Revisión 2021

Ángela Vieyto 5.487.839-8

4/6/2021

Explicativo sobre la prueba

El examen es individual y cualquier apartamiento de esto invalidará la prueba. Puede consultar el libro del curso durante la revisión http://r4ds.had.co.nz así como el libro de ggplot2 pero no consultar otras fuentes de información.

Además una vez finalizada la prueba debes mandarme el archivo pdf y Rmd a natalia@iesta.edu.uy y por favor recordame tu usuario de GitHub para que sea más sencillo encontrar tu repositorio, asegurate que haya aceptado la invitación a tu repositorio y de no ser así enviame nuevamente la invitación a natydasilva.

La Revisión vale 130 puntos donde 15 de los puntos son de reproducibilidad de la misma, organización del repositorio en GitHub, órden y organización en el código y respuestas.

Ejercicio 1 (90 puntos)

Explicativo sobre los datos

Los datos que vamos a utilizar en este ejercicio son una muestra de datos a nivel nacional sobre abandono escolar en los años 2016 que ya utilizamos en la Tarea 2.

En el Cuadro 1 se presentan las variables en el conjunto de datos **muestra.csv**.

Table 1: Variables en **muestra.csv**

Variable	Descripción
documento	Cédula de Identidad del alumno
nro_doc_centro_educ	Liceo que concurre el alumno en 2016
nombre_departamento	Nombre del Departamento del centro educativo
grupo_desc	Grupo del alumno en 2016
coberturaT	Cobertura en el primer semestre de 2016
Centro_Grupo	Liceo y grupo del alumno en 2016
cl	Cluster - contexto sociocultural del liceo en 1016
Grado_2016_UE	Grado del alumno en el 2016 según UE
Grado2013	Grado del alumno en 2013 según CRM
Grado2014	Grado del alumno en 2014 según CRM
Grado2015	Grado del alumno en 2015 según CRM
Grado 2016	Grado del alumno en 2016 según CRM
Sexo	Sexo del alumno
Fecha.nacimiento	Fecha de nacimiento del alumno
Grupo_UE_2017	Grupo del alumno en 2017
inasistencias	cantidad de inasistencias en el primer semestre de 2016
asistencias	cantidad de asistencias en el primer semestre de 2016

1. Dentro de tu proyecto de RStudio creá un subdirectorio llamado Datos y copiá el archivo muestra.csv. Lee los datos usando alguna función de la librería `readr` y `here`. **(5 puntos)**

```
library(readr)
library(here)
library(tidyverse)
library(xtable)
```

```
muestra <- read_csv(here("Prueba/Datos/muestra.csv"))
```

2. Utilizando funciones de `dplyr` transformá la variable Abandono para que sea un factor con dos niveles donde el 0 se recodifique a No y el 1 a Si. Mostrame el resultado resumido en una tabla con la cantidad de observaciones para cada categoría usando `xtable`, recordá incluir en el chunk `results='asis'`. **(10 puntos)**

```
muestra <- muestra %>%
            mutate(new_Abandono = factor(Abandono, labels = c("0" = "No",
                                                              "1" = "Si")))
#xtable(muestra$Abandono, muestra$new_Abandono)
```

3. Usando funciones de `dplyr` respondé ¿Cuál es el porcentaje de abandono en Montevideo? **(10 puntos)**

```
muestra %>%
  filter(nombre_departamento == "Montevideo") %>%
    summarise(total = n(),
              abandonos = sum(Abandono),
              porcentaje = abandonos/total)
```

```
## # A tibble: 1 x 3
```

```
##   total abandonos porcentaje
##   <int>     <dbl>      <dbl>
## 1  1248        67     0.0537
```

4. Reproducí el siguiente gráfico y en vez de "Gráfico a replicar" (`caption`) debes agregar un título que describa la figura y algún comentario interesante de lo que observás en la misma.**(10 puntos)**

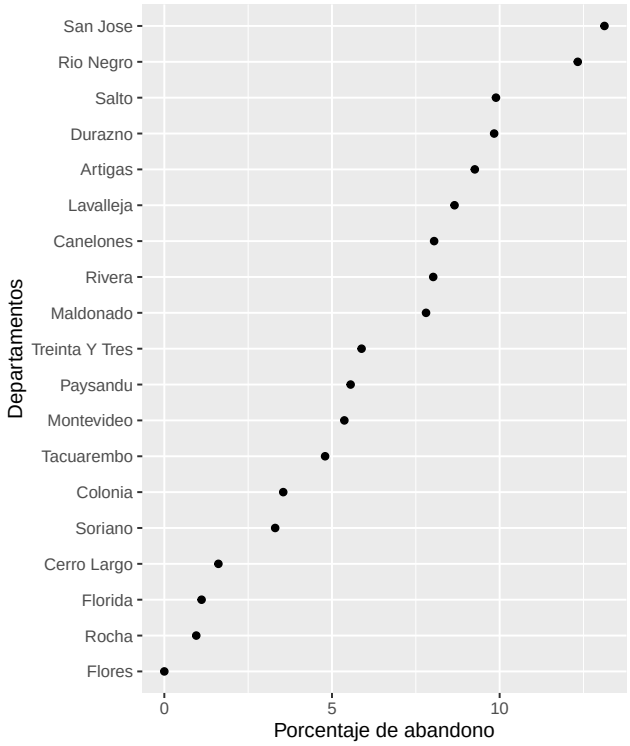

Figure 1: Gráfico a replicar

```r
muestra %>%
  group_by(nombre_departamento) %>%
    summarise(total = n(),
              abandonos = sum(Abandono),
              porcentaje = abandonos/total) %>%
  ggplot(aes(x = porcentaje, y = reorder(nombre_departamento, porcentaje))) +
  geom_point() +
  labs(x = "Porcentaje de abandono",
       y = "Departamentos")
```

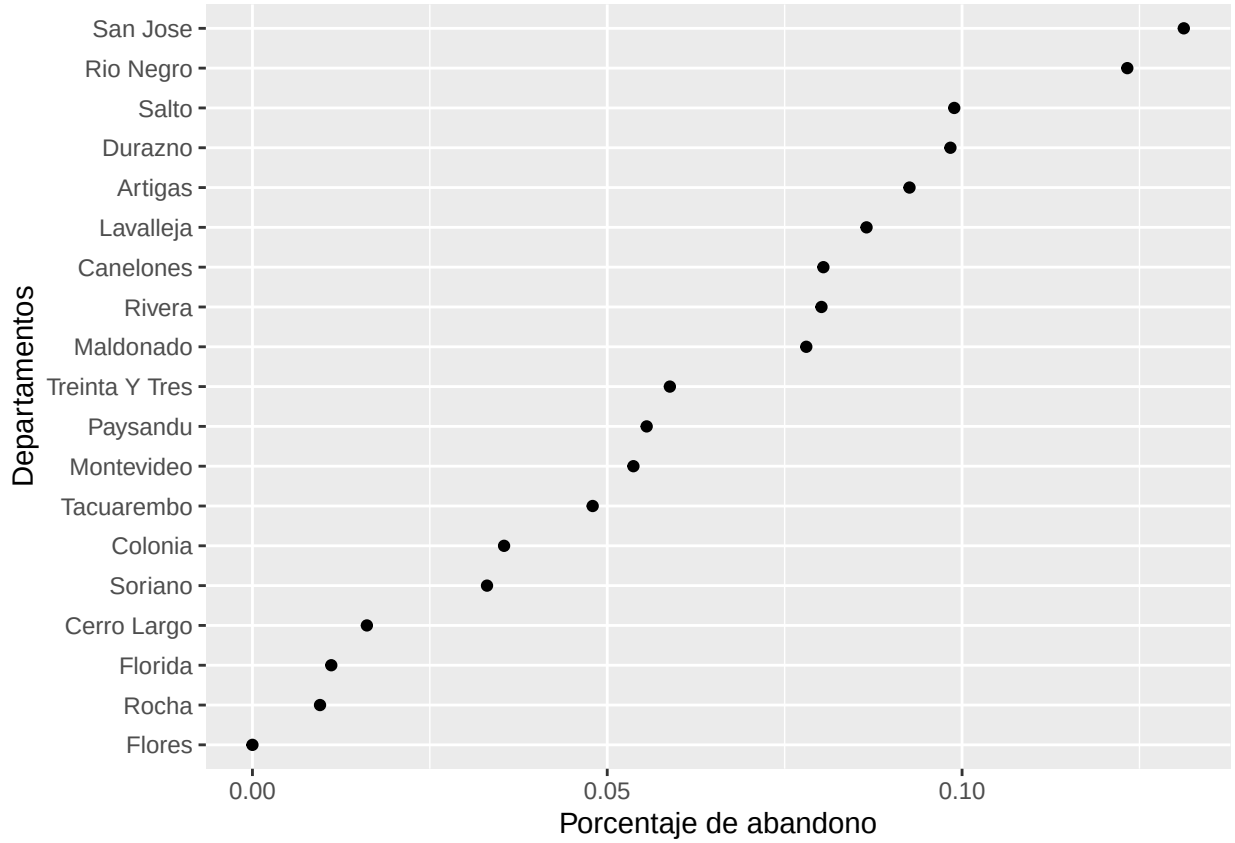

Figure 2: Porcentaje de abandono según departamento

San José es el departamento con una tasa de abandono más alta, mientras que Flores es el deaprtamento con una tasa de abandono más baja. No obstante ello, ningún departamento supera el 15% de abandono.

5. Reproducí el siguiente gráfico realizado solo con los estudiantes que abandonaron y en vez de "Gráfico a replicar" (`caption`) debes agregar un título que describa la figura y algún comentario interesante de lo que observás en la misma. La paleta usada es Dark2. **(10 puntos)**

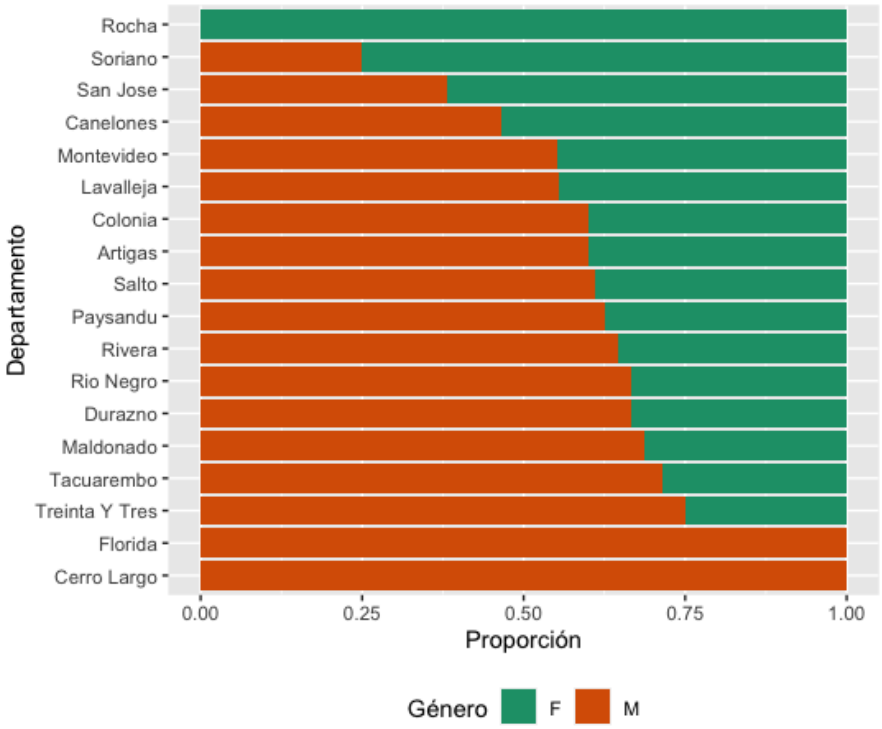

Figure 3: Gráfico a replicar

```
muestra %>%
  filter(Abandono == 1) %>%
    ggplot() +
    geom_bar(aes(x = nombre_departamento, fill = Sexo),
           position = "fill") +
    coord_flip() +
    labs(x = "Departamento",
         y = "Proporción",
         fill = "Género") +
    theme(legend.position = "bottom") +
    scale_fill_brewer( palette = "Dark2")
```

Llama la atención que en el departamento de Rocha, el 100% de los estudiantes que abandonan son mujeres, mientras que en Florida y Cerro Largo ocurre exactamente lo opuesto.

6. Reproducí el siguiente gráfico y en vez de "Gráfico a replicar" (`caption`) debes agregar un título que describa la figura y algún comentario interesante de lo que observás en la misma. La paleta usada es Dark2.**(15 puntos)**

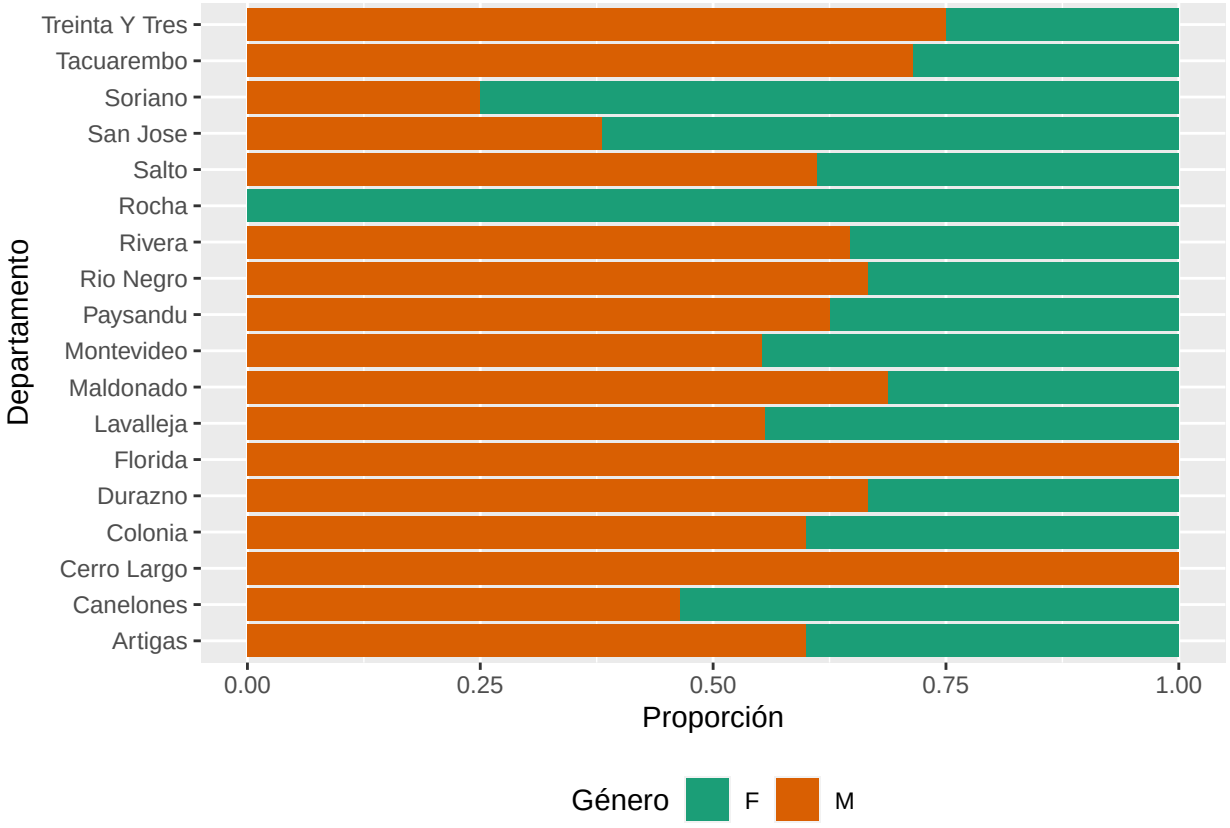

Figure 4: Distribución de estudiantes que abandonan según sexo

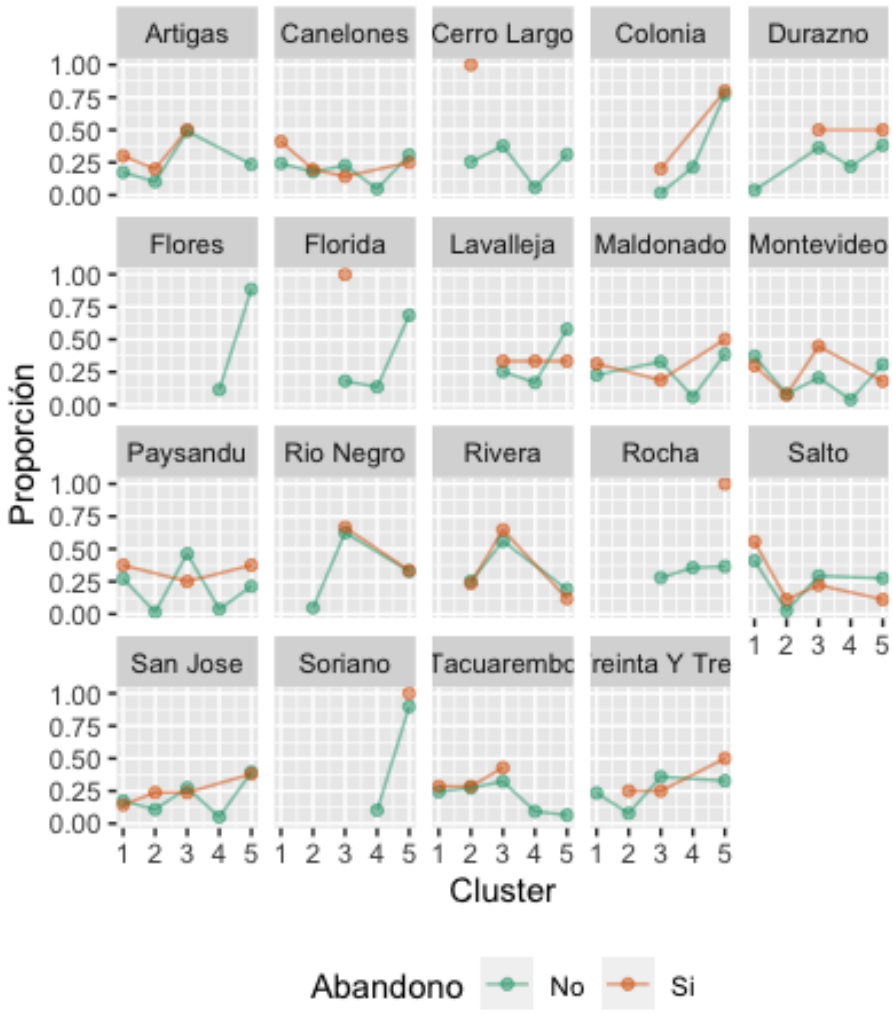

Figure 5: Gráfico a replicar

```
muestra %>%
  summarise(total = n(),
            Abandono,
            cl,
            nombre_departamento) %>%
  group_by(Abandono) %>%
    summarise(porcentaje = n()/total,
              cl,
              nombre_departamento) %>%
  ggplot(aes(x = cl, y = porcentaje), color = Abandono) +
  geom_point() +
  geom_line() +
  facet_wrap(vars(nombre_departamento)) +
  scale_color_brewer(palette = "Dark2")
```

7. Recodificá la variable `grupo_desc` que tiene 17 niveles para que de 1ro.G.1 a 1ro.G5 sea A de 1ro.G.6

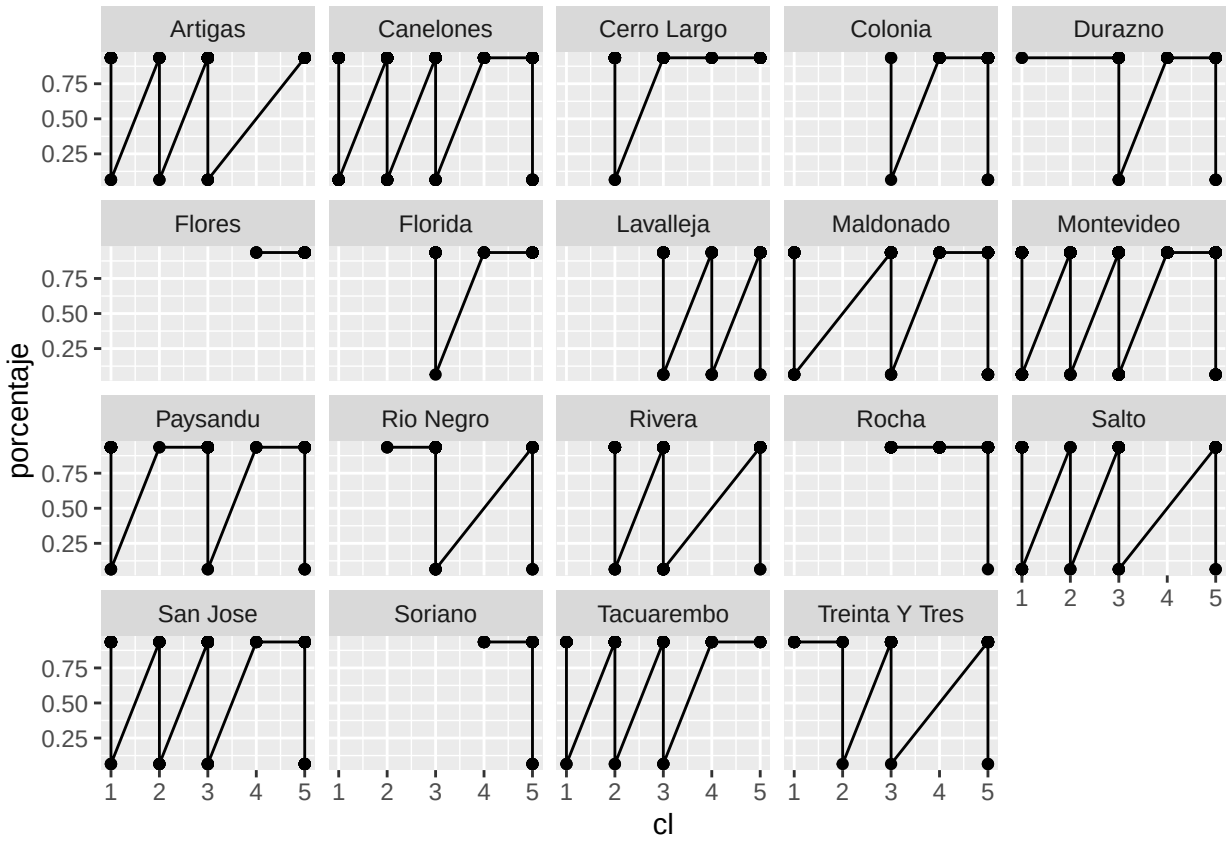

Figure 6: Abandono según departamento

a 1ro G.11 sea B y los restantes C. Mostrá el resultado seleccionando la variable recodificada y las primeras 6 filas. **(5 puntos)**

```
aux <- c()
for (i in seq_along(muestra$grupo_desc)) {
  if (muestra$grupo_desc[i] == "1ro.G.1" |
      muestra$grupo_desc[i] == "1ro.G.2" |
      muestra$grupo_desc[i] == "1ro.G.3" |
      muestra$grupo_desc[i] == "1ro.G.4" |
      muestra$grupo_desc[i] == "1ro.G.5") {
    aux <- c(aux, "A")
  } else if (muestra$grupo_desc[i] == "1ro.G.6" |
             muestra$grupo_desc[i] == "1ro.G.7" |
             muestra$grupo_desc[i] == "1ro.G.8" |
             muestra$grupo_desc[i] == "1ro.G.9" |
             muestra$grupo_desc[i] == "1ro.G.10" |
            muestra$grupo_desc[i] == "1ro.G.11") {
    aux <- c(aux, "B")
  } else {
    aux <- c(aux, "C")
  }
}

muestra$grupo_desc <- factor(aux, levels = c("A", "B", "C"))
table(muestra$grupo_desc)
```

8. Separá la variable Fecha.nacimiento en tres nuevas variables año, mes y dia, para ello usá la función `separate` de forma que sean numéricas. Mostrá el resultado seleccionando las variables documento, año, dia y mes con alguna función de `dplyr` y las primeras 6 filas.**(5 puntos)**

```
muestra %>%
  mutate(separate(col = `Fecha nacimiento`,
                  into = c("año", "mes", "día"),
                  sep = "-")) %>%
  summarise(documento,
            año,
            mes,
            día) %>%
  head(6)
```

9. Convertí la variable Fecha.nacimiento como objeto de tipo Date usando `as.Date` de R base y comprobá que la nueva variable Fecha.nacimiento es del tipo correcto. **(5 puntos)**

```
muestra %>%
  mutate(new_fecha_nacimiento = as.Date(`Fecha nacimiento`))
```

```
## # A tibble: 4,023 x 22
##       X1 documento nro_doc_centro_~ nombre_departam~ nombre_localidad grupo_desc
##    <dbl>     <dbl>            <dbl> <chr>            <chr>            <chr>
##  1     1  52401872         12101064 Montevideo       Cilindro         1ro. G. 1
##  2     2  54975382         12171702 Soriano          Cardona          1ro. G. 1
##  3     3  54944549         12101048 Montevideo       Manga            1ro. G. 2
```

```
##  4     4  56682298         12101049 Montevideo       Punta De Rieles  1ro. G. 6
##  5     5  52345771         12161614 San Jose         Delta Del Tigre  1ro. G. 5
##  6     6  57399006         12101001 Montevideo       Centro           1ro. G. 3
##  7     7  53429100         12111109 Paysandu         PaysandÃº        1ro. G. 3
##  8     8  55179587          1209911 Maldonado        Maldonado        1ro. G. 5
##  9     9  53738666          1206601 Flores           Trinidad         1ro. G. 1
## 10    10  53800394          1202220 Canelones        San Bautista     1ro. G. 2
## # ... with 4,013 more rows, and 16 more variables: coberturaT <dbl>,
## #   Centro_Grupo <chr>, cl <dbl>, Grado_2016_UE <dbl>, Grado2013 <chr>,
## #   Grado2014 <chr>, Grado2015 <chr>, Grado2016 <chr>, Sexo <chr>,
## #   Fecha nacimiento <date>, Grupo_UE_2017 <chr>, inasistencias <dbl>,
## #   asistencias <dbl>, Abandono <dbl>, new_Abandono <fct>,
## #   new_fecha_nacimiento <date>
```

10. Usando la variable Fecha.nacimiento transformada, se considera que el alumno tiene extra-edad leve cuando nace antes del 30 de abril de 2003. Es decir, tiene un a~no m'as de la edad normativa para dicha generaci'on. En base a esta definición creá una nueva variable (nombrala extra) que valga 1 si el alumno tiene extra edad leve y 0 si no la tiene. Muestra solo el resultado de las primeras 6 filas. Pista para que la condición tome en cuenta el formato fecha podrías usar `as.Date('2003-04-30')`. **(10 puntos)**

11. Trabajá con un subconjunto de datos que tenga documento, Grado2013, Grado2014,Grado2015, Grado2016 y llamale reducida. Con los datos reducidos restructuralos para que queden de la siguiente forma usando alguna de las funciones del paquete `tidyr` que vimos en la última clase.**(5 puntos)**

```
A tibble: 16,092 x 3
   documento  Grado   Nivel
       <int>  <chr>   <chr>
 1  52401872 Grado2013 4º
 2  52401872 Grado2014 5º
 3  52401872 Grado2015 6º
 4  52401872 Grado2016 1
 5  54975382 Grado2013 5º
 6  54975382 Grado2014 6º
 7  54975382 Grado2015 1u
 8  54975382 Grado2016 1
 9  54944549 Grado2013 4º
10  54944549 Grado2014 5º
```

Ejercicio 2 (25 puntos)

1. En clase vimos distintas visualizaciones para variables categóricas y mencionamos como posibles el gráfico de barras y el gráficos de torta.

¿Cuál es el argumento teórico para decir que es siempre preferible un gráfico de barras a uno de tortas para ver la distribución de una variable categórica? **(5 puntos)**

Porque el gráfico de barras facilita la visualización en tanto el gráfico de torta puede se más complicado de entender y no lograr transmitir lo que se supone debería transmitir.

2. ¿Porqué es necesario utilizar `aspect.ratio = 1` en un diagrama de dispersión? **(5 puntos)**

3. Generá una función `compra` que tenga como argumentos un vector numérico `cprod` cantidad de productos a comprar de cada tipo y un vector numérico `cdisp` con la cantidad disponible de dichos productos (ambos vectores del mismo largo) que devuelva 1 si se pude hacer la compra y 0 en caso contrario. La compra se puede realizar siempre que haya stock suficiente para cada producto, es decir que la cantidad disponible sea igual o mayor a la cantidad comprada. A su vez si alguno de los argumentos no es un vector numérico la función no debe ser evaluada y debe imprimir el mensaje "Argumento no numérico". **(15 puntos)**

Comprobá que el resultado de la función sea

`compra(c(1,4,2), 1:3) = 0`

`compra(c("A","B"), 1:3)= Argumento no numérico`

```
compra <- function (cprod, cdisp) {
  if (is.numeric(cprod) == FALSE | is.numeric(cdisp) == FALSE) {
    print("Argumento no numérico.")
  } else if (length(cprod) != length(cdisp)) {
    compra = 0
  } else if (cprod < cdisp) {
    compra = 1
  } else {
    compra = 0
  }
  return(compra)
}

compra(c(1,4,2), 1:3)
```

```
## [1] 0
```

```
compra(c("A","B"), 1:3)
```

```
## [1] "Argumento no numérico."
```

```
## function (cprod, cdisp) {
##   if (is.numeric(cprod) == FALSE | is.numeric(cdisp) == FALSE) {
##     print("Argumento no numérico.")
##   } else if (length(cprod) != length(cdisp)) {
##     compra = 0
##   } else if (cprod < cdisp) {
##     compra = 1
##   } else {
##     compra = 0
##   }
##   return(compra)
## }
## <bytecode: 0x00000000194c4010>
```