

05. - Data Collection_CU_18_09_adm_distrito_v_01

June 13, 2023

#

CU18_Infraestructuras_eventos

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 05. Agrupar datos de administración y empresas por distrito

- A partir de los datos puntuales, agregar por distrito censal y contar infraestructuras para hacer mapas de regiones.
- Adicionalmente crear metadatos con descripción y agrupamiento de variables.

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

Accions to perform

0.1 Settings

0.1.1 Packages to use

- {tcltk} para selección interactiva de archivos locales
- {readr} para leer y escribir archivos csv

- {dplyr} para explorar datos
- {dityr} para transformar datos
- {janitor} para limpiar datos

```
[1]: library(readr)
library(dplyr)
library(tidyr)
library(janitor)
```

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

Attaching package: 'janitor'

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

chisq.test, fisher.test

0.1.2 Paths

```
[2]: 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)

OPCION A: Seleccionar fichero en ventana para mayor comodidad

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

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

OPCION B: Especificar el nombre de archivo

```
[3]: iFile <- "CU_18_05_08_adm_distritos.csv"
file_data <- paste0(iPath, iFile)

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

Se leerán datos del archivo: Data/Input/CU_18_05_08_adm_distritos.csv

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

```
[4]: data <- read_csv(file_data)
```

Rows: 120280 Columns: 7
Column specification

Delimiter: ","

chr (5): Grupo, Tipo, ETIQUETA, CMUN, CDIS

dbl (2): X, Y

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:

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

Rows: 120,280

Columns: 7

\$ Grupo <chr> "Administración pública", "Administración pública", "Administ...

\$ Tipo <chr> "Agencia Tributaria", "Agencia Tributaria", "Agencia Tributaria"

\$ ETIQUETA <chr> "Administración de la Agencia Tributaria Oficina Central", "A...

\$ X <dbl> -3.698678, -3.712513, -3.745598, -3.690993, -3.672477, -3.710...

\$ Y <dbl> 40.45554, 40.44553, 40.37049, 40.41795, 40.45483, 40.48316, 4...

\$ CMUN <chr> "079", "079", "079", "079", "079", "079",

```
"079", "079", "079"...
$ CDIS      <chr> "06", "07", "11", "03", "05", "08", "04",
"14", "20", "12", "...
```

Muestra de datos:

```
[6]: slice_head(data, n = 5)
```

	Grupo <chr>	Tipo <chr>	ETIQUETA <chr>
A spec_tbl_df: 5 × 7	Administración pública	Agencia Tributaria	Administración de la Agencia Tributaria O
	Administración pública	Agencia Tributaria	Administración de la Agencia Tributaria G
	Administración pública	Agencia Tributaria	Administración de la Agencia Tributaria S
	Administración pública	Agencia Tributaria	Administración de la Agencia Tributaria M
	Administración pública	Agencia Tributaria	Administración de la Agencia Tributaria C

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:

- Infraestructuras

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

Data Transform

- Contar número de infraestructuras de cada tipo por distrito
- Extender en columnas para caracterizar distritos

```
[7]: tdata_01 <- data |>
      count(CMUN, CDIS, Tipo) |>
      pivot_wider(names_from = Tipo,
                  values_from = n,
                  values_fill = 0)
```

```
[8]: glimpse(tdata_01)
```

```
Rows: 246
Columns: 49
$ CMUN
<chr> "001", "...
$ CDIS
<chr> "01", "0...
$ `Ayuntamientos, Consejerías, Ministerios, etc.`
<int> 1, 1, 1,...
$ `Sanidad y servicios sociales`
<int> 1, 8, 0,...
$ `Actividades administrativas`
```

```

<int> 0, 13, 0...
$ `Actividades asociativas y hogares`
<int> 0, 2, 0,...
$ `Actividades profesionales`
<int> 0, 11, 1...
$ `Administraciones públicas`
<int> 0, 2, 2,...
$ Alimentación
<int> 0, 2, 0,...
$ `Centros educativos no universitarios - Centros públicos`
<int> 0, 3, 0,...
$ `Comercio mayorista`
<int> 0, 34, 0...
$ `Comercio minorista`
<int> 0, 22, 0...
$ Construcción
<int> 0, 22, 0...
$ Educación
<int> 0, 4, 0,...
$ Hostelería
<int> 0, 2, 4,...
$ `Industria química y refino`
<int> 0, 3, 0,...
$ `Información y comunicaciones`
<int> 0, 1, 0,...
$ `Maquinaria industrial`
<int> 0, 6, 0,...
$ `Material de transporte`
<int> 0, 4, 0,...
$ `Material eléctrico y electrónico`
<int> 0, 3, 0,...
$ Mercadillos
<int> 0, 1, 0,...
$ `Metálicas básicas e intermedias`
<int> 0, 25, 0...
$ `Minería y suministros`
<int> 0, 1, 0,...
$ `Otras manufactureras`
<int> 0, 14, 0...
$ `Otros servicios - Bancos`
<int> 0, 3, 0,...
$ `Papel y gráficas`
<int> 0, 14, 0...
$ `Servicios financieros`
<int> 0, 3, 0,...
$ `Servicios personales`
<int> 0, 3, 0,...
$ `Servicios recreativos`

```

```

<int> 0, 5, 0,...
$ `Textil, confección y calzado`
<int> 0, 3, 0,...
$ `Transporte y almacenamiento`
<int> 0, 8, 0,...
$ `Actividades Inmobiliarias`
<int> 0, 0, 0,...
$ `Centros educativos no universitarios - Centros privados`
<int> 0, 0, 0,...
$ `Industria no metálica`
<int> 0, 0, 0,...
$ `Administración de Justicia`
<int> 0, 0, 0,...
$ `Agencia Tributaria`
<int> 0, 0, 0,...
$ `Campus universitarios`
<int> 0, 0, 0,...
$ `Centros educativos universitarios`
<int> 0, 0, 0,...
$ `Colegios mayores`
<int> 0, 0, 0,...
$ `Galerías de alimentación`
<int> 0, 0, 0,...
$ `Grandes superficies especializadas`
<int> 0, 0, 0,...
$ Hipermercados
<int> 0, 0, 0,...
$ `Mercados de abastos`
<int> 0, 0, 0,...
$ `Agricultura y ganadería`
<int> 0, 0, 0,...
$ `Centros educativos no universitarios - Servicios educativos`
<int> 0, 0, 0,...
$ `Seguridad Social`
<int> 0, 0, 0,...
$ `Centros comerciales`
<int> 0, 0, 0,...
$ `Oficinas de empleo`
<int> 0, 0, 0,...
$ `Embajadas y consulados`
<int> 0, 0, 0,...

```

```
[9]: tdata_01 |> slice_head(n = 5)
```

	CMUN <chr>	CDIS <chr>	Ayuntamientos, Consejerías, Ministerios, etc. <int>	Sanidad y servicios sociales <int>
A tibble: 5 × 49	001	01	1	1
	002	01	1	8
	003	01	1	0
	004	01	1	10
	005	01	1	42

Data Extract

- Extraer nombres de columnas

```
[44]: tdata_02 <- data.frame(desc_var = colnames(tdata_01)[3:ncol(tdata_01)])
```

```
[45]: tdata_02
```

desc_var

<chr>

ayuntamientos_consejerias_ministerios_etc
sanidad_y_servicios_sociales
actividades_administrativas
actividades_asociativas_y_hogares
actividades_profesionales
administraciones_publicas
alimentacion
centros_educativos_no_universitarios_centros_publicos
comercio_mayorista
comercio_minorista
construccion
educacion
hosteleria
industria_quimica_y_refino
informacion_y_comunicaciones
maquinaria_industrial
material_de_transporte
material_electrico_y_electronico
mercadillos
metalicas_basicas_e_intermedias
mineria_y_suministros
otras_manufactureras
otros_servicios_bancos
papel_y_graficas
servicios_financieros
servicios_personales
servicios_recreativos
textil_confeccion_y_calzado
transporte_y_almacenamiento
actividades_inmobiliarias
centros_educativos_no_universitarios_centros_privados
industria_no_metalica
administracion_de_justicia
agencia_tributaria
campus_universitarios
centros_educativos_universitarios
colegios_mayores
galerias_de_alimentacion
grandes_superficies_especializadas
hipermercados
mercados_de_abastos
agricultura_y_ganaderia
centros_educativos_no_universitarios_servicios_educativos
seguridad_social
centros_comerciales
oficinas_de_empleo
embajadas_y_consulados

A data.frame: 47 × 1

Data Transform

- Limpiar nombres de columnas

```
[12]: tdata_01 <- tdata_01 |>  
      clean_names()
```

```
[13]: glimpse(tdata_01)
```

```
Rows: 246  
Columns: 49  
$ cmun  
<chr> "001", "002"...  
$ cdis  
<chr> "01", "01", ...  
$ ayuntamientos_consejerias_ministerios_etc  
<int> 1, 1, 1, 1, ...  
$ sanidad_y_servicios_sociales  
<int> 1, 8, 0, 10,...  
$ actividades_administrativas  
<int> 0, 13, 0, 2,...  
$ actividades_asociativas_y_hogares  
<int> 0, 2, 0, 2, ...  
$ actividades_profesionales  
<int> 0, 11, 1, 3,...  
$ administraciones_publicas  
<int> 0, 2, 2, 2, ...  
$ alimentacion  
<int> 0, 2, 0, 0, ...  
$ centros_educativos_no_universitarios_centros_publicos  
<int> 0, 3, 0, 4, ...  
$ comercio_mayorista  
<int> 0, 34, 0, 3,...  
$ comercio_minorista  
<int> 0, 22, 0, 10...  
$ construccion  
<int> 0, 22, 0, 16...  
$ educacion  
<int> 0, 4, 0, 6, ...  
$ hosteleria  
<int> 0, 2, 4, 12,...  
$ industria_quimica_y_refino  
<int> 0, 3, 0, 0, ...  
$ informacion_y_comunicaciones  
<int> 0, 1, 0, 0, ...  
$ maquinaria_industrial  
<int> 0, 6, 0, 0, ...  
$ material_de_transporte  
<int> 0, 4, 0, 0, ...
```

```

$ material_electrico_y_electronico
<int> 0, 3, 0, 0, ...
$ mercadillos
<int> 0, 1, 0, 1, ...
$ metalicas_basicas_e_intermedias
<int> 0, 25, 0, 1,...
$ mineria_y_suministros
<int> 0, 1, 0, 0, ...
$ otras_manufactureras
<int> 0, 14, 0, 2,...
$ otros_servicios_bancos
<int> 0, 3, 0, 2, ...
$ papel_y_graficas
<int> 0, 14, 0, 0,...
$ servicios_financieros
<int> 0, 3, 0, 4, ...
$ servicios_personales
<int> 0, 3, 0, 5, ...
$ servicios_recreativos
<int> 0, 5, 0, 4, ...
$ textil_confeccion_y_calzado
<int> 0, 3, 0, 0, ...
$ transporte_y_almacenamiento
<int> 0, 8, 0, 4, ...
$ actividades_inmobiliarias
<int> 0, 0, 0, 1, ...
$ centros_educativos_no_universitarios_centros_privados
<int> 0, 0, 0, 2, ...
$ industria_no_metalica
<int> 0, 0, 0, 1, ...
$ administracion_de_justicia
<int> 0, 0, 0, 0, ...
$ agencia_tributaria
<int> 0, 0, 0, 0, ...
$ campus_universitarios
<int> 0, 0, 0, 0, ...
$ centros_educativos_universitarios
<int> 0, 0, 0, 0, ...
$ colegios_mayores
<int> 0, 0, 0, 0, ...
$ galerias_de_alimentacion
<int> 0, 0, 0, 0, ...
$ grandes_superficies_especializadas
<int> 0, 0, 0, 0, ...
$ hipermercados
<int> 0, 0, 0, 0, ...
$ mercados_de_abastos
<int> 0, 0, 0, 0, ...

```

```

$ agricultura_y_ganaderia
<int> 0, 0, 0, 0, ...
$ centros_educativos_no_universitarios_servicios_educativos
<int> 0, 0, 0, 0, ...
$ seguridad_social
<int> 0, 0, 0, 0, ...
$ centros_comerciales
<int> 0, 0, 0, 0, ...
$ oficinas_de_empleo
<int> 0, 0, 0, 0, ...
$ embajadas_y_consulados
<int> 0, 0, 0, 0, ...

```

- Completar metadatos

```

[46]: tdata_02 <- tdata_02 |>
      mutate(nombre_var = colnames(tdata_01)[3:ncol(tdata_01)]) |>
      left_join(data |> count(Grupo, Tipo),
                by = c("desc_var" = "Tipo"))

```

```

[47]: glimpse(tdata_02)

```

```

Rows: 47
Columns: 4
$ desc_var    <chr>
"ayuntamientos_consejerias_ministerios_etc", "sanidad_y_ser...
$ nombre_var  <chr>
"ayuntamientos_consejerias_ministerios_etc", "sanidad_y_ser...
$ Grupo       <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA,...
$ n           <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA,...

```

```

[48]: tdata_02

```

desc_var <chr>	nombre_var <chr>
ayuntamientos_consejerias_ministerios_etc	ayuntamientos_consejerias_ministerios_etc
sanidad_y_servicios_sociales	sanidad_y_servicios_sociales
actividades_administrativas	actividades_administrativas
actividades_asociativas_y_hogares	actividades_asociativas_y_hogares
actividades_profesionales	actividades_profesionales
administraciones_publicas	administraciones_publicas
alimentacion	alimentacion
centros_educativos_no_universitarios_centros_publicos	centros_educativos_no_universitarios_centros_publicos
comercio_mayorista	comercio_mayorista
comercio_minorista	comercio_minorista
construccion	construccion
educacion	educacion
hosteleria	hosteleria
industria_quimica_y_refino	industria_quimica_y_refino
informacion_y_comunicaciones	informacion_y_comunicaciones
maquinaria_industrial	maquinaria_industrial
material_de_transporte	material_de_transporte
material_electrico_y_electronico	material_electrico_y_electronico
mercadillos	mercadillos
metalicas_basicas_e_intermedias	metalicas_basicas_e_intermedias
mineria_y_suministros	mineria_y_suministros
otras_manufactureras	otras_manufactureras
otros_servicios_bancos	otros_servicios_bancos
papel_y_graficas	papel_y_graficas
servicios_financieros	servicios_financieros
servicios_personales	servicios_personales
servicios_recreativos	servicios_recreativos
textil_confeccion_y_calzado	textil_confeccion_y_calzado
transporte_y_almacenamiento	transporte_y_almacenamiento
actividades_inmobiliarias	actividades_inmobiliarias
centros_educativos_no_universitarios_centros_privados	centros_educativos_no_universitarios_centros_privados
industria_no_metalica	industria_no_metalica
administracion_de_justicia	administracion_de_justicia
agencia_tributaria	agencia_tributaria
campus_universitarios	campus_universitarios
centros_educativos_universitarios	centros_educativos_universitarios
colegios_mayores	colegios_mayores
galerias_de_alimentacion	galerias_de_alimentacion
grandes_superficies_especializadas	grandes_superficies_especializadas
hipermercados	hipermercados
mercados_de_abastos	mercados_de_abastos
agricultura_y_ganaderia	agricultura_y_ganaderia
centros_educativos_no_universitarios_servicios_educativos	centros_educativos_no_universitarios_servicios_educativos
seguridad_social	seguridad_social
centros_comerciales	centros_comerciales
oficinas_de_empleo	oficinas_de_empleo
embajadas_y_consulados	embajadas_y_consulados

A data.frame: 47 × 4

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

0.4 Synthetic Data Generation

Estos datos no requieren tareas de este tipo.

0.5 Fake Data Generation

Estos datos no requieren tareas de este tipo.

0.6 Open Data

Estos datos no requieren tareas de este tipo.

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

1. Infraestructuras agregadas por distrito

```
[26]: data_to_save <- tdata_01
```

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

```
[30]: caso <- "CU_18"  
proceso <- '_05'  
tarea <- "_09"  
archivo <- "_01"  
proper <- "_distritos_adm"  
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

```
[ ]: # file_save_01 <- paste0(caso, proceso, tarea, tcltk::tkgetSaveFile(), proper,
    ↪extension)
# path_out_01 <- paste0(oPath, file_save_01)
# write_csv(data_to_save, path_out_01)

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

OPCION B: Especificar el nombre de archivo

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

```
[31]: file_save_01 <- paste0(caso, proceso, tarea, archivo, proper, extension)
path_out_01 <- paste0(oPath, file_save_01)
write_csv(data_to_save, path_out_01)

cat('File saved as: ')
path_out_01
```

File saved as:

'Data/Output/CU_18_05_09_01_distritos_adm.csv'

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

```
[32]: path_in_01 <- paste0(iPath, file_save_01)
file.copy(path_out_01, path_in_01, overwrite = TRUE)
```

TRUE

2. Metadatos de infraestructuras

```
[49]: data_to_save_02 <- tdata_02
```

```
[50]: archivo <- "_02"
proper <- "_admin_meta"
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

```
[ ]: # file_save_02 <- paste0(caso, proceso, tarea, tcltk::tkgetSaveFile(), proper,
    ↪extension)
# path_out_02 <- paste0(oPath, file_save_02)
# write_csv(data_to_save_02, path_out_02)

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

```
# path_out
```

OPCION B: Especificar el nombre de archivo

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

```
[51]: file_save_02 <- paste0(caso, proceso, tarea, archivo, proper, extension)
      path_out_02 <- paste0(oPath, file_save_02)
      write_csv(data_to_save_02, path_out_02)

      cat('File saved as: ')
      path_out_02
```

File saved as:

'Data/Output/CU_18_05_09_02_admin_meta.csv'

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

```
[38]: path_in_02 <- paste0(iPath, file_save_02)
      file.copy(path_out_02, path_in_02, overwrite = TRUE)
```

TRUE

0.8 Main Conclusions

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

0.8.1 Prerequisites

This working code needs the following conditions:

- For using the interactive selection of file, the {tcltk} package must be installed. It is not needed in production.
- The {readr}, {dplyr}, {tidyr} and {janitor} packages must be installed.
- The data paths Data/Input and Data/Output must exist (relative to the notebook path)

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 * tidyr 1.3.0 * dplyr 1.0.10 * janitor 2.1.0 * readr 2.1.3

0.8.3 Data structures

Objeto tdata_01

- Tenemos 246 filas, una por distrito, con los recuentos de cada uno de los tipos de infraestructuras (12)

```
[41]: glimpse(tdata_01)
```

```

Rows: 246
Columns: 49
$ cmun
<chr> "001", "002"...
$ cdis
<chr> "01", "01", ...
$ ayuntamientos_consejerias_ministerios_etc
<int> 1, 1, 1, 1, ...
$ sanidad_y_servicios_sociales
<int> 1, 8, 0, 10,...
$ actividades_administrativas
<int> 0, 13, 0, 2,...
$ actividades_asociativas_y_hogares
<int> 0, 2, 0, 2, ...
$ actividades_profesionales
<int> 0, 11, 1, 3,...
$ administraciones_publicas
<int> 0, 2, 2, 2, ...
$ alimentacion
<int> 0, 2, 0, 0, ...
$ centros_educativos_no_universitarios_centros_publicos
<int> 0, 3, 0, 4, ...
$ comercio_mayorista
<int> 0, 34, 0, 3,...
$ comercio_minorista
<int> 0, 22, 0, 10...
$ construccion
<int> 0, 22, 0, 16...
$ educacion
<int> 0, 4, 0, 6, ...
$ hosteleria
<int> 0, 2, 4, 12,...
$ industria_quimica_y_refino
<int> 0, 3, 0, 0, ...
$ informacion_y_comunicaciones
<int> 0, 1, 0, 0, ...
$ maquinaria_industrial
<int> 0, 6, 0, 0, ...
$ material_de_transporte
<int> 0, 4, 0, 0, ...
$ material_electrico_y_electronico
<int> 0, 3, 0, 0, ...
$ mercadillos
<int> 0, 1, 0, 1, ...
$ metalicas_basicas_e_intermedias
<int> 0, 25, 0, 1,...
$ mineria_y_suministros
<int> 0, 1, 0, 0, ...

```



```

$ otras_manufactureras
<int> 0, 14, 0, 2,...
$ otros_servicios_bancos
<int> 0, 3, 0, 2, ...
$ papel_y_graficas
<int> 0, 14, 0, 0,...
$ servicios_financieros
<int> 0, 3, 0, 4, ...
$ servicios_personales
<int> 0, 3, 0, 5, ...
$ servicios_recreativos
<int> 0, 5, 0, 4, ...
$ textil_confeccion_y_calzado
<int> 0, 3, 0, 0, ...
$ transporte_y_almacenamiento
<int> 0, 8, 0, 4, ...
$ actividades_inmobiliarias
<int> 0, 0, 0, 1, ...
$ centros_educativos_no_universitarios_centros_privados
<int> 0, 0, 0, 2, ...
$ industria_no_metalica
<int> 0, 0, 0, 1, ...
$ administracion_de_justicia
<int> 0, 0, 0, 0, ...
$ agencia_tributaria
<int> 0, 0, 0, 0, ...
$ campus_universitarios
<int> 0, 0, 0, 0, ...
$ centros_educativos_universitarios
<int> 0, 0, 0, 0, ...
$ colegios_mayores
<int> 0, 0, 0, 0, ...
$ galerias_de_alimentacion
<int> 0, 0, 0, 0, ...
$ grandes_superficies_especializadas
<int> 0, 0, 0, 0, ...
$ hipermercados
<int> 0, 0, 0, 0, ...
$ mercados_de_abastos
<int> 0, 0, 0, 0, ...
$ agricultura_y_ganaderia
<int> 0, 0, 0, 0, ...
$ centros_educativos_no_universitarios_servicios_educativos
<int> 0, 0, 0, 0, ...
$ seguridad_social
<int> 0, 0, 0, 0, ...
$ centros_comerciales
<int> 0, 0, 0, 0, ...

```

```
$ oficinas_de_empleo
<int> 0, 0, 0, 0, ...
$ embajadas_y_consulados
<int> 0, 0, 0, 0, ...
```

0.9 Objeto tdata_02

- Tenemos 12 filas, una por tipo de infraestructura, con sus metadatos

```
[40]: glimpse(tdata_02)
```

```
Rows: 47
Columns: 6
$ desc_var    <chr> "Ayuntamientos, Consejerías, Ministerios,
etc.", "Sanidad y...
$ nombre_var  <chr>
"ayuntamientos_consejerias_ministerios_etc", "sanidad_y_ser...
$ Grupo.x     <chr> "Administración pública", "Colectivo
empresarial por tamaño...
$ n.x         <int> 245, 6422, 7265, 3103, 10696, 965, 806,
1983, 7391, 16908, ...
$ Grupo.y     <chr> "Administración pública", "Colectivo
empresarial por tamaño...
$ n.y         <int> 245, 6422, 7265, 3103, 10696, 965, 806,
1983, 7391, 16908, ...
```

Observaciones generales sobre los datos

- No aplica

0.9.1 Consideraciones para despliegue en piloto

- Utilizar los metadatos para etiquetar gráficos y otras salidas.

0.9.2 Consideraciones para despliegue en producción

- No aplica

0.10 Main Actions

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

- Se han agrupado las infraestructuras por distrito
- Se han calculado el número de infraestructuras por tipo en cada distrito

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

- Los datos se unirán a los de infraestructuras para ajustar los modelos

0.11 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

```
[ ]: # incluir código
```