Ejercicios Tidyverse 2

Laboratorio Políticas Públicas

28/4/2020

Practiquemos lo que vimos hoy!:

Generemos una tabla única que aglutine la información de "data_clase_final", "poblacion_edad" y "casos_muertos"!

Para esto vamos a trabajar con los dataset presentes en el siguiente link.

1. Llamemos a Tidyverse!

```
#activemos tidyverse
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.6.3
```

```
## -- Attaching packages ---------------------------------------------------------------------------- t
```

```
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
## Warning: package 'tibble' was built under R version 3.6.3
```

```
## Warning: package 'tidyr' was built under R version 3.6.3
```

```
## Warning: package 'readr' was built under R version 3.6.3
```

```
## Warning: package 'purrr' was built under R version 3.6.3
```

```
## Warning: package 'dplyr' was built under R version 3.6.3
```

```
## Warning: package 'stringr' was built under R version 3.6.3
```

```
## Warning: package 'forcats' was built under R version 3.6.3
```

```
## -- Conflicts ------------------------------------------------------------------------------- tidyver
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

2.a. Carguemos nuestro dataset llamado *"data_clase_final.csv"* en una nueva variable llamada "data_clase_final" (acordate de modificar el encoding en Latin1)

```
#carguemos nuestro dataset
data_clase_final <- read.csv("https://raw.githubusercontent.com/labpoliticasuba/Clases_2020/master/Clas
                             encoding = "UTF-8")
```

2.b. Demos un pantallazo a nuestro dataset usando la función tail():

```
#demos un pantallazo a nuestro dataset con tail
head(data_clase_final)
```

```
##     REGION     DISTRITO CASOS MUERTOS POBLACION                TIPO_CAMA
## 1 PAMPEANA Buenos Aires   741      39  17196396 Aislamiento y casos leves
## 2 PAMPEANA Buenos Aires   741      39  17196396              Casos graves
## 3 PAMPEANA Buenos Aires   741      39  17196396          Casos graves Neo
## 4 PAMPEANA         CABA  3046     141   3068043 Aislamiento y casos leves
## 5 PAMPEANA         CABA  3046     141   3068043              Casos graves
## 6 PAMPEANA         CABA  3046     141   3068043          Casos graves Neo
##   CANTIDAD_CAMAS CANTIDAD_MEDICOS
## 1          37337            46223
## 2          57151            46223
## 3           2180            46223
## 4          16173            48447
## 5           5095            48447
## 6           1000            48447
```

2.c. Conozcamos cuáles son las columnas que lo integran:

```
colnames(data_clase_final)
```

```
## [1] "REGION"           "DISTRITO"         "CASOS"            "MUERTOS"
## [5] "POBLACION"        "TIPO_CAMA"        "CANTIDAD_CAMAS"   "CANTIDAD_MEDICOS"
```

3.a Traigamos nuestro dataset *"poblacion_edad.csv"* y llamemoslo "poblacion_edad"

```
poblacion_edad <- read.csv("https://raw.githubusercontent.com/labpoliticasuba/Clases_2020/master/Clase%
                           encoding = "Latin1")
```

3.b. Examinemos las columnas con un names(): - PROVINCIA: contiene los datos del distrito - X0.19: Cantidad de hab. de 0 a 19 años - X20.39: Cantidad de hab. de 20 a 39 años - X40.59: Cantidad de hab. de 40 a 59 años - X60.79: Cantidad de hab. de 60 a 79 años - X80.99: Cantidad de hab. de 80 a 99 años

```
names(poblacion_edad)
```

```
## [1] "PROVINCIA" "X0.19"     "X20.39"    "X40.59"    "X60.79"    "X80.99"
```

3. c. Utilicemos la función gather() para generar una columna llamada “grupo_etario” que aglutine a las diferentes columnas que tienen la cantidad de hab x provincia

```
poblacion_edad <- gather(poblacion_edad, grupo_etario, value, X0.19:X80.99 )
```

3.d. Modifiquemos como estan redactadas la observaciones en la columna “grupo_etario” con la función case_when(). Deben ser reemplazadas así: - X0.19: “De 0 a 19 años” - X20.39: “De 20 a 39 años” - X40.59: “De 40 a 59 años” - X60.79: “De 60 a 79 años” - X80.99: “De 80 a 99 años”

```
poblacion_edad <- poblacion_edad %>% mutate( grupo_etario = case_when(
                grupo_etario == "X0.19" ~ 'De 0 a 19 años',
                grupo_etario == "X20.39" ~ 'De 20 a 39 años',
                grupo_etario == "X40.59" ~ 'De 40 a 59 años',
                grupo_etario == "X60.79" ~ 'De 60 a 79 años',
                grupo_etario == "X80.99" ~ 'De 80 a 99 años'))
```

3.e. ¿Cómo podemos unir el dataset de “data_clase_final” con el de “poblacion_edad” ?

3.e.i. ¿Cuál es nuestra primary key? Es decir, ¿cuál es nuestra columna que nos va a permitir unir ambas tablas ya que tienen los mismos valores?

```
#DISTRITO es nuestra primary key
```

3.e.ii. Fijemosnos el nombre de las columnas, ¿son las mismas para ambos casos? Modifiquemsolas si es necesario.

```
#nos fijamos los nombres de data_clase_final
names(data_clase_final)
```

```
## [1] "REGION"           "DISTRITO"         "CASOS"            "MUERTOS"
## [5] "POBLACION"        "TIPO_CAMA"        "CANTIDAD_CAMAS"   "CANTIDAD_MEDICOS"
```

```
#nos fijamos los nombres de poblacion_edad
names(poblacion_edad)
```

```
## [1] "PROVINCIA"    "grupo_etario" "value"
```

```
#ACA HAY DOS OPCIONES A LA HORA DE CAMBIARLO:

#OPCION1:
poblacion_edad <- poblacion_edad %>%
  rename(
    DISTRITO = PROVINCIA)
```

```
#OPCION2:
names(poblacion_edad)[1] <- "DISTRITO"
```

3.e.iii. Revisemos que tengan los mismos valores en ambas columnas -de ambos datasets- para que la unión se dé bien entre todas las observaciones.

```r
#revisamos data_clase_final
unique(data_clase_final$DISTRITO)
```

```
##  [1] Buenos Aires        CABA                Catamarca
##  [4] Chaco               Chubut              Corrientes
##  [7] Córdoba             Entre Rios          Formosa
## [10] Jujuy               La Pampa            La Rioja
## [13] Mendoza             Misiones            Neuquen
## [16] Río Negro           Salta               San Juan
## [19] San Luis            Santa Cruz          Santa Fe
## [22] Santiago del Estero Tierra del Fuego    Tucumán
## 24 Levels: Buenos Aires CABA Catamarca Chaco Chubut Córdoba ... Tucumán
```

```r
#revisamos poblacion_edad
unique(poblacion_edad$DISTRITO)
```

```
##  [1] Buenos Aires                    Catamarca
##  [3] Chaco                           Chubut
##  [5] Ciudad Autónoma de Buenos Aires Córdoba
##  [7] Corrientes                      Entre Rios
##  [9] Formosa                         Jujuy
## [11] La Pampa                        La Rioja
## [13] Mendoza                         Misiones
## [15] Neuquen                         Río Negro
## [17] Salta                           San Juan
## [19] San Luis                        Santa Cruz
## [21] Santa Fe                        Santiago del Estero
## [23] Tierra del Fuego                Tucumán
## 24 Levels: Buenos Aires Catamarca Chaco ... Tucumán
```

3.e.iv. Modificá los valores utilizando el siguiente fragmento:

```r
#cambiamos el tipo de objeto:
data_clase_final$DISTRITO <- as.character(as.factor(data_clase_final$DISTRITO))
#cambiamos el tipo de objeto:
poblacion_edad$DISTRITO <- as.character(as.factor(poblacion_edad$DISTRITO))
#le pedimos que cada vez que encuentre una observacion en la columna DISTRITO
#y que sea igual a "Ciudad A..." la reemplace por "CABA"
poblacion_edad$DISTRITO[poblacion_edad$DISTRITO == "Ciudad Autónoma de Buenos Aires"] <- 'CABA'
poblacion_edad$DISTRITO[poblacion_edad$DISTRITO == "Entre Rios"] <- 'Entre Ríos'
poblacion_edad$DISTRITO[poblacion_edad$DISTRITO == "Neuquen"] <- 'Neuquén'
```

4. Realicemos el JOIN entre “data_clase_final” y “poblacion_edad” en un nuevo dataset llamado “data_poblacion”

```r
data_poblacion <- left_join(data_clase_final, poblacion_edad)
```

```
## Joining, by = "DISTRITO"
```

```r
head(data_poblacion)
```

```
##     REGION     DISTRITO CASOS MUERTOS POBLACION                TIPO_CAMA
## 1 PAMPEANA Buenos Aires   741      39  17196396 Aislamiento y casos leves
## 2 PAMPEANA Buenos Aires   741      39  17196396 Aislamiento y casos leves
## 3 PAMPEANA Buenos Aires   741      39  17196396 Aislamiento y casos leves
## 4 PAMPEANA Buenos Aires   741      39  17196396 Aislamiento y casos leves
## 5 PAMPEANA Buenos Aires   741      39  17196396 Aislamiento y casos leves
## 6 PAMPEANA Buenos Aires   741      39  17196396              Casos graves
##   CANTIDAD_CAMAS CANTIDAD_MEDICOS    grupo_etario   value
## 1          37337            46223  De 0 a 19 años 5575989
## 2          37337            46223 De 20 a 39 años 5116515
## 3          37337            46223 De 40 a 59 años 4003370
## 4          37337            46223 De 60 a 79 años 2346634
## 5          37337            46223 De 80 a 99 años  495175
## 6          57151            46223  De 0 a 19 años 5575989
```

5. Ahora realice un right_join() entre "data_poblacion" y "caso_muertos.csv" utilizando lo aprendido! Genere un nuevo dataset llamado: "data_poblacion_casos" donde haya una columna para la cantidad de infectados y otra para la cantidad de fallecidos que se suma al dataset recien realizado de "data_poblacion"

```r
#traemos nuestro dataset de casos_muertos
casos_muertos <- read.csv("D:/Guada/Clases/Lab_Pol_Publ/Tidyverse/vf2/casos_muertos.csv",
                          encoding = "Latin1")
head(casos_muertos,24)
```

```
##                          Provincia Infectados_Muertos Cantidad
## 1  Ciudad Autónoma de Buenos Aires         Infectados     1020
## 2        Provincia de Buenos Aires         Infectados     1334
## 3                          Córdoba         Infectados      277
## 4                         Santa Fe         Infectados      241
## 5                          Mendoza         Infectados       75
## 6                            Chaco         Infectados      292
## 7                 Tierra del Fuego         Infectados      130
## 8                       Entre Rios         Infectados       22
## 9                          Tucumán         Infectados       35
## 10                           Salta         Infectados        4
## 11                           Jujuy         Infectados        5
## 12                        La Rioja         Infectados       50
## 13                        San Juan         Infectados        2
## 14                       Catamarca         Infectados        0
## 15                        San Luis         Infectados       10
## 16                          Neuquen         Infectados      106
## 17                       Río Negro         Infectados      184
## 18                          Chubut         Infectados        2
## 19                       Santa Cruz         Infectados       43
## 20                       Corrientes         Infectados       47
## 21                         Misiones         Infectados        7
## 22                        La Pampa         Infectados        5
## 23             Santiago del Estero         Infectados       15
## 24                         Formosa         Infectados        0
```

```r
#utilizamos el spread para dividir en dos columnas infectados y muertos
casos_muertos <- spread(casos_muertos, Infectados_Muertos, Cantidad)
head(casos_muertos)
```

```
##                         Provincia Infectados Muertos
## 1                       Catamarca          0       0
## 2                           Chaco        292      12
## 3                         Chubut           2       0
## 4 Ciudad Autónoma de Buenos Aires       1020      58
## 5                          Córdoba        277      12
## 6                       Corrientes         47       0
```

```r
#cambiamos el nombre de nuestra columna Provincia para hacer el join
names(casos_muertos)[1] <- "DISTRITO"
```

```r
#conocemos los valores únicos de nuestra clave primaria en data_poblacion
unique(data_poblacion$DISTRITO)
```

```
##  [1] "Buenos Aires"        "CABA"                "Catamarca"
##  [4] "Chaco"               "Chubut"              "Corrientes"
##  [7] "Córdoba"             "Entre Rios"          "Formosa"
## [10] "Jujuy"               "La Pampa"            "La Rioja"
## [13] "Mendoza"             "Misiones"            "Neuquen"
## [16] "Río Negro"           "Salta"               "San Juan"
## [19] "San Luis"            "Santa Cruz"          "Santa Fe"
## [22] "Santiago del Estero" "Tierra del Fuego"    "Tucumán"
```

```r
#conocemos los valores únicos de nuestra clave primaria en casos_muertos
unique(casos_muertos$DISTRITO)
```

```
##  [1] Catamarca                       Chaco
##  [3] Chubut                          Ciudad Autónoma de Buenos Aires
##  [5] Córdoba                         Corrientes
##  [7] Entre Rios                      Formosa
##  [9] Jujuy                           La Pampa
## [11] La Rioja                        Mendoza
## [13] Misiones                        Neuquen
## [15] Provincia de Buenos Aires       Río Negro
## [17] Salta                           San Juan
## [19] San Luis                        Santa Cruz
## [21] Santa Fe                        Santiago del Estero
## [23] Tierra del Fuego                Tucumán
## 24 Levels: Catamarca Chaco Chubut Ciudad Autónoma de Buenos Aires ... Tucumán
```

```r
#cambiamos el nombre de los distritos que no figuran igual
casos_muertos$DISTRITO <- as.character(as.factor(casos_muertos$DISTRITO))
casos_muertos$DISTRITO[casos_muertos$DISTRITO == "Ciudad Autónoma de Buenos Aires"] <- 'CABA'
casos_muertos$DISTRITO[casos_muertos$DISTRITO == "Entre Rios"] <- 'Entre Ríos'
casos_muertos$DISTRITO[casos_muertos$DISTRITO == "Neuquen"] <- 'Neuquén'
casos_muertos$DISTRITO[casos_muertos$DISTRITO == "Provincia de Buenos Aires"] <- 'Buenos Aires'
```

```r
#realizamos el right join
data_poblacion_casos <- right_join(casos_muertos, data_poblacion)
```

```
## Joining, by = "DISTRITO"
```

```r
#vemos las columnas que tenemos
names(data_poblacion_casos)
```

```
##  [1] "DISTRITO"         "Infectados"       "Muertos"          "REGION"
##  [5] "CASOS"            "MUERTOS"          "POBLACION"        "TIPO_CAMA"
##  [9] "CANTIDAD_CAMAS"   "CANTIDAD_MEDICOS" "grupo_etario"     "value"
```

6.a Ordenemos nuestras columnas de "data_poblacion_casos" y eliminemos la columna "X"

```r
data_poblacion_casos <- data_poblacion_casos  %>%
  select("REGION", "DISTRITO", "POBLACION","grupo_etario",  "value", "TIPO_CAMA",
         "CANTIDAD_CAMAS", "CANTIDAD_MEDICOS", "Infectados", "Muertos")
```

6.b. Reemplacemos las columnas para que queden todos en mayusculas (y cambiemos "value" por "CANT_HAB_X_GRUPO")

```r
data_poblacion_casos <- data_poblacion_casos %>%
  rename(
    GRUPO_ETARIO = grupo_etario,
    CANT_HAB_X_GRUPO = value,
    INFECTADOS = Infectados,
    MUERTOS = Muertos)
head(data_poblacion_casos, 15)
```

```
##      REGION     DISTRITO POBLACION    GRUPO_ETARIO CANT_HAB_X_GRUPO
## 1  PAMPEANA Buenos Aires  17196396  De 0 a 19 años          5575989
## 2  PAMPEANA Buenos Aires  17196396 De 20 a 39 años          5116515
## 3  PAMPEANA Buenos Aires  17196396 De 40 a 59 años          4003370
## 4  PAMPEANA Buenos Aires  17196396 De 60 a 79 años          2346634
## 5  PAMPEANA Buenos Aires  17196396 De 80 a 99 años           495175
## 6  PAMPEANA Buenos Aires  17196396  De 0 a 19 años          5575989
## 7  PAMPEANA Buenos Aires  17196396 De 20 a 39 años          5116515
## 8  PAMPEANA Buenos Aires  17196396 De 40 a 59 años          4003370
## 9  PAMPEANA Buenos Aires  17196396 De 60 a 79 años          2346634
## 10 PAMPEANA Buenos Aires  17196396 De 80 a 99 años           495175
## 11 PAMPEANA Buenos Aires  17196396  De 0 a 19 años          5575989
## 12 PAMPEANA Buenos Aires  17196396 De 20 a 39 años          5116515
## 13 PAMPEANA Buenos Aires  17196396 De 40 a 59 años          4003370
## 14 PAMPEANA Buenos Aires  17196396 De 60 a 79 años          2346634
## 15 PAMPEANA Buenos Aires  17196396 De 80 a 99 años           495175
##                    TIPO_CAMA CANTIDAD_CAMAS CANTIDAD_MEDICOS INFECTADOS MUERTOS
## 1  Aislamiento y casos leves          37337            46223       1334      76
## 2  Aislamiento y casos leves          37337            46223       1334      76
## 3  Aislamiento y casos leves          37337            46223       1334      76
## 4  Aislamiento y casos leves          37337            46223       1334      76
## 5  Aislamiento y casos leves          37337            46223       1334      76
```

```
## 6              Casos graves          57151           46223       1334       76
## 7              Casos graves          57151           46223       1334       76
## 8              Casos graves          57151           46223       1334       76
## 9              Casos graves          57151           46223       1334       76
## 10             Casos graves          57151           46223       1334       76
## 11         Casos graves Neo           2180           46223       1334       76
## 12         Casos graves Neo           2180           46223       1334       76
## 13         Casos graves Neo           2180           46223       1334       76
## 14         Casos graves Neo           2180           46223       1334       76
## 15         Casos graves Neo           2180           46223       1334       76
```

7.a. Supongamos que en Argentina se va a contagiar con coronavirus el 10% de la población mayor de 60 años. En base al dataset generado anteriormente, ¿cuántas camas necesitaríamos para atenderlos si todos se enfermaran a la vez? Para responder la pregunta supongamos que el contagio respeta los patrones que se han observado en otras experiencias: el 5% de los contagiados son casos graves y el 15% de los contagiados son casos que necesitan aislamiento y cuidados paliativos. (atención: en la respuesta tiene que estar distinguida la cantidad de camas para casos graves y leves)

```
#acomodamos nuestra data:
data_7 <- spread(data_poblacion_casos, key = TIPO_CAMA, value = CANTIDAD_CAMAS)
```

```
#ccambiamos el nombre de las columnas:
names(data_7)[9] <- "CANTIDAD_CAMAS_AISLAMIENTO_LEVES"
names(data_7)[10] <- "CANTIDAD_CAMAS_GRAVES"
names(data_7)[11] <- "CANTIDAD_GRAVES_NEO"
```

```
#Agrupo la poblacion mayor en una sola categoria "POBL_MAYOR"
data_7 <- data_7 %>%

    select(DISTRITO, GRUPO_ETARIO, CANT_HAB_X_GRUPO, CANTIDAD_CAMAS_AISLAMIENTO_LEVES,
            CANTIDAD_CAMAS_GRAVES) %>%

    filter((GRUPO_ETARIO == "De 60 a 79 años" | GRUPO_ETARIO == "De 80 a 99 años")) %>%

    mutate( GRUPO_ETARIO = case_when(
          GRUPO_ETARIO == "De 60 a 79 años" ~ 'POBL_MAYOR',
          GRUPO_ETARIO == "De 80 a 99 años" ~ 'POBL_MAYOR'))
head(data_7 )
```

```
##       DISTRITO GRUPO_ETARIO CANT_HAB_X_GRUPO CANTIDAD_CAMAS_AISLAMIENTO_LEVES
## 1 Buenos Aires   POBL_MAYOR          2346634                              37337
## 2 Buenos Aires   POBL_MAYOR           495175                              37337
## 3         CABA   POBL_MAYOR           511513                              16173
## 4         CABA   POBL_MAYOR           145465                              16173
## 5    Catamarca   POBL_MAYOR            48828                               1444
## 6    Catamarca   POBL_MAYOR             9194                               1444
##   CANTIDAD_CAMAS_GRAVES
## 1                 57151
## 2                 57151
## 3                  5095
## 4                  5095
## 5                   539
## 6                   539
```

```r
#calculo la cantidad de camas necesitadas según leves y contagiados a NIVEL NACIONAL
data_7a <- data_7 %>% summarise(TOTAL_POBL_MAYOR = sum(CANT_HAB_X_GRUPO, na.rm = T),
                      CONTAGIADOS = ceiling(TOTAL_POBL_MAYOR*0.1),
                      NEC_CAMA_LEVE = ceiling(CONTAGIADOS*0.15),
                      NEC_CAMA_GRAVE = ceiling(CONTAGIADOS*0.05))
head(data_7a)
```

```
##   TOTAL_POBL_MAYOR CONTAGIADOS NEC_CAMA_LEVE NEC_CAMA_GRAVE
## 1          6811100      681110        102167          34056
```

7b. ¿Cuantas camas faltan (o sobran) en cada distrito para atender a esta demanda?

```r
#calculo la cantidad de camas necesitadas, faltantes y sobrantes según leves y contagiados
# a NIVEL PROVINCIAL

#la funcion ceiling nos redondea para arriba ya que no podemos tener 284180.9 contagiados,
# sino que tenemos 284181
data_7b <- data_7 %>% group_by(DISTRITO) %>%
            mutate(TOTAL_POBL_MAYOR = sum(CANT_HAB_X_GRUPO, na.rm = T),
                      CONTAGIADOS = ceiling(TOTAL_POBL_MAYOR*0.1),
                      NEC_CAMA_LEVE = ceiling(CONTAGIADOS*0.15),
                      NEC_CAMA_GRAVE = ceiling(CONTAGIADOS*0.05),
                      DIF_CAMAS_LEVE =  CANTIDAD_CAMAS_AISLAMIENTO_LEVES-NEC_CAMA_LEVE,
                      DIF_CAMAS_GRAVE =  CANTIDAD_CAMAS_GRAVES-NEC_CAMA_GRAVE) %>%
            select(DISTRITO, NEC_CAMA_LEVE, NEC_CAMA_GRAVE,
                   DIF_CAMAS_LEVE, DIF_CAMAS_GRAVE) %>% distinct()
data_7b #si da negativo faltan camas, si da positivo sobran camas
```

```
## # A tibble: 22 x 5
## # Groups:   DISTRITO [22]
##    DISTRITO     NEC_CAMA_LEVE NEC_CAMA_GRAVE DIF_CAMAS_LEVE DIF_CAMAS_GRAVE
##    <chr>                <dbl>          <dbl>          <dbl>           <dbl>
##  1 Buenos Aires         42628          14210          -5291           42941
##  2 CABA                  9855           3285           6318            1810
##  3 Catamarca              871            291            573             248
##  4 Chaco                 2200            734            614             687
##  5 Chubut                1214            405            656              52
##  6 Corrientes            2321            774            857            -191
##  7 Córdoba               9433           3145           -122           10215
##  8 Formosa               1174            392            736            -221
##  9 Jujuy                 1475            492            915             -95
## 10 La Pampa               938            313            -51            -109
## # ... with 12 more rows
```