

INFERENCIA ESTADÍSTICA

GRUPO: 9178

EQUIPO:

- Vargas Reyes Alberto**
- García Pacheco Jorge Axel**
- Real Herrera Juan Carlos**

**TAREA-EXAMEN PRIMERA PARTE
REPORTE**

ÍNDICE

Introducción

Metodología

Resultados

Referencias

Anexos

INTRODUCCIÓN

En esta tarea, el objetivo es determinar la composición del Senado de la República de México para las elecciones de 2024. Para ello, se hace uso de los cómputos distritales presentes en el archivo SEN_2024.csv y de los convenios de coalición entre partidos disponibles en el archivo convenios_coaliciones_senadores_2024.xls. El análisis se basa en la asignación de senadores por los principios de mayoría relativa y primera minoría.

METODOLOGÍA

- DATOS:

Se utilizó el archivo SEN_2024.csv para obtener los cómputos distritales por entidad federativa. El archivo fue leído utilizando la función read_csv(). El original se editó e igual me deshice de todas las columnas que no son los votos para no llegar a cometer errores y tener menos ruido.

```
base <- read_csv("C:/Users/IDINSA/Downloads/SEN_2024.csv", locale = locale(encoding = "WINDOWS-1252"))
```

- AJUSTE DE NOMBRES DE COLUMNAS:

Se renombraron algunas columnas para facilitar su uso posterior, en particular las relacionadas con "Independientes" y "Nulos".

```
colnames(base)[17] <- "Independientes"
```

```
colnames(base)[18] <- "Nulos"
```

- AGREGACIÓN DE DATOS POR ENTIDAD:

Se sumaron los resultados de votos por partido para cada entidad usando la función aggregate()

```
df <- aggregate(. ~ ENTIDAD, data = base, sum)
```

- CREACIÓN DE UN NUEVO DATAFRAME:

Se creó un nuevo DataFrame para calcular las sumas de los votos obtenidos por los partidos en coalición, como PRI-PAN-PRD y PT-VER-MORENA, así como los votos individuales por partido.

```
df_nuevo <- data.frame(
```

```
  PAN = df[, 2],
```

```
  PRI = df[, 3],.....
```

- CORRECCIÓN DE VALORES DE COALICIONES:

Se ajustaron los valores de las coaliciones para evitar dobles conteos en caso de que los votos totales de una coalición coincidieran con la suma individual de los partidos.

```
df_nuevo$PRI_PAN_PRD <- ifelse(df_nuevo$PRI_PAN_PRD == (df_nuevo$PRI +  
df_nuevo$PAN + df_nuevo$PRD), 0, df_nuevo$PRI_PAN_PRD).....
```

- IDENTIFICACIÓN DE MAYORÍA RELATIVA:

Se determinó qué partido o coalición obtuvo el mayor número de votos en cada entidad usando la función `apply()` para recorrer cada fila del `DataFrame` y seleccionar la columna con el valor más alto.

```
get_max_col_name <- function(x) { return(names(x)[which.max(x)]) }
```

- ASIGNACIÓN DE SENADORES POR MAYORÍA RELATIVA:

Se creó un `DataFrame` con la asignación de senadores por mayoría relativa en cada entidad.

```
df_n <- data.frame( Entidad = df$ENTIDAD, Max_Columna = max_cols
```

- MODIFICACIÓN PARA CALCULAR LA PRIMERA MINORÍA:

Después de asignar los senadores de mayoría relativa, se eliminó el partido o coalición con el mayor número de votos de cada entidad para calcular la primera minoría, es decir, el segundo lugar en las elecciones.

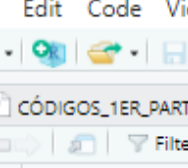
```
update_values <- function(row) { max_val <- max(row) max_col <- which.max(row) row[max_col].....
```

- Asignación de senadores por primera minoría:

Finalmente, se identificaron los partidos o coaliciones que quedaron en segundo lugar y se asignaron los senadores correspondientes.

```
max_col_names <- apply(df_nuevo_modificado, 1, get_max_col_name, colnames =  
colnames(df_nuevo_modificado)) df_max_col_names <- data.frame(
```

RESULTADOS

[illegible]

The screenshot shows the RStudio interface. At the top, the title bar reads 'INF - RStudio'. Below it is the menu bar with 'File', 'Edit', 'Code', 'View', 'Plots', 'Session', and 'Build'. The toolbar contains icons for adding files, saving, and navigating. The main editor window displays a data table with two columns: 'Valor' and 'Frecuencia'. The table has six rows of data. A red box highlights the value '2' in the 'Frecuencia' column of the last row.

	Valor	Frecuencia
1	MC	2
2	MORENA	1
3	PAN	1
4	PRI_PAN_PRD	25
5	PT_VER_MORENA	1
6	VER	2

The screenshot shows the RStudio environment. The top menu bar includes File, Edit, Code, View, Plots, Session, and Build. Below the menu is a toolbar with icons for adding files, saving, and navigating. The main window displays a script editor with a file named 'CÓDIGOS_1ER PARTE.R' and a data table titled 'Mayoría_relativa'.

	Valor	Frecuencia
1	MORENA	10
2	PRI_PAN_PRD	2
3	PT_VER_MORENA	19
4	VER	1

Below the table, a box labeled 'VER' is highlighted.

Al analizar los datos de mayoría relativa y primera minoría, se destaca: Los partidos y coaliciones que sobresalieron en términos de número de senadores asignados fueron MORENA y la coalición PRI-PAN-PRD.

El análisis de los resultados de la asignación de senadores para las elecciones de 2024 permite entender cómo se distribuye el poder político en el Senado de la República. Los resultados muestran la importancia de las coaliciones como estrategias electorales, que permitieron a partidos como el PT, PVEM, PRI, PAN, y PRD maximizar su representación en el Senado. Asimismo, MORENA sigue siendo el partido dominante en gran parte del país, pero enfrenta una competencia significativa en muchas entidades, especialmente de la coalición PRI-PAN-PRD.

REFERENCIAS

- Cómputos distritales SEN_2024.csv
- Convenios de coalición convenios_coaliciones_senadores_2024.xls

ANEXOS

Códigos

```
# install.packages("readxl")

# install.packages("tidyverse")

# install.packages("dplyr")

library(readxl)

library(tidyverse)

library(dplyr)

base <- read_csv("C:/Users/IDINSA/Downloads/SEN_2024.csv", locale = locale(encoding =
"WINDOWS-1252"))

colnames(base)[17] <- "Independientes"

colnames(base)[18] <- "Nulos"

df <- aggregate(. ~ ENTIDAD, data = base, sum)

df_nuevo <- data.frame(

  PAN = df[, 2],

  PRI = df[, 3],

  PRD = df[, 4],

  PRI_PAN_PRD = df[, 2] + df[, 3] + df[, 4] + df[, 9] + df[, 10] + df[, 11] + df[, 12],

  VER = df[, 5],

  PT = df[, 6],

  MORENA = df[, 8],

  PT_VER_MORENA = df[, 5] + df[, 6] + df[, 8] + df[, 13] + df[, 14] + df[, 15] + df[, 16],

  MC = df[, 7],

  Independientes = df[, 17],

  Nulos = df[, 18]
```

```

)

df_nuevo$PRI_PAN_PRD <- ifelse(df_nuevo$PRI_PAN_PRD == (df_nuevo$PRI + df_nuevo$PAN +
df_nuevo$PRD), 0, df_nuevo$PRI_PAN_PRD)

df_nuevo$PT_VER_MORENA <- ifelse(df_nuevo$PT_VER_MORENA == (df_nuevo$PT +
df_nuevo$VER + df_nuevo$MORENA), 0, df_nuevo$PT_VER_MORENA)


get_max_col_name <- function(x) {
  return(names(x)[which.max(x)])
}

max_cols <- apply(df_nuevo, 1, get_max_col_name)

df_n <- data.frame(
  Entidad = df$ENTIDAD,
  Max_Columna = max_cols
)

conteo <- table(df_n$Max_Columna)

Mayoría_relativa <- as.data.frame(conteo)

colnames(Mayoría_relativa) <- c("Valor", "Frecuencia")

View(Mayoría_relativa)

update_values <- function(row) {
  max_val <- max(row)
  max_col <- which.max(row)
  row[max_col] <- 0
  if (max_col == 4) {
    row[c(1, 2, 3)] <- 0
  }
  if (max_col == 8) {
    row[c(5, 6, 7)] <- 0
  }
  return(row)
}

```

```

df_nuevo_modificado <- t(apply(df_nuevo, 1, update_values))
df_nuevo_modificado <- as.data.frame(df_nuevo_modificado)

colnames(df_nuevo_modificado) <- colnames(df_nuevo)

get_max_col_name <- function(row, colnames) {
  max_col_index <- which.max(row)
  return(colnames[max_col_index])
}

colnames_df <- colnames(df_nuevo_modificado)
max_col_names <- apply(df_nuevo_modificado, 1, get_max_col_name, colnames = colnames_df)
df_max_col_names <- data.frame(
  Entidad = df$ENTIDAD,
  Max_Columna = max_col_names
)

conteo_max_col <- table(df_max_col_names$Max_Columna)
Primera_minoría <- as.data.frame(conteo_max_col)
colnames(Primera_minoría) <- c("Valor", "Frecuencia")
View(Primera_minoría)

```