05. - Data Collection_CU_04_17_indicadores_vacunacion

June 8, 2023

#

CU04_Optimización de vacunas

Citizenlab Data Science Methodology > II - Data Processing Domain *** > # 05.- Data Collection

Data Collection is the process to obtain and generate (if required) necessary data to model the problem.

0.0.1 16. Unión de datos globales por zonas de vacunación

- Consolidación de todas las variables por zona sin detalle de la semana del caso de uso
- Sanitarias, indicadores INE

Table of Contents

Settings

Data Load

ETL Processes

Import data from: CSV, Excel, Tab, JSON, SQL, and Parquet files

Synthetic Data Generation

Fake Data Generation

Open Data

Data Save

Main Conclusions

Main Actions

Acciones done

Acctions to perform

0.1 Settings

0.1.1 Encoding

Con la siguiente expresión se evitan problemas con el encoding al ejecutar el notebook. Es posible que deba ser eliminada o adaptada a la máquina en la que se ejecute el código.

[1]:
```
Sys.setlocale(category = "LC_ALL", locale = "es_ES.UTF-8")
```

'es_ES.UTF-8/es_ES.UTF-8/es_ES.UTF-8/C/es_ES.UTF-8/C'

0.1.2 Packages to use

- {tcltk} para selección interactiva de archivos locales
- {readxl} para leer archivos de Excel
- {readr} para leer y escribir archivos csv
- {dplyr} para explorar datos

[2]:
```
library(readxl)
library(readr)
library(dplyr)
```

```
Attaching package: 'dplyr'


The following objects are masked from 'package:stats':

    filter, lag


The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

0.1.3 Paths

[3]:
```
iPath <- "Data/Input/"
oPath <- "Data/Output/"
```

0.2 Data Load

If there are more than one input file, make as many sections as files to import.

Instrucciones - Los ficheros de entrada del proceso están siempre en Data/Input/.
- Si hay más de un fichero de entrada, se crean tantos objetos iFile_xx y file_data_xx como ficheros de entrada (xx número correlativo con dos dígitos, rellenar con ceros a la izquierda)

1. Datos de vacunación y capacidad

OPCION A: Seleccionar fichero en ventana para mayor comodidad

Data load using the {tcltk} package. Ucomment the line if not using this option

[4]:
```
# file_data_01 <- tcltk::tk_choose.files(multi = FALSE)
```

OPCION B: Especificar el nombre de archivo

[5]:
```
iFile_01 <- "Output.xlsx"
file_data_01 <- paste0(iPath, iFile_01)

if(file.exists(file_data_01)){
    cat("Se leerán datos del archivo: ", file_data_01)
} else{
    warning("Cuidado: el archivo no existe.")
}
```

```
Se leerán datos del archivo:  Data/Input/Output.xlsx
```

Data file to dataframe Usar la función adecuada según el formato de entrada (xlsx, csv, json, ...)

[6]:
```
data_01 <- read_excel(file_data_01)
```

Estructura de los datos:

[7]:
```
data_01 |> glimpse()
```

```
Rows: 20,868
Columns: 22
$ ID                  <chr> "1", "2", "3", "4", "5", "6",
"7", "8", "9", "10"…
$ CODBDT              <chr> "686213", "686213", "686213",
"686213", "686213",…
$ GEOCODIGO           <chr> "001", "001", "001", "001",
"001", "001", "001", …
$ DESBDT              <chr> "Abrantes", "Abrantes",
"Abrantes", "Abrantes", "…
$ semana              <dbl> 36, 37, 38, 39, 40, 41, 42, 43,
44, 45, 46, 47, 4…
$ ano                 <dbl> 2021, 2021, 2021, 2021, 2021,
2021, 2021, 2021, 2…
$ n_vacunas           <dbl> 0, 0, 0, 0, 0, 328, 344, 353,
371, 341, 389, 349,…
$ n_citas             <dbl> 0, 0, 0, 0, 0, 305, 327, 328,
341, 334, 373, 332,…
$ capacidad_zona      <dbl> 7742, 7654, 7425, 7189, 7566,
7417, 7638, 7711, 7…
$ prop_riesgo         <dbl> 0.1954797, 0.1906348, 0.1930880,
0.1918070, 0.185…
$ tasa_riesgo         <dbl> 0.1950852, 0.1870784, 0.1874322,
0.2031197, 0.188…
$ poblacion_total     <dbl> 29872, 29872, 29872, 29872,
29872, 29872, 29872, …
$ poblacion_mujeres   <dbl> 0.5345094, 0.5345094, 0.5345094,
```

```
0.5345094, 0.534…
$ poblacion_mayores    <dbl> 0.1781619, 0.1781619, 0.1781619,
0.1781619, 0.178…
$ poblacion_inmigrante <dbl> 0.2183993, 0.2183993, 0.2183993,
0.2183993, 0.218…
$ tasa_paro            <dbl> 0.1665607, 0.1665607, 0.1665607,
0.1665607, 0.166…
$ tasa_mayores         <dbl> 0.1644978, 0.1882310, 0.1940977,
0.1781833, 0.181…
$ temperatura          <lgl> NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, N…
$ humedad              <lgl> NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, N…
$ NO2                  <lgl> NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, N…
$ busquedas_gripe      <lgl> NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, N…
$ tuits_gripe          <dbl> 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 3…
```

Muestra de datos:

[8]:
```
data_01 |> slice_head(n = 5)
```

A tibble: 5 x 22

ID <chr>	CODBDT <chr>	GEOCODIGO <chr>	DESBDT <chr>	semana <dbl>	ano <dbl>	n_vacunas <dbl>	n_citas <dbl>	ca <c
1	686213	001	Abrantes	36	2021	0	0	77
2	686213	001	Abrantes	37	2021	0	0	76
3	686213	001	Abrantes	38	2021	0	0	74
4	686213	001	Abrantes	39	2021	0	0	71
5	686213	001	Abrantes	40	2021	0	0	75

2. Datos de indicadores

OPCION A: Seleccionar fichero en ventana para mayor comodidad

Data load using the {tcltk} package. Ucomment the line if not using this option

[9]:
```
# file_data_02 <- tcltk::tk_choose.files(multi = FALSE)
```

OPCION B: Especificar el nombre de archivo

[10]:
```
iFile_02 <- "CU_04_05_05_02_indicadores_zonas.csv"
file_data_02 <- paste0(iPath, iFile_02)

if(file.exists(file_data_02)){
    cat("Se leerán datos del archivo: ", file_data_02)
} else{
    warning("Cuidado: el archivo no existe.")
}
```

```
Se leerán datos del archivo:  Data/Input/CU_04_05_05_02_indicadores_zonas.csv
```

Data file to dataframe Usar la función adecuada según el formato de entrada (xlsx, csv, json, …)

[11]:
```
data_02 <- read_csv(file_data_02)
```

```
Rows: 286 Columns: 20
  Column specification

Delimiter: ","
chr  (2): id_zona, nombre_zona
dbl (18): nsec, t3_1, t1_1, t2_1, t2_2, t4_1, t4_2, t4_3, t5_1, t6_1,
t7_1, …

 Use `spec()` to retrieve the full column specification for this
data.
 Specify the column types or set `show_col_types = FALSE` to quiet
this message.
```

Estructura de los datos:

[12]:
```
data_02 |> glimpse()
```

```
Rows: 286
Columns: 20
$ id_zona         <chr> "001", "002", "003", "004", "005",
"006", "007", "008"…
$ nombre_zona     <chr> "Abrantes", "Acacias", "Adelfas",
"Alameda", "Alameda …
$ nsec            <dbl> 23, 11, 19, 18, 18, 15, 10, 17, 11,
12, 14, 17, 8, 20,…
$ t3_1            <dbl> 42.31249, 46.15717, 45.30472,
43.70575, 43.31968, 44.2…
$ t1_1            <dbl> 29872, 17961, 28933, 21393, 27259,
21186, 12367, 27448…
$ t2_1            <dbl> 0.5345094, 0.5328775, 0.5383134,
0.4919805, 0.5153616,…
$ t2_2            <dbl> 0.4654906, 0.4671225, 0.4616866,
0.5080195, 0.4846384,…
$ t4_1            <dbl> 0.15619346, 0.10929873, 0.12408817,
0.07409008, 0.1742…
$ t4_2            <dbl> 0.6656459, 0.6516215, 0.6541986,
0.7685363, 0.6063697,…
$ t4_3            <dbl> 0.17816190, 0.23909411, 0.22173279,
0.15740288, 0.2193…
$ t5_1            <dbl> 0.21839930, 0.04313645, 0.07236948,
0.26145728, 0.0705…
$ t6_1            <dbl> 0.34508551, 0.07700825, 0.12420792,
```

```
0.35309699, 0.1280…
$ t7_1            <dbl> 0.04090796, 0.08494593, 0.07195008,
0.04949630, 0.0715…
$ t8_1            <dbl> 0.02880326, 0.07576304, 0.06532400,
0.04323113, 0.0626…
$ t9_1            <dbl> 0.2685716, 0.6368488, 0.6031800,
0.5672350, 0.5862647,…
$ t10_1           <dbl> 0.16656066, 0.08377987, 0.08766432,
0.12302998, 0.0851…
$ t11_1           <dbl> 0.4723181, 0.5083022, 0.5262385,
0.5410470, 0.5028547,…
$ t12_1           <dbl> 0.5637381, 0.5548573, 0.5770966,
0.6171510, 0.5490471,…
$ area            <dbl> 1571618.8, 771569.7, 854805.6,
547596.1, 35137989.9, 6…
$ densidad_hab_km <dbl> 19007.1534, 23278.5190, 33847.4619,
39067.1135, 775.77…
```

Muestra de datos:

[13]:
```
data_02 |> slice_head(n = 5)
```

A spec_tbl_df: 5 x 20

id_zona <chr>	nombre_zona <chr>	nsec <dbl>	t3_1 <dbl>	t1_1 <dbl>	t2_1 <dbl>	t2_2 <dbl>	t4_ <d
001	Abrantes	23	42.31249	29872	0.5345094	0.4654906	0.1
002	Acacias	11	46.15717	17961	0.5328775	0.4671225	0.1
003	Adelfas	19	45.30472	28933	0.5383134	0.4616866	0.1
004	Alameda	18	43.70575	21393	0.4919805	0.5080195	0.0
005	Alameda de Osuna	18	43.31968	27259	0.5153616	0.4846384	0.1

0.3 ETL Processes

0.3.1 Import data from: CSV, Excel, Tab, JSON, SQL, and Parquet files

Se han importado en el apartado Data Load anterior:

- Datos simulados sanitarios
- Indicadores demográficos

Incluir apartados si procede para: Extracción de datos (select, filter), Transformación de datos, (mutate, joins, …). Si es necesario tratar datos perdidos, indicarlo también en NB 09.2

Si no aplica: Estos datos no requieren tareas de este tipo.

Data extract

- Seleccionar columnas de la simulación que deberían ser solo por zona

[14]:
```
edata_01 <- data_01 |>
    select(GEOCODIGO, DESBDT, ano, semana,
```

```
        capacidad_zona, prop_riesgo, tasa_riesgo, tasa_mayores,␣
 ↪poblacion_mayores)
```

[15]:
```
glimpse(edata_01)
```

```
Rows: 20,868
Columns: 9
$ GEOCODIGO         <chr> "001", "001", "001", "001", "001",
"001", "001", "00…
$ DESBDT            <chr> "Abrantes", "Abrantes", "Abrantes",
"Abrantes", "Abr…
$ ano               <dbl> 2021, 2021, 2021, 2021, 2021, 2021,
2021, 2021, 2021…
$ semana            <dbl> 36, 37, 38, 39, 40, 41, 42, 43, 44,
45, 46, 47, 48, …
$ capacidad_zona    <dbl> 7742, 7654, 7425, 7189, 7566, 7417,
7638, 7711, 7249…
$ prop_riesgo       <dbl> 0.1954797, 0.1906348, 0.1930880,
0.1918070, 0.185116…
$ tasa_riesgo       <dbl> 0.1950852, 0.1870784, 0.1874322,
0.2031197, 0.188523…
$ tasa_mayores      <dbl> 0.1644978, 0.1882310, 0.1940977,
0.1781833, 0.181451…
$ poblacion_mayores <dbl> 0.1781619, 0.1781619, 0.1781619,
0.1781619, 0.178161…
```

Data transform

- Agregación de datos por zona y transformar proporciones a tasas

[16]:
```
set.seed(1)
tdata_01 <- edata_01 |>
    select(-ano, -semana) |>
    group_by(GEOCODIGO, DESBDT) |>
    summarise(across(everything(), ~mean(.x, na.rm = TRUE)))|>
    ungroup() |>
    mutate(tasa_riesgo = rnorm(n(), 0.01, 0.005),
        tasa_mayores = rnorm(n(), 0.02, 0.01),
        capacidad_zona = round(capacidad_zona))
```

```
`summarise()` has grouped output by 'GEOCODIGO'. You can override using
the
`.groups` argument.
```

[17]:
```
tdata_01 |> glimpse()
```

```
Rows: 282
Columns: 7
```

```
$ GEOCODIGO         <chr> "001", "002", "003", "004", "005",
"006", "007", "00…
$ DESBDT            <chr> "Abrantes", "Acacias", "Adelfas",
"Alameda", "Alamed…
$ capacidad_zona    <dbl> 7477, 4482, 7235, 5343, 6816, 5318,
3081, 6843, 3776…
$ prop_riesgo       <dbl> 0.19535423, 0.26312125, 0.24362661,
0.17253854, 0.24…
$ tasa_riesgo       <dbl> 6.867731e-03, 1.091822e-02,
5.821857e-03, 1.797640e-…
$ tasa_mayores      <dbl> 0.033242586, 0.012987683,
0.014193857, 0.009989278, …
$ poblacion_mayores <dbl> 0.17816190, 0.23909411, 0.22173279,
0.15740288, 0.21…
```

- Unión de tablas por zona

[18]:
```
data <- tdata_01 |>
    full_join(data_02, by = c("GEOCODIGO" = "id_zona"))
```

[19]:
```
glimpse(data)
```

```
Rows: 286
Columns: 26
$ GEOCODIGO         <chr> "001", "002", "003", "004", "005",
"006", "007", "00…
$ DESBDT            <chr> "Abrantes", "Acacias", "Adelfas",
"Alameda", "Alamed…
$ capacidad_zona    <dbl> 7477, 4482, 7235, 5343, 6816, 5318,
3081, 6843, 3776…
$ prop_riesgo       <dbl> 0.19535423, 0.26312125, 0.24362661,
0.17253854, 0.24…
$ tasa_riesgo       <dbl> 6.867731e-03, 1.091822e-02,
5.821857e-03, 1.797640e-…
$ tasa_mayores      <dbl> 0.033242586, 0.012987683,
0.014193857, 0.009989278, …
$ poblacion_mayores <dbl> 0.17816190, 0.23909411, 0.22173279,
0.15740288, 0.21…
$ nombre_zona       <chr> "Abrantes", "Acacias", "Adelfas",
"Alameda", "Alamed…
$ nsec              <dbl> 23, 11, 19, 18, 18, 15, 10, 17, 11,
12, 14, 17, 8, 2…
$ t3_1              <dbl> 42.31249, 46.15717, 45.30472,
43.70575, 43.31968, 44…
$ t1_1              <dbl> 29872, 17961, 28933, 21393, 27259,
21186, 12367, 274…
$ t2_1              <dbl> 0.5345094, 0.5328775, 0.5383134,
0.4919805, 0.515361…
```

```
$ t2_2              <dbl> 0.4654906, 0.4671225, 0.4616866,
0.5080195, 0.484638…
$ t4_1              <dbl> 0.15619346, 0.10929873, 0.12408817,
0.07409008, 0.17…
$ t4_2              <dbl> 0.6656459, 0.6516215, 0.6541986,
0.7685363, 0.606369…
$ t4_3              <dbl> 0.17816190, 0.23909411, 0.22173279,
0.15740288, 0.21…
$ t5_1              <dbl> 0.21839930, 0.04313645, 0.07236948,
0.26145728, 0.07…
$ t6_1              <dbl> 0.34508551, 0.07700825, 0.12420792,
0.35309699, 0.12…
$ t7_1              <dbl> 0.04090796, 0.08494593, 0.07195008,
0.04949630, 0.07…
$ t8_1              <dbl> 0.02880326, 0.07576304, 0.06532400,
0.04323113, 0.06…
$ t9_1              <dbl> 0.2685716, 0.6368488, 0.6031800,
0.5672350, 0.586264…
$ t10_1             <dbl> 0.16656066, 0.08377987, 0.08766432,
0.12302998, 0.08…
$ t11_1             <dbl> 0.4723181, 0.5083022, 0.5262385,
0.5410470, 0.502854…
$ t12_1             <dbl> 0.5637381, 0.5548573, 0.5770966,
0.6171510, 0.549047…
$ area              <dbl> 1571618.8, 771569.7, 854805.6,
547596.1, 35137989.9,…
$ densidad_hab_km   <dbl> 19007.1534, 23278.5190, 33847.4619,
39067.1135, 775.…
```

0.4 Synthetic Data Generation

- Algunos de los datos se habían generado de forma sintética
- En esta tarea se han vuelto a generar las tasas

0.5 Fake Data Generation

No aplica

0.6 Open Data

Los indicadores provienen de datos abiertos del INE. Las superficies y densidades de población se calcularon con las geometrías que venían también de fuentes públicas.

0.7 Data Save

Este proceso, puede copiarse y repetirse en aquellas partes del notebbok que necesiten guardar datos. Recuerde cambiar las cadenas añadida del fichero para diferenciarlas

Identificamos los datos a guardar

[20]:
```
data_to_save <- data
```

Estructura de nombre de archivos:

- Código del caso de uso, por ejemplo “CU_04”
- Número del proceso que lo genera, por ejemplo ”_05”.
- Número de la tarea que lo genera, por ejemplo ”_01”
- En caso de generarse varios ficheros en la misma tarea, llevarán _01 _02 … después
- Nombre: identificativo de “properData”, por ejemplo ”_zonasgeo”
- Extensión del archivo

Ejemplo: ”CU_04_05_01_01_zonasgeo.json, primer fichero que se genera en la tarea 01 del proceso 05 (Data Collection) para el caso de uso 04 (vacunas)

Importante mantener los guiones bajos antes de proceso, tarea, archivo y nombre

0.7.1 Proceso 05

[21]:
```
caso <- "CU_04"
proceso <- '_05'
tarea <- "_17"
archivo <- ""
proper <- "_indicadores_vacunacion"
extension <- ".csv"
```

OPCION A: Uso del paquete “tcltk” para mayor comodidad

- Buscar carpeta, escribir nombre de archivo SIN extensión (se especifica en el código)
- Especificar sufijo2 si es necesario
- Cambiar datos por datos_xx si es necesario

[22]:
```
# file_save <- paste0(caso, proceso, tarea, tcltk::tkgetSaveFile(), proper,␣
 ↪extension)
# path_out <- paste0(oPath, file_save)
# write_csv(data_to_save, path_out)

# cat('File saved as: ')
# path_out
```

OPCION B: Especificar el nombre de archivo

- Los ficheros de salida del proceso van siempre a Data/Output/.

[23]:
```
file_save <- paste0(caso, proceso, tarea, archivo, proper, extension)
path_out <- paste0(oPath, file_save)
write_csv(data_to_save, path_out)

cat('File saved as: ')
path_out
```

```
File saved as:
```

'Data/Output/CU_04_05_17_indicadores_vacunacion.csv'

Copia del fichero a Input Si el archivo se va a usar en otros notebooks, copiar a la carpeta Input

[24]:
```
path_in <- paste0(iPath, file_save)
file.copy(path_out, path_in, overwrite = TRUE)
```

TRUE

0.8 Main Conclusions

List and describe the general conclusions of the analysis carried out.

0.8.1 Prerequisites

Para que funcione este código se necesita:

- Las rutas de archivos `Data/Input` y `Data/Output` deben existir (relativas a la ruta del *notebook*)
- El paquete tcltk instalado para seleccionar archivos interactivamente. No se necesita en producción.
- Los paquetes tcltk, readxl, readr, dplyr deben estar instalados.

0.8.2 Configuration Management

This notebook has been tested with the following versions of R and packages. It cannot be assured that later versions work in the same way: * R 4.2.2 * tcltk 4.2.2 * readxl 1.4.1 * readr 2.1.3 * dplyr 1.0.10

0.8.3 Data structures

Objeto `data`

- Los datos de origen
- Hay 21736 filas con la información de las siguientes variables:
  - GEOCODIGO
  - DESBDT
  - ano
  - semana
  - n_vacunas
  - n_citas
  - tmed
  - prec
  - velmedia
  - presMax
  - benzene
  - co
  - no
  - no2

- nox
- o3
- pm10
- pm2.5
- so2

Observaciones generales sobre los datos

- Había más datos de vacunación que de las otras variables porque las primeras se simularon de las semanas exactas de la campaña, y el resto se obtuvieron de meses completos
- Se han mantenido todas las filas en este conjunto de datos por si son útiles para los modelos
- Los datos son únicos para cada zona, año y semana

0.8.4 Consideraciones para despliegue en piloto

- Ninguna

0.8.5 Consideraciones para despliegue en producción

- Se deben crear los procesos ETL en producción necesarios para que los datos de entrada estén actualizados

0.9 Main Actions

Acciones done Indicate the actions that have been carried out in this process

- Se han unido las tablas con datos por semana y zona

Acctions to perform Indicate the actions that must be carried out in subsequent processes

- Se deben unir a las tablas con otra granularidad, repitiendo datos, para tener un csv que incluya todo

0.10 CODE TO DEPLOY (PILOT)

A continuación se incluirá el código que deba ser llevado a despliegue para producción, dado que se entiende efectúa operaciones necesarias sobre los datos en la ejecución del prototipo

Description

- No hay nada que desplegar en el piloto, ya que estos datos son estáticos o en todo caso cambian con muy poca frecuencia, altamente improbable durante el proyecto.

CODE

[25]:
```
# incluir código
```