

# Problem set 3

Juan Kamilo Narvaez

## 0. Configuración inicial

```
#Limpiar el entorno
rm(list=ls())
setwd("C:/Users/narva/OneDrive/Documentos/GitHub/pset3-202113808")

# instalar/llamar pacman
require(pacman)
```

```
## Loading required package: pacman
```

```
p_load(rio,
        skimr,
        janitor,
        data.table,
        tidyverse,
        ggplot2,
        dplyr,
        tinytex,
        rmarkdown)

#Verificar el directorio
getwd()
```

```
## [1] "C:/Users/narva/OneDrive/Documentos/GitHub/pset3-202113808"
```

```
list.files()
```

```
## [1] "code"          "input"          "output"
## [4] "pset3-202113808.Rproj" "pset3-markdown.pdf" "pset3-markdown.Rmd"
```

## 1. Bucle

### 1.1. Lista de archivos en input

```
rutas <- list.files("input/" , recursive=T , full.names=T)
```

## 1.2. Importar archivos con características generales

```
## Extraer las rutas
rutas_resto <- str_subset(string = rutas, pattern = "Resto - C")

## Cargar en lista
lista_resto <- import_list(file = rutas_resto)

## Textear la cadena de caracteres
rutas_resto[1]
```

```
## [1] "input/2019/Abril.csv/Resto - Características generales (Personas).csv"
```

```
str_sub(rutas_resto[20], start = 7 , 10)
```

```
## [1] "2020"
```

```
## Agregar ruta
View(lista_resto[[1]])
lista_resto[[1]]$path <- rutas_resto[1]

## Aplicar loop
for (i in 1:length(lista_resto)){
  lista_resto[[i]]$path <- rutas_resto[i]
  lista_resto[[i]]$year <- str_sub(lista_resto[[i]]$path, start = 7 , 10)
}
View(lista_resto[[35]])
```

## 1.3. Combinar conjunto de datos

```
lista_resto[[36]] <- NULL #elimina el data frame 36 no cumple con los requerimientos
df_rest <- rbindlist(l=lista_resto , use.names=T , fill=T)
## exporta
export(df_rest, "output/cg.rds")
```

## 2. Visualizaciones

```
#base de datos que se usara
cg_geih = import("output/cg.rds")
```

### Grafico 1: Raza y edad

```

#reemplazar nulos en raza (P6080) como ninguna raza
cg_geih$P6080 <- ifelse(is.na(cg_geih$P6080), 6, cg_geih$P6080)

#cambiar los valores para tenerlas como categoricas
cg_geih$P6080 <- factor(cg_geih$P6080, levels = 1:6, labels = c(
  "Indigena",
  "Gitano",
  "Raizal",
  "Palenquero",
  "Afro",
  "Ninguno"
))

# conjunto de datos para los graficos cg_geih
# No se usaran estas grupos raciales
cg_geih_g1 <- subset(cg_geih, !(P6080 %in% c("Gitano", "Raizal", "Palenquero")))

## Grafico de violin
g1 <- ggplot(cg_geih_g1, aes(x = P6080, y = P6040, fill = P6080)) +
  geom_violin(color = "white", trim = FALSE) +
  scale_fill_brewer(palette = "Set3") + # Cambia la paleta de colores
  labs(title = "Diagrama de Violín de Edad por Raza", x = "Raza", y = "Edad") +
  theme_minimal() + # Cambia el tema del gráfico
  theme(legend.position = "none")

# guarda el grafico
ggsave("output/plot1.jpeg", plot = g1, device = "jpeg")

## Saving 6.5 x 4.5 in image

```

## Grafico

```
plot(g1)
```

Diagrama de Violín de Edad por Raza

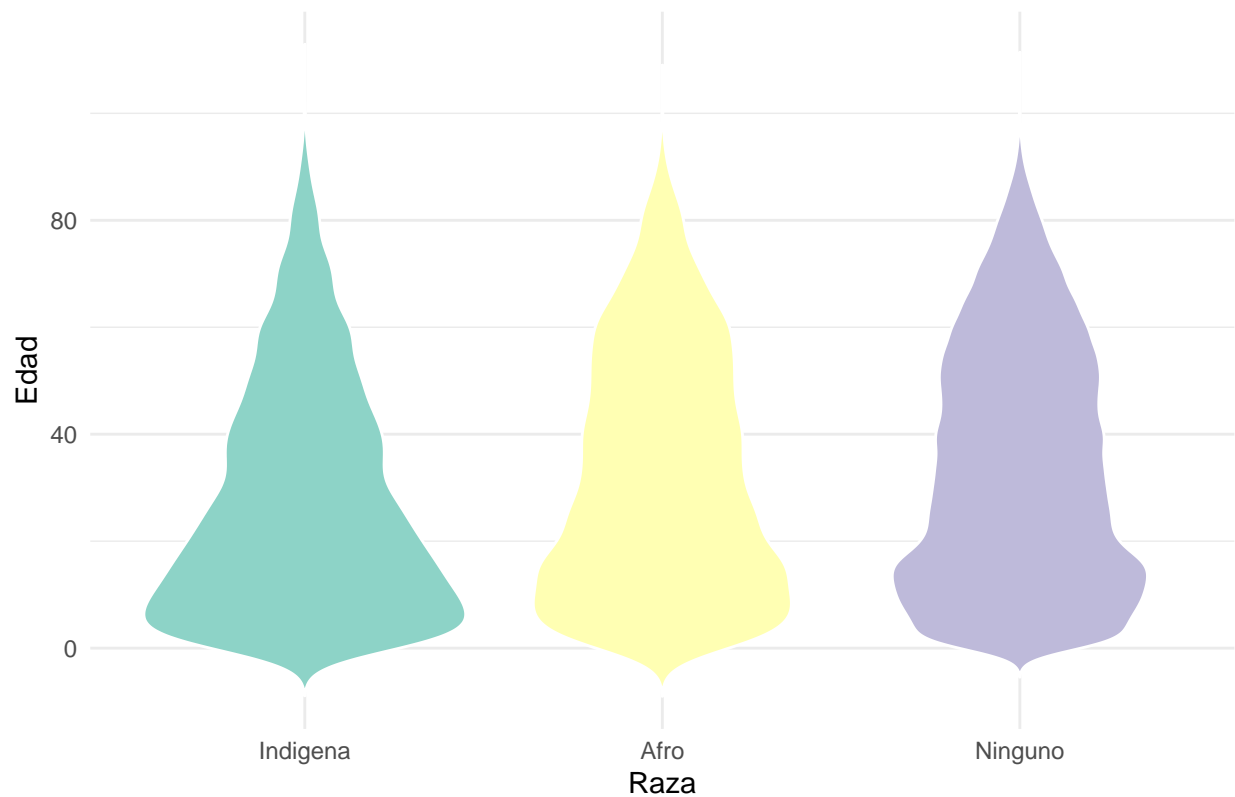


Grafico 2: Alfabetismo por raza

```
#Base de datos para grafico: mayores de 15 años
cg_geih_g2 <- subset(cg_geih_g1, P6040 > 15)
#asignar etiquetas de alfabetismo
cg_geih_g2 <- cg_geih_g2%>%
  mutate(P6160 = factor(P6160, levels = c(1, 2), labels = c("Sabe Leer", "No Sabe Leer")))

#Generar el grafico
g2 <- ggplot(cg_geih_g2, aes(fill = P6160, x = P6080)) +
  geom_bar(position = "fill", color = "white", width = 0.7) +
  labs(title = "Proporción de Personas que Saben Leer según la Raza (Mayores de 15)",
       x = "Raza",
       y = "Proporcion",
       fill = "Sabe leer") +
  theme_minimal()

#Guardar el grafico
ggsave("output/plot2t.jpeg", plot = g2, device = "jpeg")
```

```
## Saving 6.5 x 4.5 in image
```

## Grafico

