in the 11th row, `label = 1` and `probability = 0.935`. This means the machine thinks this tally sheet IS FRAUDULENT with a probability of 0.935. Then, the probability that it is NOT FRAUDULENT is `1 - 0.9354 = 0.0646`.

Task 1.2. Clean columns label and probability

As you have seen in the printed outputs, columns label and probability are read as chr variables when they are actually numbers. A close look at the data may tell you why — they are "wrapped" by some non-numeric characters. In this task, you will clean these two variables and make them valid numeric variables. You are required to use tidyverse operations to for this task. Show appropriate summary statistics of label and probability respectively after you have transformed them into numeric variables.

```
d_tally <- d_tally |>
  mutate(label = parse_number(label),probability = parse_number(probability))
```

```
d_tally |>
  summarise(percentage_fraudulent = 100*mean(label,na.rm = TRUE),avg_model_certainty = mean(probability
```

Task 1.3. Extract state and district information from `name_image`

As explained in the note, the column `name_image`, which has the names of tally sheets' images, contains information about locations where the tally sheets are produced. Specifically, the first two elements of these file names indicates the **`states'`** and districts' identifiers respectively, for example, `name_image = "Aguascalientes_I_2014-05-26 00.00.10.jpg"`. It means this tally sheet is produced in state **`Aguascalientes`**, district `I`. In this task, you are required to obtain this information. Specifically, create two columns named `state` and `district` as state and district identifiers respectively. You are required to use `tidyverse` functions to perform the task.

```r
d_tally <- d_tally |>
  mutate(state = str_extract(name_image, "^[^_]+"),
         district = str_extract(name_image,"(?<=_)[^_]+(?=_)"))
# use of R documentation to understand function "str_exact", and creat relevant argument (pattern)
```

Task 1.4. Re-code a state's name

One of the states (in the newly created column `state`) is coded as "`Estado de Mexico`." The researchers decide that it should instead re-coded as "**`Edomex`**." Please use a `tidyverse` function to perform this task.

Hint: Look up functions `ifelse` and `case_match`.

```r
d_tally <- d_tally |>
  mutate(state = ifelse(state == "Estado de Mexico","Edomex",state))
```

Task 1.5. Create a *probability of fraud* indicator

As explained in Note 1, we need to interpret `label` and `probability` with caution, as the meaning of `probability` is conditional on the value of `label`. To avoid confusion in the analysis, your next task is to create a column named `fraud_proba` which indicates the probability that a tally sheet is is fraudulent. After you have created the column, drop the `label` and `probability` columns.

Hint: Look up the `ifelse` function and the `case_when` function (but you just need either one of them).

```r
d_tally <- d_tally |>
  mutate(fraud_proba = ifelse(label == 1,probability,1 - probability)) |>
  select(-label,-probability)
```

Task 1.6. Create a binary *fraud* indicator

In this task, you will create a binary indicator called `fraud_bin` in indicating whether a tally sheet is fraudulent. Following the researcher's rule, we consider a tally sheet fraudulent only when the machine thinks it is at least 2/3 likely to be fraudulent. That is, `fraud_bin` is set to TRUE when `fraud_proba` is greater to `2/3` and is FALSE otherwise.

```r
d_tally <- d_tally |>
  mutate(fraud_bin = ifelse(fraud_proba >= 2/3,TRUE,FALSE))
```

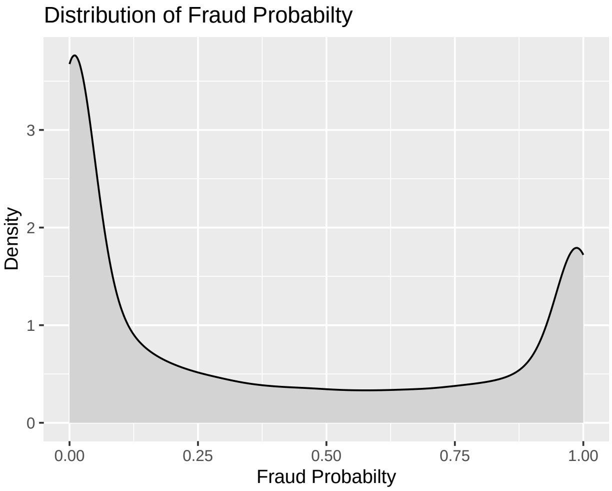
Task 2. Visualize machine classification results (3pt)

In this section, you will visualize the `tally` dataset that you have cleaned in Task 1. Unless otherwise specified, you are required to use the `ggplot` packages to perform all the tasks.

Task 2.1. Visualize distribution of `fraud_proba`

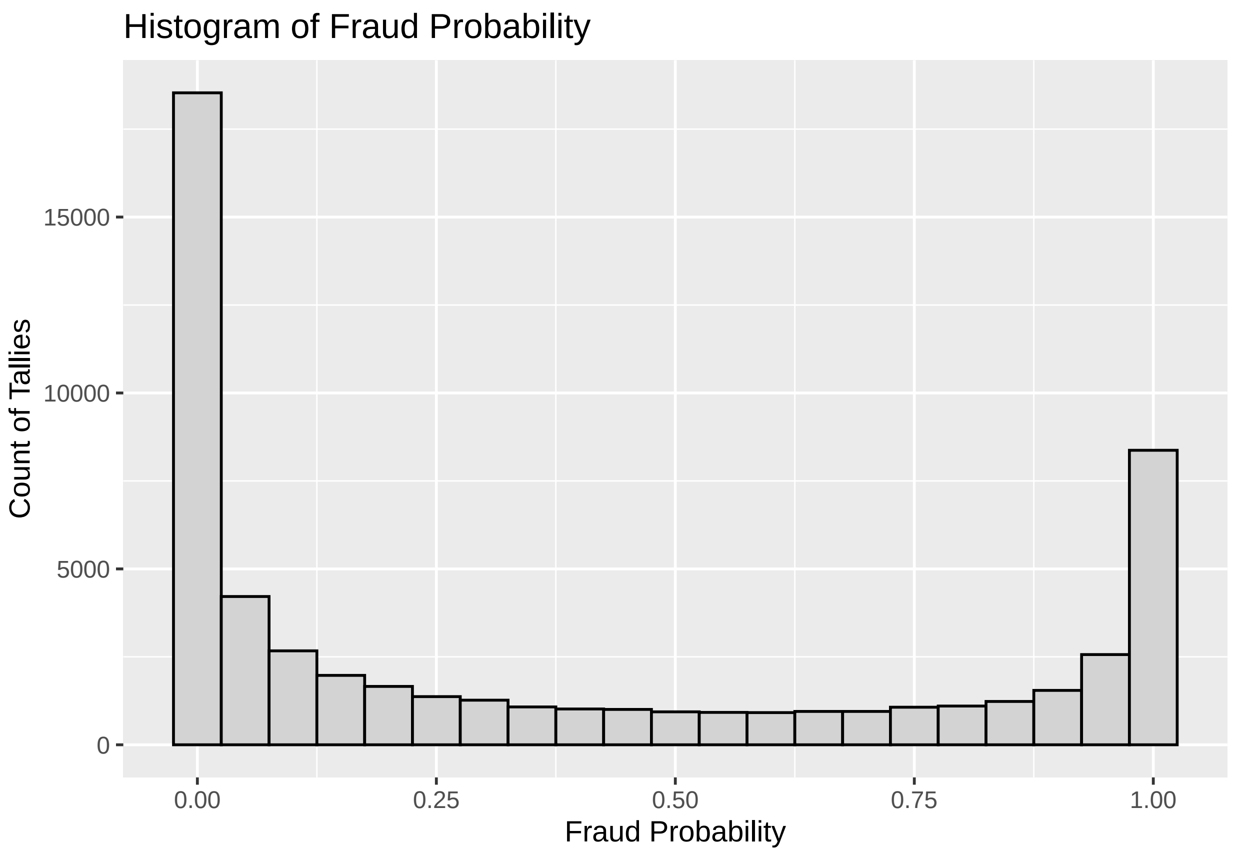
How is the predicted probability of fraud (`fraud_proba`) distributed? Use two methods to visualize the distribution. Remember to add informative labels to the figure. Describe the plot with a few sentences.

```
d_tally |>
  ggplot(aes(x = fraud_proba)) +
  geom_density(fill = "lightgray", color = "black") +
  labs(x = "Fraud Probabilty", y = "Density", title = "Distribution of Fraud Probabilty")
```



```
# This plot represents the distribution of Fraud Probability, with the total area under the curve
# equaling 1. We can clearly see a U shape, showing many very likely fraudulent tallies, and very
# unlikely fraudulent tallies, with few in between.
```

```
d_tally |>
  ggplot(aes(x = fraud_proba)) +
  geom_histogram(binwidth = 0.05, fill = "lightgray", color = "black") +
  labs(x = "Fraud Probability", y = "Count of Tallies",
       title = "Histogram of Fraud Probability")
```

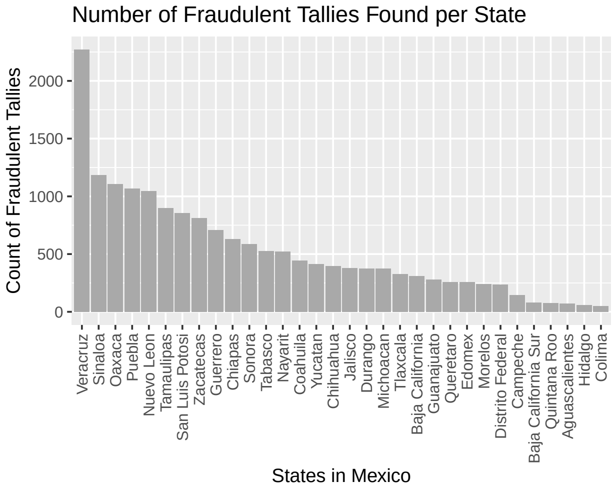


```
# This Histogram is also very informative as it shows actual counts of tallies
# in 20 smaller intervals between very likely to be fraudulent, to very unlikely
# to be fraudulent. While the Distribution is more precise, this Histogram's
# instrumental value lies in the units on the y-axis, tally count, which is
# likely more useful to the viewer.
```

Task 2.2. Visualize distribution of `fraud_bin`

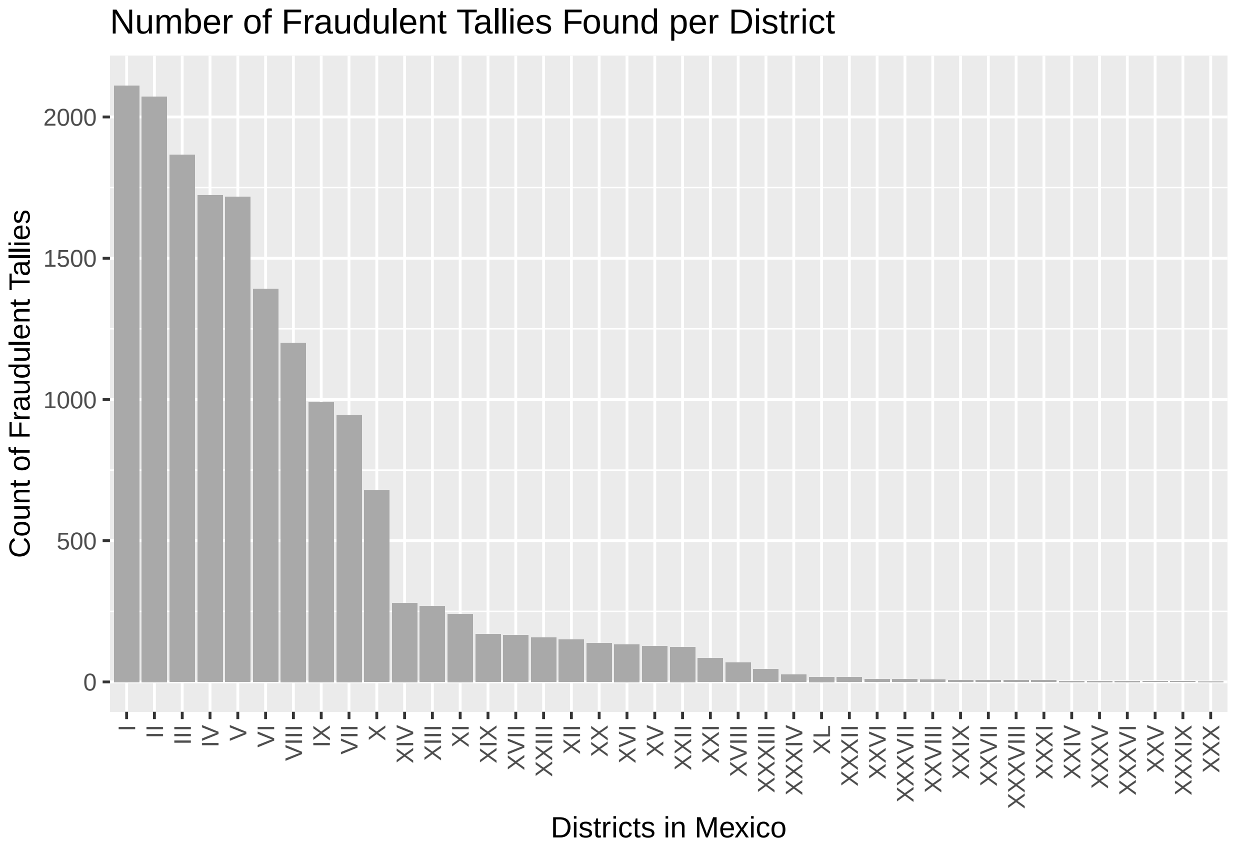
How many tally sheets are fraudulent and how many are not? We may answer this question by visualizing the binary indicator of tally-level states of fraud. Use at least two methods to visualize the distribution of `fraud_bin`. Remember to add informative labels to the figure. Describe your plots with a few sentences.

```r
d_tally |>
  group_by(state) |>
  summarise(n_obs = sum(fraud_bin, na.rm = TRUE)) |>
  ggplot(aes(x = reorder(state, -n_obs), y = n_obs)) +
  geom_bar(stat = "identity", fill = "darkgray") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  labs(x = "States in Mexico", y = "Count of Fraudulent Tallies",
       title = "Number of Fraudulent Tallies Found per State")
```



```r
# This Histogram sheds light on the count of fraudulent tallies per State,
# ordered by number of fraudulent tallies. Note that this Histogram only shows
# total fraudulent tallies, and not percentage of fraudulent tallies per State.
```

```r
d_tally |>
  group_by(district) |>
  summarise(n_obs = sum(fraud_bin, na.rm = TRUE)) |>
  ggplot(aes(x = reorder(district, -n_obs), y = n_obs)) +
  geom_bar(stat = "identity", fill = "darkgray") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  labs(x = "Districts in Mexico", y = "Count of Fraudulent Tallies",
       title = "Number of Fraudulent Tallies Found per District")
```



```
# This Histogram sheds light on the count of fraudulent tallies per District,
# ordered by number of fraudulent tallies. Note that this Histogram only shows
# total fraudulent tallies, and not percentage of fraudulent tallies per
# District.
```

The figure below serve as a reference. Feel free to try alternative approach(es) to make your visualization nicer and more informative.

Task 2.3. Summarize prevalence of fraud by state

Next, we will examine the between-state variation with regards to the prevalence of election fraud. In this task, you will create a new object that contains two state-level indicators regarding the prevalence of election fraud: The count of fraudulent tallies and the proportion of fraudulent tallies.

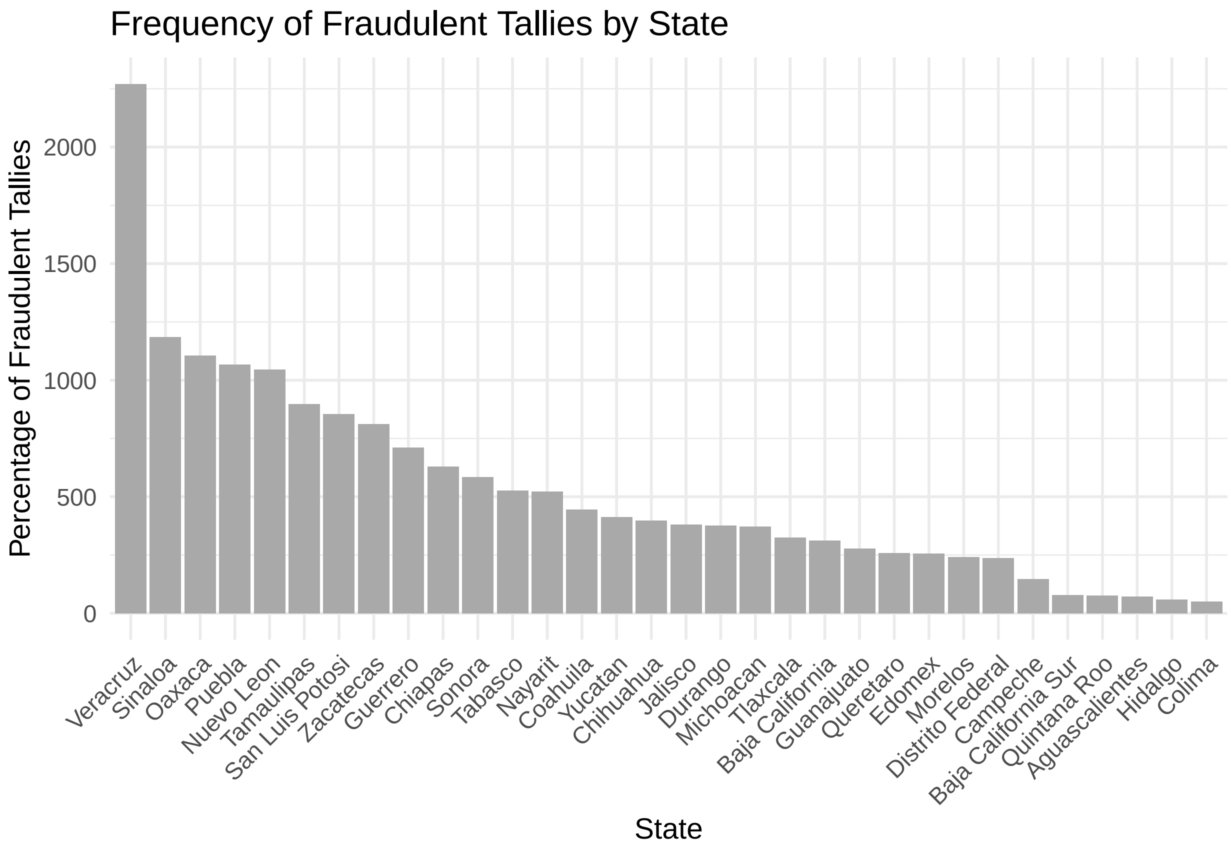
```
state_fraudulent_obj <- d_tally |>
  group_by(state) |>
    summarise(count_fraud_state = sum(fraud_bin, na.rm = TRUE),
    percentage_fraud_state = 100 * sum(fraud_bin, na.rm = TRUE) / n())
```

Task 2.4. Visualize frequencies of fraud by state

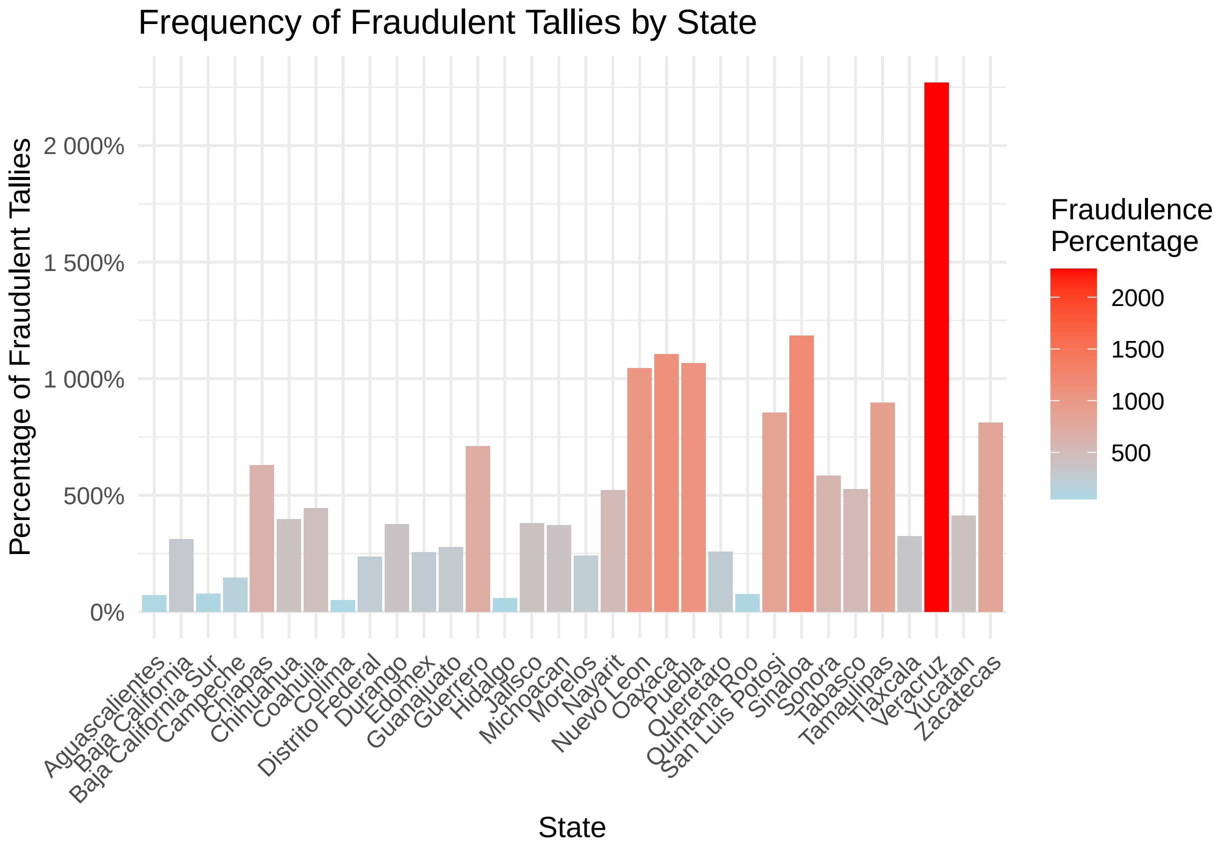
Using the new data frame created in Task 2.3, please visualize the *frequencies* of fraudulent tallies of every state. Describe the key takeaway from the visualization with a few sentences.

Feel free to try alternative approach(es) to make your visualization nicer and more informative.

```r
# The graphs below emphasis the amount of fraudulent tallies in many states.
# More specifically, one can see that "Veracruz" has by very far the highes
# number of fraudulent tallies. The other states seam to have more or less the
# same number of fraudulent tallies.

# Method 1:
state_fraudulent_obj |>
  mutate(state = reorder(state, -count_fraud_state)) |>
  ggplot(aes(x = state, y = count_fraud_state)) +
  geom_bar(stat = "identity", fill = "darkgray") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(y = "Percentage of Fraudulent Tallies",
       x = "State",
       title = "Frequency of Fraudulent Tallies by State")
```



```r
# Method 2:
state_fraudulent_obj |>
  ggplot(aes(x = reorder(state, state), y = count_fraud_state,
             fill = count_fraud_state)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(y = "Percentage of Fraudulent Tallies",
       x = "State",
       title = "Frequency of Fraudulent Tallies by State",
       fill = "Fraudulence\nPercentage") +
  scale_y_continuous(labels = scales::percent_format(scale = 1)) +
  scale_fill_gradient(low = "lightblue", high = "red")
```



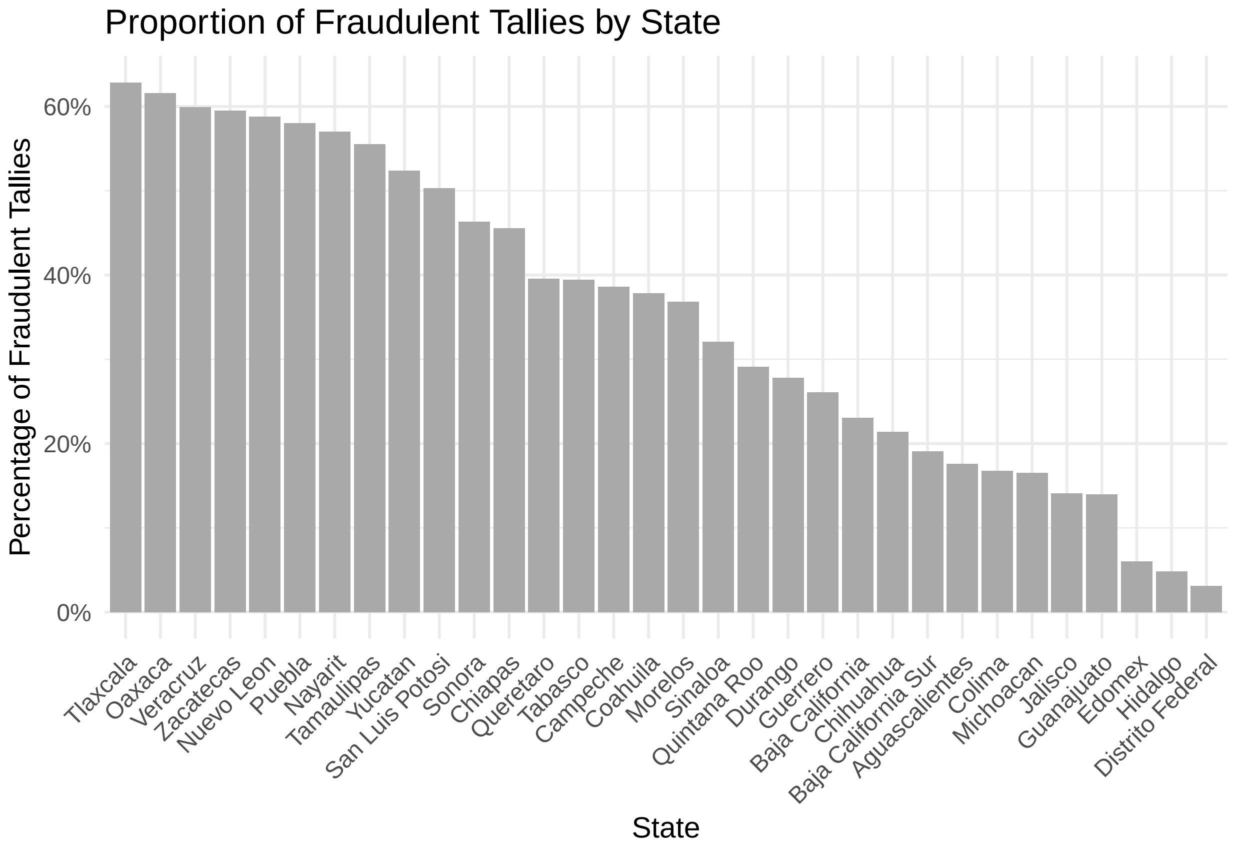
```
# some help from ChatGPT to write correctly this code chunk,
# especially the use of "scale_fill_gradient"
```

Task 2.5. Visualize proportions of fraud by state

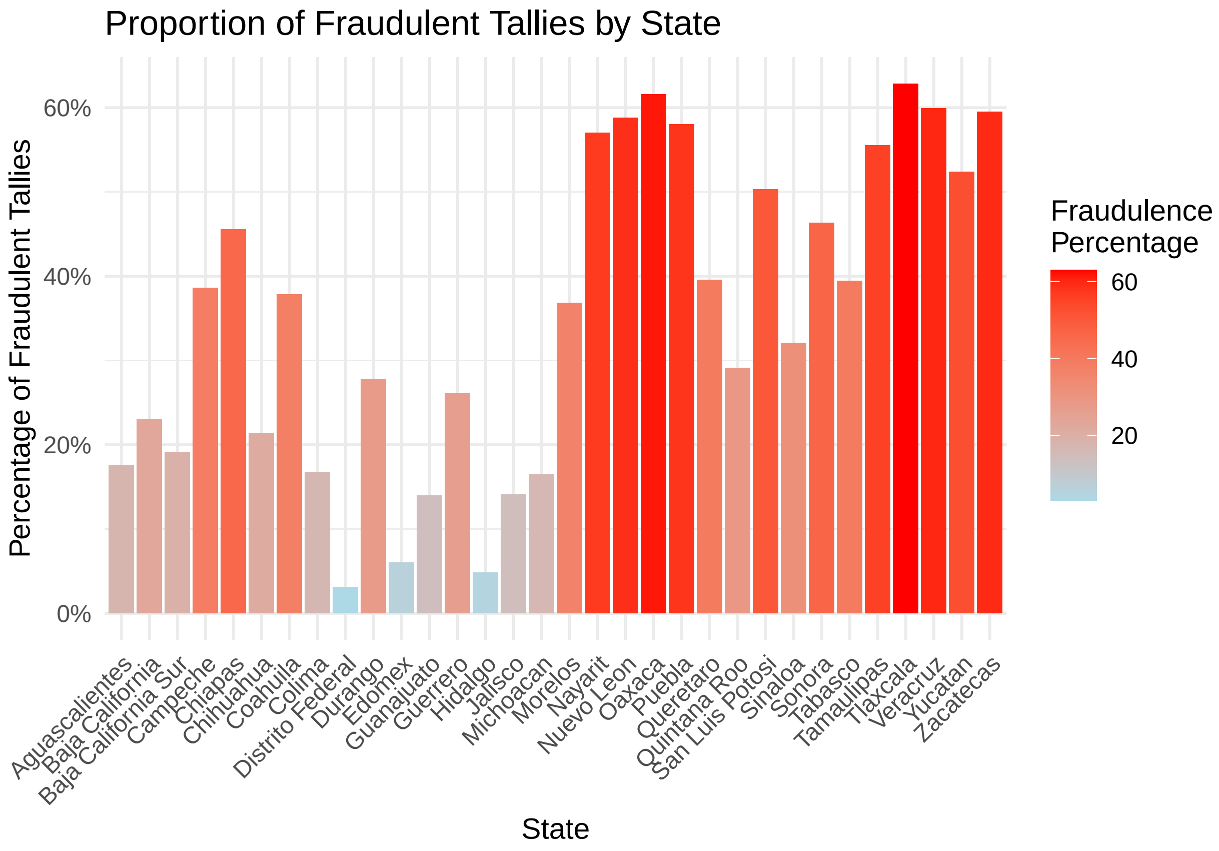
Using the new data frame created in Task 2.3, please visualize the *proportion of* of fraudulent tallies of every state. Describe the key takeaway from the visualization with a few sentences.

Feel free to try alternative approach(es) to make your visualization nicer and more informative.

```
# The graphs below emphasis the amount of fraudulent tallies in many states.
# Only 3 out of the 32 states in the data set have near zero fraudulent tally
# frequency. However multiple states have close to or more than 60% of there
# tallies fraudulent.

# Method 1:
state_fraudulent_obj |>
  mutate(state = reorder(state, -percentage_fraud_state)) |>
  ggplot(aes(x = state, y = percentage_fraud_state)) +
  geom_bar(stat = "identity", fill = "darkgray") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(y = "Percentage of Fraudulent Tallies",
       x = "State",
       title = "Proportion of Fraudulent Tallies by State") +
  scale_y_continuous(labels = scales::percent_format(scale = 1))
```



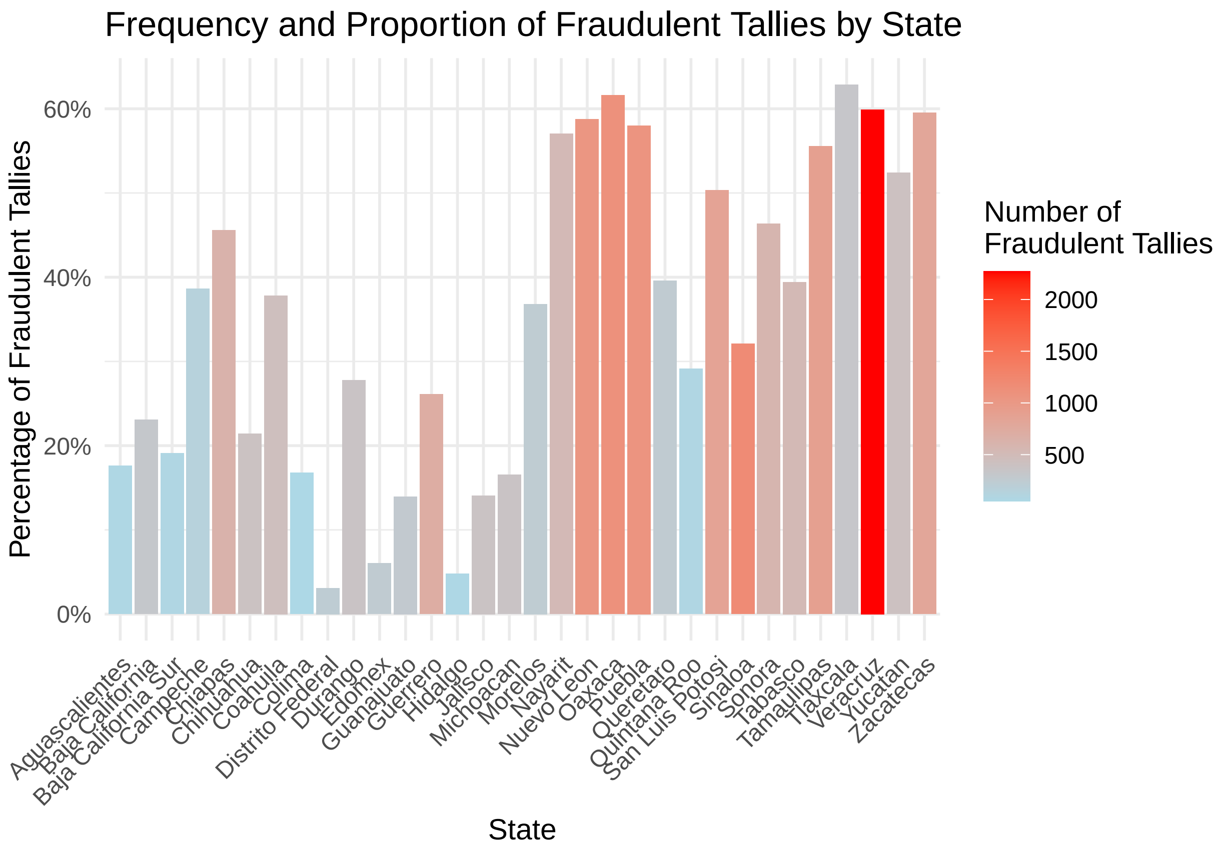
```r
# Method 2:
state_fraudulent_obj |>
  ggplot(aes(x = reorder(state, state), y = percentage_fraud_state,
             fill = percentage_fraud_state)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(y = "Percentage of Fraudulent Tallies",
       x = "State",
       title = "Proportion of Fraudulent Tallies by State",
       fill = "Fraudulence\nPercentage") +
  scale_y_continuous(labels = scales::percent_format(scale = 1)) +
  scale_fill_gradient(low = "lightblue", high = "red")
```



Task 2.6. Visualize both proportions & frequencies of fraud by state

Create data visualization to show BOTH the *proportions* and *frequencies* of fraudulent tally sheets by state in one figure. Include annotations to highlight states with the highest level of fraud. Add informative labels to the figure. Describe the takeaways from the figure with a few sentences.

```r
# The Number of Fraudulent Tallies per state showed by the colour of the bars
# reveales what looks like an outlier,"Veracruz", which seamingly has
# significantly more Fraudulent Tallies than all the other states. However, by
# looking at the bar graph (specifically it's shape), one can see that this
# state does not seam to be an outlier in terms of proportion of Fraudulent
# tallies.

state_fraudulent_obj |>
  ggplot(aes(x = reorder(state, state), y = percentage_fraud_state,
             fill = count_fraud_state)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(y = "Percentage of Fraudulent Tallies",
       x = "State",
       title = "Frequency and Proportion of Fraudulent Tallies by State",
       fill = "Number of\nFraudulent Tallies") +
  scale_y_continuous(labels = scales::percent_format(scale = 1)) +
  scale_fill_gradient(low = "lightblue", high = "red")
```



Task 3. Clean vote return data (3pt)

Your next task is to clean a different dataset from the researchers' replication dossier. Its path is `data/Mexican_Election_Fraud/dataverse/VoteReturns.csv`. This dataset contains information about vote returns recorded in every tally sheet. This dataset is essential for the replication of Figure 4 in the research article.

Task 3.1. Load vote return data

Load the dataset onto your R environment. Name this dataset `d_return`. Show summary statistics of this dataset and describe the takeaways using a few sentences.

```r
d_return <- read_csv("data/VoteReturns.csv")
```

```r
# for a quick summary of the variables in the data set, "summary" is perfect:
summary(d_return)
```

```
##      foto             seccion            casilla              dtto
##  Length:53499       Length:53499       Length:53499       Length:53499
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##       dto           municipio             edo              entidad
##  Min.   :  1.000   Length:53499       Length:53499       Length:53499
##  1st Qu.:  3.000   Class :character   Class :character   Class :character
##  Median :  6.000   Mode  :character   Mode  :character   Mode  :character
##  Mean   :  8.704
##  3rd Qu.: 10.000
##  Max.   :341.000
##  NA's   :4
##      pagina           p1                 p2                p3
##  Min.   :   1   Min.   :     0.0   Min.   :    0.0   Min.   :   0.0
##  1st Qu.:  45   1st Qu.:   250.0   1st Qu.:   67.0   1st Qu.:  98.0
##  Median :  92   Median :   530.0   Median :  245.0   Median : 233.0
##  Mean   : 104   Mean   :   671.9   Mean   :  343.3   Mean   : 319.3
##  3rd Qu.: 146   3rd Qu.:   941.5   3rd Qu.:  482.0   3rd Qu.: 442.0
##  Max.   :2020   Max.   :364105.0   Max.   :48225.0   Max.   :9127.0
##  NA's   :39                                          NA's   :1
##        p4                p5               pan               pri
##  Min.   :    0.0   Min.   :   0.00   Min.   :   0.00   Min.   :   0.0
##  1st Qu.:   73.0   1st Qu.:   0.00   1st Qu.:   2.00   1st Qu.:  52.0
##  Median :  222.0   Median :  13.00   Median :  18.00   Median : 107.0
##  Mean   :  369.7   Mean   :  29.36   Mean   :  56.88   Mean   : 162.7
##  3rd Qu.:  464.0   3rd Qu.:  36.00   3rd Qu.:  72.00   3rd Qu.: 195.0
##  Max.   :21265.0   Max.   :6650.00   Max.   :4436.00   Max.   :6080.0
##
##       pps               psm                pms              pfcrn
##  Min.   :   0.00   Min.   :   0.000   Min.   :   0.00   Min.   :   0.00
##  1st Qu.:   0.00   1st Qu.:   0.000   1st Qu.:   0.00   1st Qu.:   0.00
##  Median :   9.00   Median :   1.000   Median :   2.00   Median :  11.00
##  Mean   :  35.04   Mean   :   3.637   Mean   :  12.19   Mean   :  34.17
##  3rd Qu.:  47.00   3rd Qu.:   3.000   3rd Qu.:  13.00   3rd Qu.:  45.00
##  Max.   :1056.00   Max.   :1802.000   Max.   :5511.00   Max.   :1011.00
##
##       prt               parm            noregis           nombrenore
##  Min.   :  0.000   Min.   :   0.00   Min.   :   0.0000   Length:53499
##  1st Qu.:  0.000   1st Qu.:   0.00   1st Qu.:   0.0000   Class :character
##  Median :  0.000   Median :   5.00   Median :   0.0000   Mode  :character
##  Mean   :  1.912   Mean   :  20.44   Mean   :   0.8175
##  3rd Qu.:  1.000   3rd Qu.:  23.00   3rd Qu.:   0.0000
##  Max.   :592.000   Max.   :1170.00   Max.   :1604.0000
##                                      NA's   :1
##      otros           otroscan              pan2               pri2
##  Min.   :   0.00   Length:53499       Min.   :   0.000   Min.   :   0.00
##  1st Qu.:   0.00   Class :character   1st Qu.:   0.000   1st Qu.:   0.00
##  Median :   0.00   Mode  :character   Median :   0.000   Median :   0.00
##  Mean   :   3.17                      Mean   :   1.475   Mean   :   3.94
##  3rd Qu.:   0.00                      3rd Qu.:   0.000   3rd Qu.:   0.00
##  Max.   :1734.00                      Max.   :1239.000   Max.   :2651.00
##  NA's   :4
##       pps2               psm2              pms2              pfcrn2
##  Min.   :  0.0000   Min.   :  0.000   Min.   :  0.0000   Min.   :   0.0000
##  1st Qu.:  0.0000   1st Qu.:  0.000   1st Qu.:  0.0000   1st Qu.:   0.0000
##  Median :  0.0000   Median :  0.000   Median :  0.0000   Median :   0.0000
##  Mean   :  0.7557   Mean   :  0.116   Mean   :  0.3039   Mean   :   0.7968
##  3rd Qu.:  0.0000   3rd Qu.:  0.000   3rd Qu.:  0.0000   3rd Qu.:   0.0000
##  Max.   :680.0000   Max.   :429.000   Max.   :427.0000   Max.   :1319.0000
##
##       prt2             parm2             noregis2           otro2
##  Min.   :  0.000   Min.   :  0.0000   Min.   :  0.00000   Min.   : 0.000000
##  1st Qu.:  0.000   1st Qu.:  0.0000   1st Qu.:  0.00000   1st Qu.: 0.000000
##  Median :  0.000   Median :  0.0000   Median :  0.00000   Median : 0.000000
##  Mean   :  0.073   Mean   :  0.5122   Mean   :  0.01837   Mean   : 0.002935
##  3rd Qu.:  0.000   3rd Qu.:  0.0000   3rd Qu.:  0.00000   3rd Qu.: 0.000000
##  Max.   :429.000   Max.   :429.0000   Max.   :259.00000   Max.   :26.000000
##
##       pan3              pri3             pps3             psm3
##  Min.   :   0.00   Min.   :   0.0   Min.   :  0.00   Min.   :  0.000
##  1st Qu.:   0.00   1st Qu.:   0.0   1st Qu.:  0.00   1st Qu.:  0.000
##  Median :   0.00   Median :  32.0   Median :  0.00   Median :  0.000
##  Mean   :  39.36   Mean   :  93.5   Mean   : 22.08   Mean   :  2.094
##  3rd Qu.:  45.00   3rd Qu.: 127.0   3rd Qu.: 21.00   3rd Qu.:  1.000
##  Max.   :2194.00   Max.   :6080.0   Max.   :921.00   Max.   :856.000
##                    NA's   :1                         NA's   :2
##       pms3              pfcrn3            prt3             parm3
##  Min.   :   0.000   Min.   :  0.00   Min.   :  0.000   Min.   :   0.00
##  1st Qu.:   0.000   1st Qu.:  0.00   1st Qu.:  0.000   1st Qu.:   0.00
##  Median :   0.000   Median :  0.00   Median :  0.000   Median :   0.00
##  Mean   :   7.803   Mean   : 21.63   Mean   :  1.077   Mean   :  12.68
##  3rd Qu.:   5.000   3rd Qu.: 23.00   3rd Qu.:  1.000   3rd Qu.:  11.00
##  Max.   :8932.000   Max.   :992.00   Max.   :413.000   Max.   :1170.00
##  NA's   :1          NA's   :1
##     noregis3            otro3                suma            nulos
##  Min.   :  0.0000   Min.   :   0.0000   Min.   :   0.0   Min.   :   0.00
##  1st Qu.:  0.0000   1st Qu.:   0.0000   1st Qu.:  82.0   1st Qu.:   0.00
##  Median :  0.0000   Median :   0.0000   Median : 217.0   Median :   3.00
##  Mean   :  0.3498   Mean   :   0.3016   Mean   : 296.4   Mean   :  21.93
##  3rd Qu.:  0.0000   3rd Qu.:   0.0000   3rd Qu.: 420.0   3rd Qu.:  11.00
##  Max.   :747.0000   Max.   :1353.0000   Max.   :9962.0   Max.   :8770.00
##                     NA’s   :1           NA’s   :1        NA’s   :1
##      total             suma1             nulos1             total1
##  Min.   :    0.0   Min.   :   0.000   Min.   :   0.000   Min.   :   0.000
##  1st Qu.:   90.0   1st Qu.:   0.000   1st Qu.:   0.000   1st Qu.:   0.000
##  Median :  229.0   Median :   0.000   Median :   0.000   Median :   0.000
##  Mean   :  315.7   Mean   :   4.865   Mean   :   0.635   Mean   :   7.175
##  3rd Qu.:  440.0   3rd Qu.:   0.000   3rd Qu.:   0.000   3rd Qu.:   0.000
##  Max.   :16811.0   Max.   :3333.000   Max.   :1600.000   Max.   :2787.000
##  NA’s   :1         NA’s   :2          NA’s   :2          NA’s   :2
##      suma2            nulos2             total2         inciden
##  Min.   :   0.0   Min.   :   0.00   Min.   :   0.0   Length:53499
##  1st Qu.:   0.0   1st Qu.:   0.00   1st Qu.:   0.0   Class :character
##  Median :   0.0   Median :   0.00   Median :   0.0   Mode  :character
##  Mean   : 176.9   Mean   :  11.38   Mean   : 192.6
##  3rd Qu.: 280.0   3rd Qu.:   5.00   3rd Qu.: 299.0
##  Max.   :7633.0   Max.   :7734.00   Max.   :9855.0
##  NA’s   :2        NA’s   :2         NA’s   :2
##  representante_pan  representante_pri  representante_pps  representante_pms
##  Length:53499       Length:53499       Length:53499       Length:53499
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  representante_psm  representante_pfcrn representante_prt  representante_parm
##  Length:53499       Length:53499        Length:53499       Length:53499
##  Class :character   Class :character    Class :character   Class :character
##  Mode  :character   Mode  :character    Mode  :character   Mode  :character
##
##
##
##
##  protesta_pan       protesta_pri       protesta_pps       protesta_pms
##  Length:53499       Length:53499       Length:53499       Length:53499
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  protesta_psm       protesta_pfcrn     protesta_prt       protesta_parm
##  Length:53499       Length:53499       Length:53499       Length:53499
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

```
##  protesta_otro       presidente         secretario           primer
##  Length:53499       Length:53499       Length:53499       Length:53499
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    segundo            observa              var79           salinas
##  Length:53499       Length:53499       Min.   :   1.0   Min.   :   0.0
##  Class :character   Class :character   1st Qu.:   1.0   1st Qu.:  63.0
##  Mode  :character   Mode  :character   Median :   1.0   Median : 115.0
##                                        Mean   : 131.2   Mean   : 174.4
##                                        3rd Qu.:   2.0   3rd Qu.: 206.0
##                                        Max.   :9999.0   Max.   :6080.0
##                                        NA's   :53422
##    clouthier           ibarra           castillo        ppsccs
##  Min.   :   0.00   Min.   :  0.000   Min.   :   0   Min.   :   0.00
##  1st Qu.:   3.00   1st Qu.:  0.000   1st Qu.:   0   1st Qu.:   1.00
##  Median :  23.00   Median :  0.000   Median :   1   Median :  12.00
##  Mean   :  61.37   Mean   :  2.185   Mean   :   4   Mean   :  37.67
##  3rd Qu.:  78.00   3rd Qu.:  2.000   3rd Qu.:   3   3rd Qu.:  51.00
##  Max.   :4436.00   Max.   :592.000   Max.   :1802   Max.   :1056.00
##
##     pfcrnccs          parmccs            nrccs             noregccs
##  Min.   :   0.00   Min.   :   0.00   Min.   :0.000000   Min.   :   0.0000
##  1st Qu.:   1.00   1st Qu.:   0.00   1st Qu.:0.000000   1st Qu.:   0.0000
##  Median :  14.00   Median :   6.00   Median :0.000000   Median :   0.0000
##  Mean   :  36.85   Mean   :  21.98   Mean   :0.006654   Mean   :   0.1439
##  3rd Qu.:  48.00   3rd Qu.:  25.00   3rd Qu.:0.000000   3rd Qu.:   0.0000
##  Max.   :1319.00   Max.   :1170.00   Max.   :1.000000   Max.   :1125.0000
##
##       occs           otrosccs            cardenas
##  Min.   :0.0000   Min.   :   0.000   Min.   :   0.00
##  1st Qu.:1.0000   1st Qu.:   0.000   1st Qu.:  10.00
##  Median :1.0000   Median :   0.000   Median :  53.00
##  Mean   :0.9942   Mean   :   3.106   Mean   :  99.75
##  3rd Qu.:1.0000   3rd Qu.:   0.000   3rd Qu.: 141.00
##  Max.   :1.0000   Max.   :1734.000   Max.   :2280.00
##
```

```r
# to have an (very broad) idea of the extent to which we have missing values
# (NA), this scripte works quite well:
sum(is.na(d_return))
```

```
## [1] 257601
```

Note 2. What are in this dataset?

This table contains a lot of different variables. The researcher offers no comprehensive documentation to tell us what every column means. For the sake of this problem set, you only need to know the meanings of the following columns:

- `foto` is an identifier of the images of tally sheets in this dataset. We will need it to merge this dataset with the `d_tally` data.
- `edo` contains the names of states.
- `dto` contains the names of districts (in Arabic numbers).
- `salinas`, `clouthier`, and `ibarra` contain the counts of votes (as recorded in the tally sheets) for presidential candidates Salinas (PRI), Cardenas (FDN), and Clouthier (PAN). In addition, the summation of all three makes the total number of **presidential votes**.
- `total` contains the total number of **legislative votes**.

Task 3.2. Recode names of states

A state whose name is `Chihuahua` is mislabelled as `Chihuhua`. A state whose name is currently `Edomex` needs to be recoded to `Estado de Mexico`. Please re-code the names of these two states accordingly.

```r
d_return <- d_return |>
  mutate(ifelse(edo == "Chihuahua","Chihuhua",edo),
         ifelse(edo == "Edomex","Estado de Mexico",edo))
```

Task 3.3. Recode districts' identifiers

Compare how districts' identifiers are recorded differently in the tally (`d_tally`) from vote return (`d_return`) datasets. Specifically, in the `d_tally` dataset, `district` contains Roman numbers while in the `d_return` dataset, `dto` contains Arabic numbers. Recode districts' identifiers in the `d_return` dataset to match those in the `d_tally` dataset. To complete this task, first summarize the values of the two district identifier columns in the two datasets respectively to verify the above claim. Then do the requested conversion.

```r
tally_dis_summary <- d_tally |>
  group_by(district) |>
  summarise(count = n()) |>
  arrange(district)
return_dis_summary <- d_return |>
  group_by(dto) |>
  summarise(count = n()) |>
  arrange(dto)
# comparing the two, we notice a few differences in the number of districts and
# the data for each district between the two datasets

# some cleaning:
rm(tally_dis_summary,return_dis_summary)

d_return <- d_return |>
  mutate(dto = as.roman(dto))
```

Task 3.4. Create a name_image identifier for the d_return dataset

In the d_return dataset, create a column named name_image as the first column. The column concatenate values in the three columns: edo, dto, and foto with an underscore _ as separators.

```
d_return <- d_return |>
  mutate(name_image = stringr::str_c(edo, dto, foto, sep = "_"))
```

Task 3.5. Wrangle the name_image column in two datasets

As a final step before merging d_return and d_tally, you are required to perform the following data wrangling. For the name_image column in BOTH d_return and d_tally:

- Convert all characters to lower case.
- Remove ending substring .jpg.

```r
d_return <- d_return |>
  mutate(name_image = str_to_lower(name_image)) |>
  mutate(name_image = str_replace(name_image, "\\.jpg$", ""))

d_tally <- d_tally |>
  mutate(name_image = str_to_lower(name_image)) |>
  mutate(name_image = str_replace(name_image, "\\.jpg$", ""))
```

Task 3.6 Join classification results and vote returns

After you have successfully completed all the previous steps, join `d_return` and `d_tally` by column `name_image`. This task contains two part. First, use appropriate `tidyverse` functions to answer the following questions:

- How many rows are in `d_return` but not in `d_tally`? Which states and districts are they from?
- How many rows are in `d_tally` but not in `d_return`? Which states and districts are they from?

```r
# First, rows in `d_return` but not in `d_tally`:
ex_d_return <- anti_join(d_return, d_tally, by = "name_image")
num_ex_d_return <- nrow(ex_d_return)
summary_exclusive_d_return <- ex_d_return |>
  group_by(edo, dto) |>
  summarise(count = n()) |>
  ungroup()

# Second, rows in `d_tally` but not in `d_return`:
ex_d_tally <- anti_join(d_tally, d_return, by = "name_image")
num_ex_d_return <- nrow(ex_d_tally)
summary_ex_d_tally <- ex_d_tally |>
  group_by(state, district) |>
  summarise(count = n()) |>
  ungroup()
```

Second, create a dataset call `d` by joining `d_return` and `d_tally` by column `name_image`. `d` contains rows whose identifiers appear in *both* datasets and columns from *both* datasets.

```r
d <- inner_join(d_return, d_tally, by = "name_image")
```

Task 4. Visualize distributions of fraudulent tallies across candidates (6pt)

In this task, you will visualize the distributions of fraudulent tally sheets across three presidential candidates: **Sarinas (PRI)**, **Cardenas (FDN)**, and **Clouthier (PAN)**. The desired output of is reproducing and extending Figure 4 in the research article (Cantu 2019, pp. 720).

Task 4.1. Calculate vote proportions of Salinas, Clouthier, and Cardenas

Before getting to the visualization, you should first calculate the proportion of votes (among all) received by the three candidates of interest. As additional background information, there are two more presidential candidates in this election, whose votes received are recorded in `ibarra` and `castillo` respectively. Please perform the tasks in the following two steps on the `d` dataset:

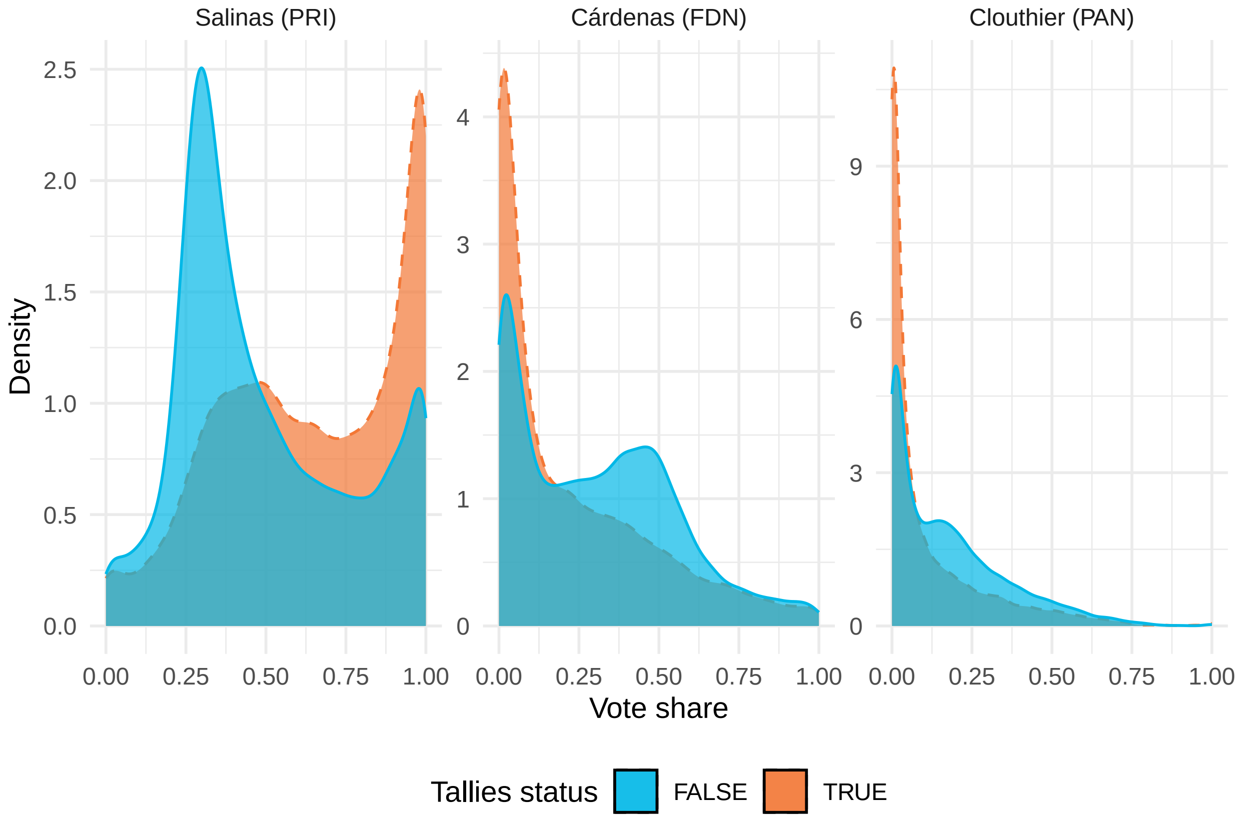
- Create a new column named `total_president` as an indicator of the total number of votes of the 5 presidential candidates.
- Create three columns `salinas_prop`, `cardenas_prop`, and `clouthier_prop` that indicate the proportions of the votes these three candidates receive respectively.

```r
d <- d |>
  mutate(total_president = salinas + cardenas + clouthier + ibarra + castillo,
    salinas_prop = salinas / total_president,
    cardenas_prop = cardenas / total_president,
    clouthier_prop = clouthier / total_president)
```

Task 4.2. Replicate Figure 4

Based on all the previous step, reproduce Figure 4 in Cantu (2019, pp. 720).

```r
l_data <- d |>
  pivot_longer(
    cols = ends_with("_prop"),
    names_to = "candidate",
    values_to = "vote_share_prop"
  ) |>
  mutate(
    candidate = str_remove(candidate, "_prop"),
    # Update the factor levels with new labels
    candidate = factor(candidate, levels = c("salinas", "cardenas", "clouthier"),
                       labels = c("Salinas (PRI)", "Cárdenas (FDN)", "Clouthier (PAN)")),
    fraud_bin = factor(fraud_bin, levels = c(TRUE, FALSE))
  ) |>
  filter(!is.na(vote_share_prop))

fig_4 <- ggplot(l_data, aes(x = vote_share_prop, fill = fraud_bin, color = fraud_bin)) +
  geom_density(data = filter(l_data, fraud_bin == TRUE), alpha = 0.7,
               linetype = "dashed", na.rm = TRUE) +
  geom_density(data = filter(l_data, fraud_bin == FALSE), alpha = 0.7,
               linetype = "solid", na.rm = TRUE) +
  scale_fill_manual(values = c("TRUE" = "#f37735", "FALSE" = "#00B8E7")) +
  scale_color_manual(values = c("TRUE" = "#f37735", "FALSE" = "#00B8E7")) +
  facet_wrap(~candidate, scales = "free", nrow = 1) +
  theme_minimal() +
  labs(x = "Vote share", y = "Density", fill = "Fraud Status") +
  theme(legend.position = "bottom") +
  guides(fill = guide_legend(title = "Tallies status"), color = FALSE)

plot(fig_4)
```



```
# the values for the color came from Google images as well as trial and error
```

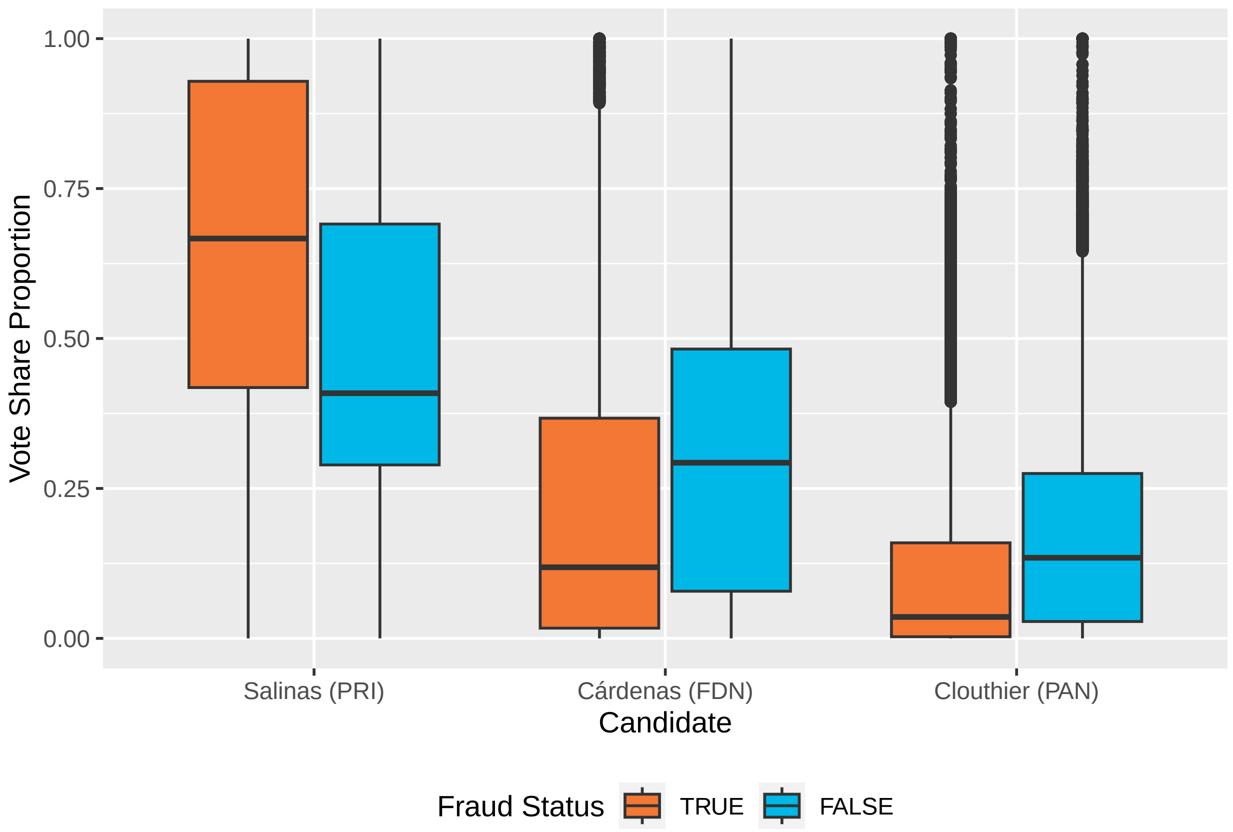
Note: Your performance in this task will be mainly evaluated based on your output's similarity with the original figure. Pay attention to the details. For your reference, below is a version created by the instructor.

Task 4.3. Discuss and extend the reproduced figure

Referring to your reproduced figures and the research articles, in what way is the researcher's argument supported by this figure? Make an alternative visualization design that can substantiate and even augment the current argument. After you have shown your alternative design, in a few sentences, describe how your design provides visual aid as effectively as or more effectively than the original figure.

Note: Feel free to make *multiple* alternative designs to earn bonus credits. However, please be selective. Only a design with major differences from the existing ones can be counted as an alternative design.

```r
# The researcher argues that by his empirical methodology, he has found clear
# evidence for electoral fraud. In this figure, the top plot showing the
# distribution of altered and not altered tallies for Salinas reveals the most
# on potential electoral fraud. As opposed to the other two, alterations are
# almost unanimously supporting this candidate. More interestingly even, is the
# large difference in the distributions of altered vs not altered tallies for
# this candidate, as opposed to other candidates where there distributions
# almost coincide: this makes election fraud much likelier in favor of this
# candidate.

l_data <- d |>
  pivot_longer(
    cols = ends_with("_prop"),
    names_to = "candidate",
    values_to = "vote_share_prop"
  ) |>
  mutate(
    candidate = str_remove(candidate, "_prop"),
    candidate = factor(candidate, levels = c("salinas", "cardenas", "clouthier"),
                       labels = c("Salinas (PRI)", "Cárdenas (FDN)", "Clouthier (PAN)")),
    fraud_bin = factor(fraud_bin, levels = c(TRUE, FALSE))
  ) |>
  filter(!is.na(vote_share_prop))

boxplot <- ggplot(l_data, aes(x = candidate, y = vote_share_prop, fill = fraud_bin)) +
  geom_boxplot(size = 0.5) +
  scale_fill_manual(values = c("TRUE" = "#f37735", "FALSE" = "#00B8E7")) +
  labs(x = "Candidate", y = "Vote Share Proportion", fill = "Fraud Status") +
  theme(legend.position = "bottom")

plot(boxplot)
```



```
# I attempted to fix some issues such as the thick lines appearing
# inconsistently using ChatGPT with no success.
```

Note: Feel free to suggest *multiple* alternative designs to earn bonus credits. However, please be selective. Only a design with major differences from the existing ones can be counted as an alternative design.

Task 5. Visualize the discrepancies between presidential and legislative Votes (6pt)

In this task, you will visualize the differences between the number of presidential votes across tallies. The desired output of is reproducing and extending Figure 5 in the research article (Cantu 2019, pp. 720).

Task 5.1. Get district-level discrepancies and fraud data

As you might have noticed in the caption of Figure 5 in Cantu (2019, pp. 720), the visualized data are aggregated to the *district* level. In contrast, the unit of analysis in the dataset we are working with, `d`, is *tally*. As a result, the first step of this task is to aggregate the data. Specifically, please aggregate `d` into a new data frame named `sum_fraud_by_district`, which contains the following columns:

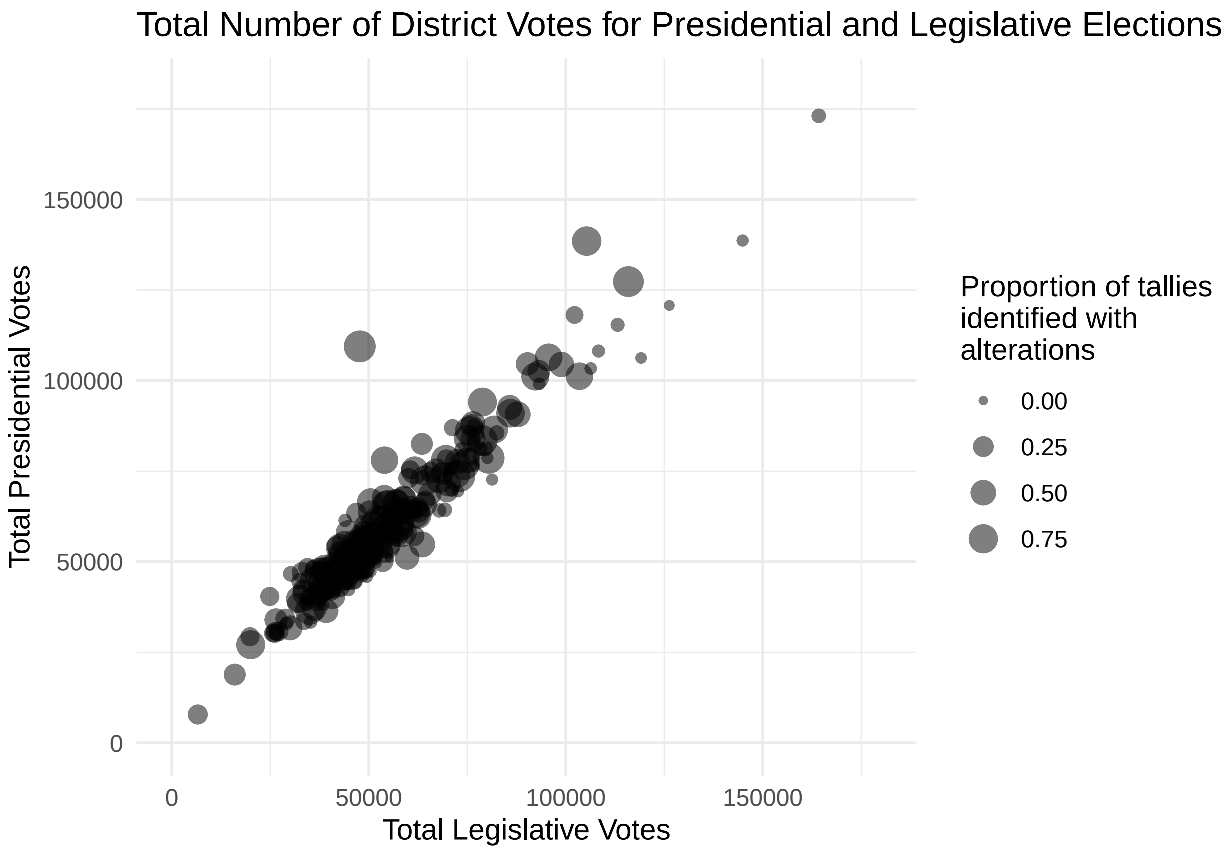
- `state`: Names of states
- `district`: Names of districts
- `vote_president`: Total numbers of presidential votes
- `vote_legislature`: Total numbers of legislative votes
- `vote_diff`: Total number of presidential votes minus total number of legislative votes
- `prop_fraud`: Proportions of fraudulent tallies (hint: using `fraud_bin`)

```r
sum_fraud_by_state <- d |>
  group_by(state, district) |>
  summarise(
    vote_president = sum(total_president, na.rm = TRUE),
    vote_legislature = sum(total, na.rm = TRUE),
    vote_diff = sum(vote_president, na.rm = TRUE) - sum(vote_legislature, na.rm = TRUE),
    prop_fraud = mean(fraud_bin, na.rm = TRUE),
  .groups = "drop"
  )
# Not so intuitive replacement of *ungroup()* by *.groups = "drop"*, help found
# on stack overflow:
# *https://stackoverflow.com/questions/62140483/how-to-interpret-dplyr-
# message-summarise-regrouping-output-by-x-override*
```

Task 5.2. Replicate Figure 5

Based on all the previous step, reproduce Figure 5 in Cantu (2019, pp. 720).

```r
sum_fraud_by_state <- sum_fraud_by_state |>
  mutate(fraud_quartiles = cut(prop_fraud, breaks = c(0, 0.25, 0.5, 0.75, 1),
  labels = c(0, 0.25, 0.5, 0.75)),prop_fraud_numeric = as.numeric(as.character(prop_fraud)))

fig_5 <- ggplot(sum_fraud_by_state, aes(x = vote_legislature, y = vote_president, size = prop_fraud_nume
  geom_point(alpha = 0.5) +
  scale_size_continuous(range = c(1, 5),
  name = "Proportion of tallies\nidentified with\nalterations") +
  scale_x_continuous(limits = c(0, 180000)) +
  scale_y_continuous(limits = c(0, 180000)) +
  labs(x = "Total Legislative Votes",
  y = "Total Presidential Votes",
  title = "Total Number of District Votes for Presidential and Legislative Elections") +
  theme_minimal()

plot(fig_5)
```



Note 1: Your performance in this task will be mainly evaluated based on your output's similarity with the original figure. Pay attention to the details.

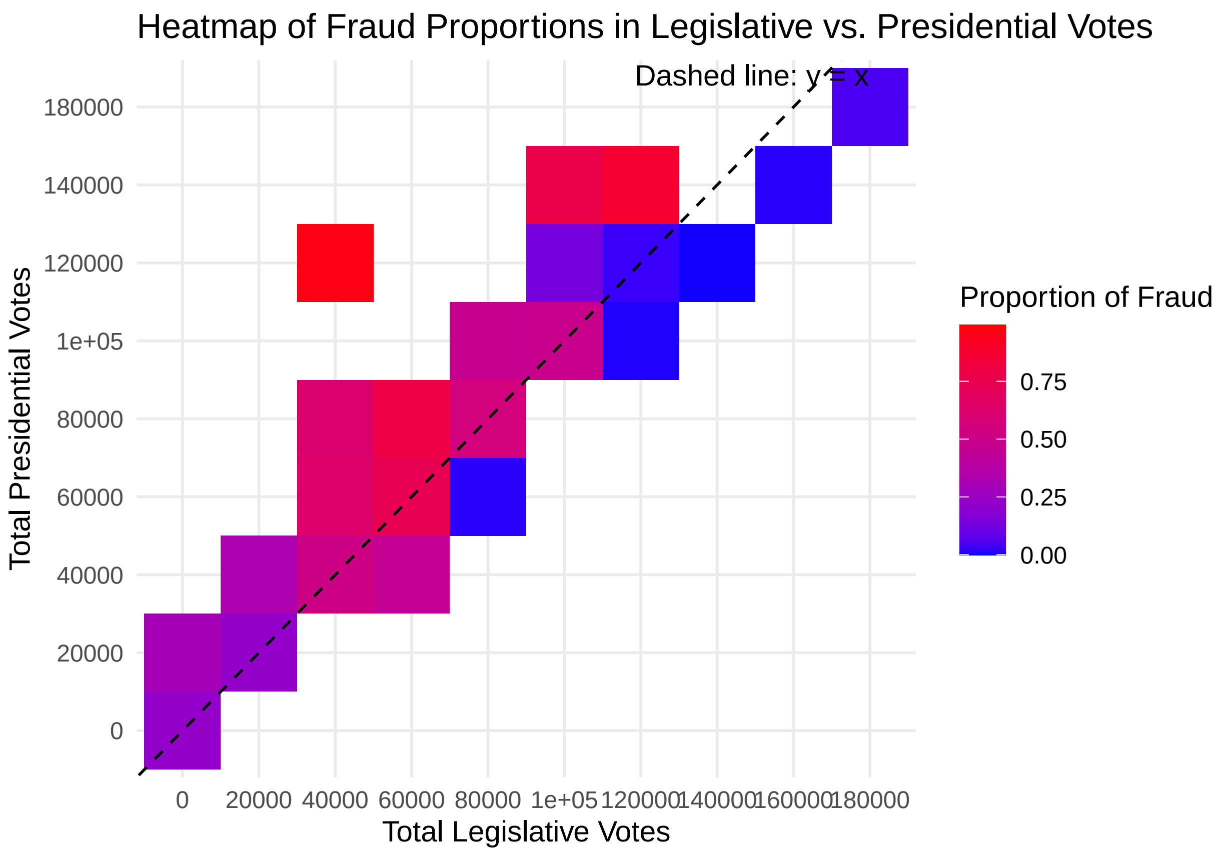
Note 2: The instructor has detected some differences between the above figure with Figure 5 on the published article. Please use the instructor's version as your main benchmark.

Task 5.3. Discuss and extend the reproduced figure

Referring to your reproduced figures and the research articles, in what way is the researcher's argument supported by this figure? Make an alternative visualization design that can substantiate and even augment the current argument. After you have shown your alternative design, in a few sentences, describe how your design provides visual aid as effectively as or more effectively than the original figure.

Note: Feel free to make *multiple* alternative designs to earn bonus credits. However, please be selective. Only a design with major differences from the existing ones can be counted as an alternative design.

```
# According to the author of the paper, both presidential and legislative
# ballots were given to every person voting. Yet we can see that some districts
# (bubbles) with very important proportions of tally alteration have more
# presidential votes than legislative votes. One district in particular is very
# suspicious, with its presidential votes more than double its legislative
# votes, and the proportion of its tally alteration >75%. All in all, this graph
# highlights well the authors strong suspicion for election fraud.

# Alternative design: Heat map
# Define bounds and labels for axes
breaks_leg_pres <- seq(0, 180000, length.out = 11)
labels_leg_pres <- seq(0, 180000, by = 20000)

# Turning the continuous variables into categorical ones, to have data by
# interval (use of ChatGPT to for the syntax helping me segmenting the data in
# such a way). This Heat map is great because it simplifies Figure 5, the
# audience can quite clearly see there is more fraud (red) above the dashed line
# than under, meaning (at least proportionally) more fraud was done in districts
# where there were more presidential votes than legislative votes. This is no
# proof for election fraud, but is very suspicious and reinforces well the
# authors point(s).
factor_breaks_leg_pres <- cut(breaks_leg_pres, breaks = breaks_leg_pres, include.lowest = TRUE,
  labels = labels_leg_pres)
sum_fraud_by_state$vote_legislature_binned <- cut(sum_fraud_by_state$vote_legislature,
  breaks = breaks_leg_pres, labels = labels_leg_pres)
sum_fraud_by_state$vote_president_binned <- cut(sum_fraud_by_state$vote_president,
  breaks = breaks_leg_pres, labels = labels_leg_pres)

# Building the actual Heat map
fig_heatmap <- ggplot(sum_fraud_by_state, aes(x = vote_legislature_binned,
  y = vote_president_binned, fill = prop_fraud_numeric)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red", name = "Proportion of Fraud") +
  geom_abline(intercept = 0, slope = 1, color = "black", linetype = "dashed") +
  scale_x_discrete(breaks = labels_leg_pres, labels = labels_leg_pres) +
  scale_y_discrete(breaks = labels_leg_pres, labels = labels_leg_pres) +
  labs(x = "Total Legislative Votes",
       y = "Total Presidential Votes",
       title = "Heatmap of Fraud Proportions in Legislative vs. Presidential Votes") +
  theme_minimal() +   annotate("text", x = Inf, y = Inf, label = "Dashed line: y = x",
  hjust = 1.2, vjust = 1.2)

print(fig_heatmap)
```


Heatmap of Fraud Proportions in Legislative vs. Presidential Votes
Dashed line: y = x
Total Presidential Votes
180000
140000
120000
1e+05
80000
60000
40000
20000
0
Total Legislative Votes
0
20000
40000
60000
80000
1e+05
120000
140000
160000
180000
Proportion of Fraud
0.75
0.50
0.25
0.00

Task 6. Visualize the spatial distribution of fraud (6pt)

In this final task, you will visualize the spatial distribution of electoral fraud in Mexico. The desired output of is reproducing and extending Figure 3 in the research article (Cantu 2019, pp. 720).

Note 3. Load map data

As you may recall, map data can be stored and shared in **two** ways. The simpler format is a table where each row has information of a point that "carves" the boundary of a geographic unit (a Mexican state in our case). In this type of map data, a geographic unit is is represented by multiple rows. Alternatively, a map can be represented by a more complicated and more powerful format, where each geographic unit (a Mexican state in our case) is represented by an element of a `geometry` column. For this task, I provide you with a state-level map of Mexico represented by both formats respectively.

Below the instructor provide you with the code to load the maps stored under the two formats respectively. Please run them before starting to work on your task.

```r
# IMPORTANT: Remove eval=FALSE above when you start this part!

# Load map (simple)
map_mex <- read_csv("data/map_mexico/map_mexico.csv")
# Load map (sf): You need to install and load library "sf" in advance
library(sf)
map_mex_sf <- st_read("data/map_mexico/shapefile/gadm36_MEX_1.shp")
map_mex_sf <- st_simplify(map_mex_sf, dTolerance = 100)
```

Bonus question: Explain the operations on `map_mex_sf` in the instructor's code above.

– As the name sugests, `st_simplify` is a function simplifying a map, here `map_mex_sf`. The second argument, `dTolerance` defines the extent to which the map is simplified. Here the value is 100, a larger value means simpler and conversely a smaller value closer to the orginal map. –

Note: The map (sf) data we use are from https://gadm.org/download_country_v3.html.

Task 6.1. Reproduce Figure 3 with `map_mex`

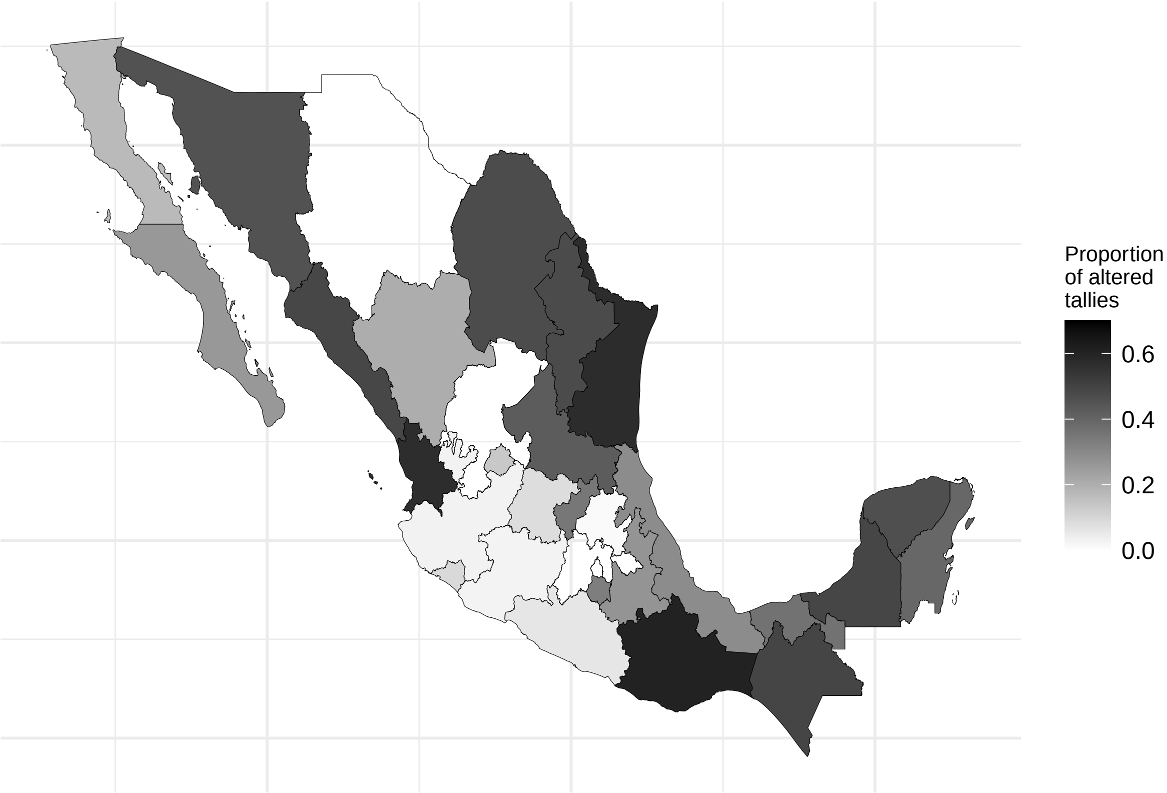
In this task, you are required to reproduce Figure 3 with the `map_mex` data.

Note:

- Your performance in this task will be mainly evaluated based on your output's similarity with the original figure. Pay attention to the details. For your reference, below is a version created by the instructor.
- Hint: Check the states' names in the map data and the electoral fraud data. Recode them if necessary.

```r
sum_fraud_by_state <- sum_fraud_by_state |>
  mutate(
    state = case_when(
      state == "Michoacan" ~ "Michoacán",
      state == "Queretaro" ~ "Querétaro",
      state == "Yucatan" ~ "Yucatán",
      state == "Mexico" ~ "México",
      state == "Nuevo Leon" ~ "Nuevo León",
      state == "San Luis Potosi" ~ "San Luis Potosí",
      TRUE ~ state
    )
  )
map_mex <- map_mex |>
  mutate(state = state_name)

map_data <- map_mex |>
  left_join(sum_fraud_by_state, by = "state")

fig_3.1 <- ggplot(data = map_data, aes(x = long, y = lat, group = group,
  fill = prop_fraud)) +
  geom_polygon(color = "black", size = 0.1) +
  scale_fill_gradient(low = "white", high = "black", na.value = NA,
  limits = c(0, 0.7),
    guide = "colourbar",name = "Proportion\nof altered\ntallies") +
  coord_fixed(1.3) + labs(fill = "Prop. Fraud",
  title = "Figure 3.1: Rates of Tallies Classified as Altered by State") +
  theme_minimal() +
  theme_minimal() + theme(axis.text.x = element_blank(),
      axis.text.y = element_blank(),
      axis.title.x = element_blank(),
      axis.title.y = element_blank(),
      plot.title = element_text(size = 9),
      legend.title = element_text(size = 8))

plot(fig_3.1)
```

Figure 3.1: Rates of Tallies Classified as Altered by State



Task 6.2. Reproduce Figure 3 with `map_mex_sf`

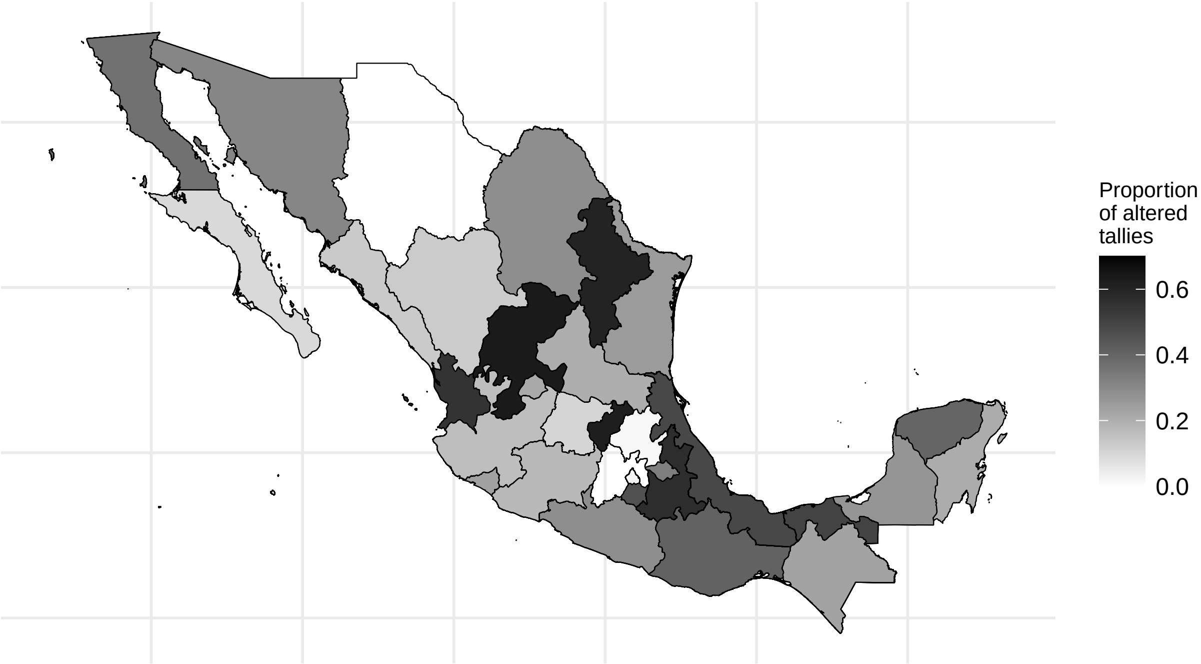
In this task, you are required to reproduce Figure 3 with the `map_mex` data.

Note:

- Your performance in this task will be mainly evaluated based on your output's similarity with the original figure. Pay attention to the details. For your reference, below is a version created by the instructor.
- Hint: Check the states' names in the map data and the electoral fraud data. Recode them if necessary.

```r
map_data_sf <- map_mex_sf |>
  left_join(sum_fraud_by_state, by = c("NAME_1" = "state"))

fig_3.2_sf <- ggplot(data = map_data_sf) +
  geom_sf(aes(fill = prop_fraud), color = "black", size = 0.1) +
  scale_fill_gradient(low = "white", high = "black", na.value = NA, limits = c(0, 0.7),
    guide = "colourbar", name = "Proportion\nof altered\ntallies") +
  labs(title = "Figure 3.2: Rates of Tallies Classified as Altered by State (sf)") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        plot.title = element_text(size = 10),
        legend.title = element_text(size = 8))

plot(fig_3.2_sf)
```

Figure 3.2: Rates of Tallies Classified as Altered by State (sf)



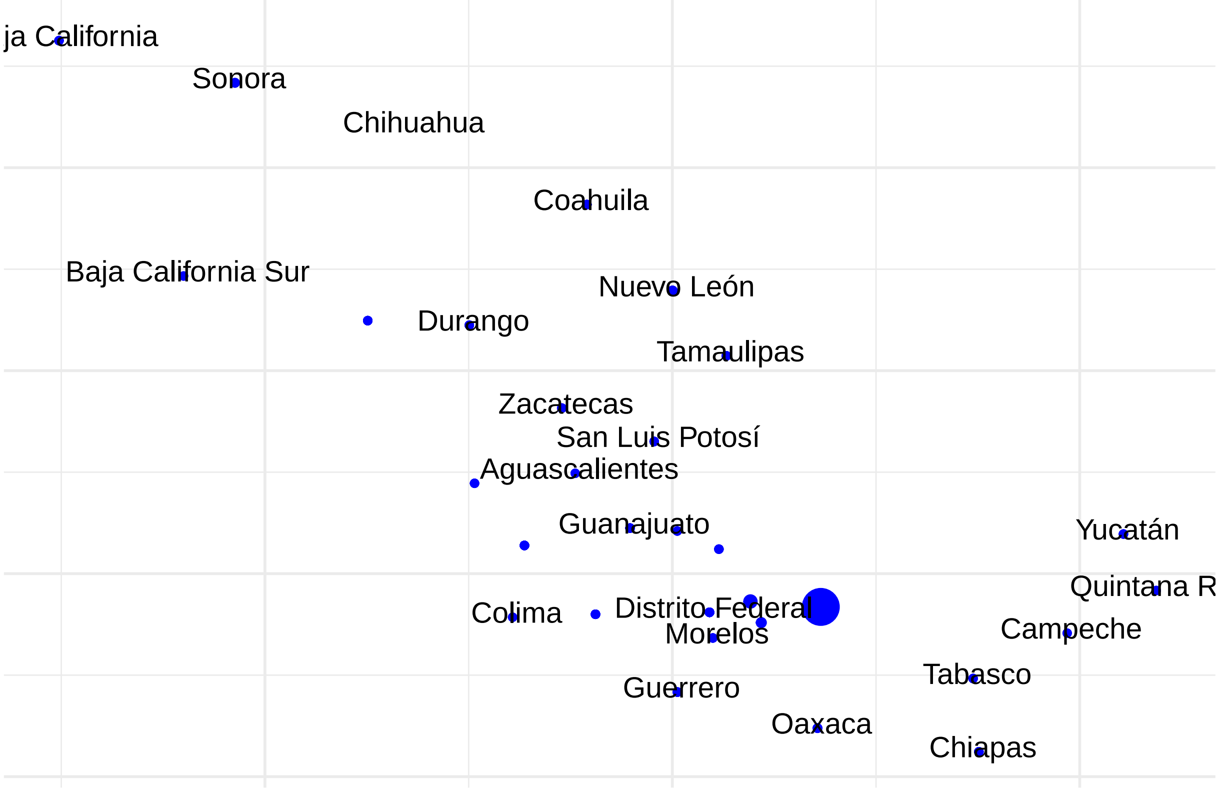
Task 6.3. Discuss and extend the reproduced figures

Referring to your reproduced figures and the research articles, in what way is the researcher's argument supported by this figure? Make an alternative visualization design that can substantiate and even augment the current argument. After you have shown your alternative design, in a few sentences, describe how your design provides visual aid as effectively as or more effectively than the original figure.

Note: Feel free to make *multiple* alternative designs to earn bonus credits. However, please be selective. Only a design with major differences from the existing ones can be counted as an alternative design.

```
# Both Figure 3.1 and 3.2 help the audience visualize which part of Mexico were
# most affected by fraudulent tallies, and to what extent. The viewer can
# observe important differences between states in terms of tally fraud, which
# supports the claim of the author that some states were more prone to fraud
# than others, potentially due to their political influence or their leader's.

map_data_sf <- map_mex_sf |>
  left_join(sum_fraud_by_state, by = c("NAME_1" = "state"))

# Calculate centroids of each circle
centroids <- st_centroid(map_data_sf$geometry)
map_data_sf$lon <- st_coordinates(centroids)[, 1]
map_data_sf$lat <- st_coordinates(centroids)[, 2]

map_data_sf$size <- (map_data_sf$prop_fraud + 1) ^ 100

# Plotting the map with circles/state names
fig_3.2_sf <- ggplot(data = map_data_sf) +
  geom_point(aes(x = lon, y = lat, size = size), color = "blue") +
  geom_text(aes(x = lon, y = lat, label = NAME_1), check_overlap = TRUE,
            nudge_x = 0.1, nudge_y = 0.1) +
  labs(title = "Figure 3.2: Representation of Fraud by State Size") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        plot.title = element_text(size = 10),
        legend.position = "none")

plot(fig_3.2_sf)
```

Figure 3.2: Representation of Fraud by State Size



```
# I believe this simple design with (blue) circles is much simpler yet
# successfully conveys the same message as the more complex map of Mexico. One
# can quickly locate a state of interest by geographical location, and the size
# of the state is meaningful. Moreover this graph stands out and for that very
# reason may get the audience's attention better.
```