Manejo de datos en R

Usos de tidyverse

Elena Quintero | Curso R AEET | 19 Sept 2022

Paquetes que usaremos:

```
install.packages(c("tidyverse",
                   "here",
                   "readxl",
                   "tidylog",
                   "summarytools",
                   "knitr"))
```

Paquetes incluidos en tidyverse:

```
library(tidyverse)
tidyverse_packages()
```

```
##  [1] "broom"        "cli"          "crayon"       "dbplyr"       "dplyr"        "dtplyr"
##  [7] "forcats"      "googledrive"  "googlesheets4" "ggplot2"      "haven"        "hms"
## [13] "httr"         "jsonlite"     "lubridate"    "magrittr"     "modelr"       "pillar"
## [19] "purrr"        "readr"        "readxl"       "reprex"       "rlang"        "rstudioapi"
## [25] "rvest"        "stringr"      "tibble"       "tidyr"        "xml2"         "tidyverse"
```

```
library(readr)      #leer archivos
library(readxl)     #leer archivos excel
library(dplyr)      #manipular datos
library(tidyr)      #ordenar y trasformar datasets
library(stringr)    #manipular caracteres
library(forcats)    #manipular factores
library(lubridate)  #manipular fechas
```

Otros paquetes que utilizaremos:

```
library(here)            #refiere la ruta a la carpeta del proyecto
library(tidylog)         #informa sobre operaciones dplyr y tidyr
library(summarytools)    #resume de forma clara y rápida datos numéricos y categóricos
library(knitr)           #reportar datos en varios formatos
```

Leer datos

- Library *base*

`read.table`, `read.csv`, `readRDS`

Argumentos útiles: sep, dec, comment.char, na.strings, stringsAsFactors

Leer datos

- Library *base*

`read.table`, `read.csv`, `readRDS`

Argumentos útiles: sep, dec, comment.char, na.strings, stringsAsFactors

- Library *readr*

`read_delim`, `read_csv`, `read_csv2`, `read_table`

Más rapido, produce "tibbles", no convierte characteres a factors automaticamente, no usa los nombres de fila.

Argumentos útiles: delim, comment, na, col_types, skip_empty_rows, guess_max

Leer datos

- Library *base*

`read.table`, `read.csv`, `readRDS`

Argumentos útiles: sep, dec, comment.char, na.strings, stringsAsFactors

- Library *readr*

`read_delim`, `read_csv`, `read_csv2`, `read_table`

Más rapido, produce "tibbles", no convierte characteres a factors automaticamente, no usa los nombres de fila.

Argumentos útiles: delim, comment, na, col_types, skip_empty_rows, guess_max

- Library *readxl*

`read_excel`, `read_xls`, `read_xlsx`

Argumentos útiles: sheet, col_types, skip

Leer datos con `readr`

```
library(readr)
```

La función `read_delim()` lee varios tipos de archivo. El argumento delim, especifica el separador.

Además tiene funciones específicas como:

- `read_csv()` usa `','` como campo de separación, y `'.'` para el punto decimal.
- `read_csv2()` usa `';'` como campo de separación, y `','` para el punto decimal.

library(here)

La función `here()` permite hacer referencia siempre al directorio donde se encuentra el proyecto.

Allison Horst Illustration

library(here)

Ejemplo de uso.

Usando ruta absoluta:

```
data <- read_csv("C:/Usuarios/Elena/Documentos/Proyectos/Proyecto_peces/datos/medida_peces.csv")
```

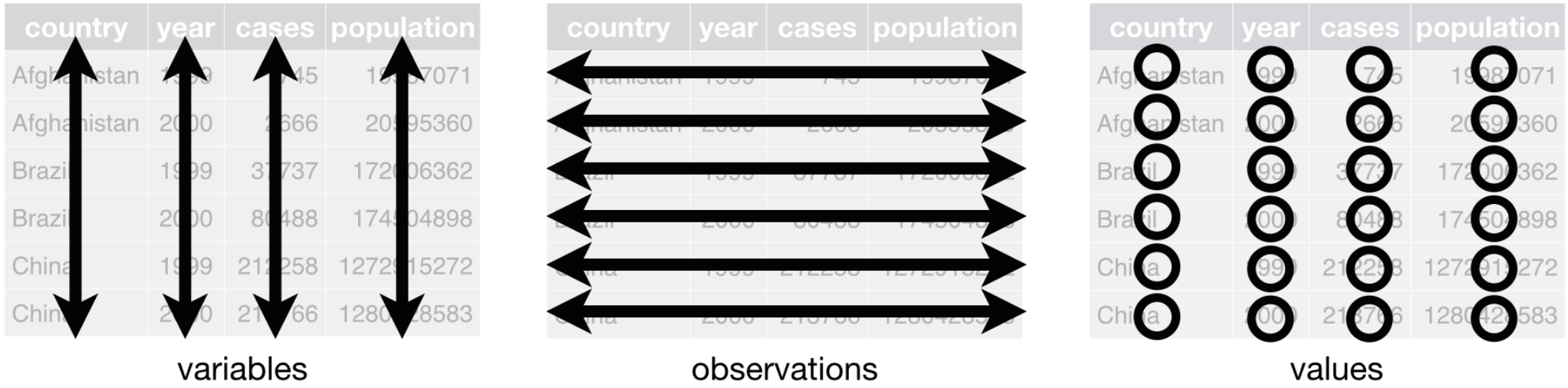
Usando ruta relativa al proyecto:

```
data <- read_csv(here("datos/medida_peces"))
```

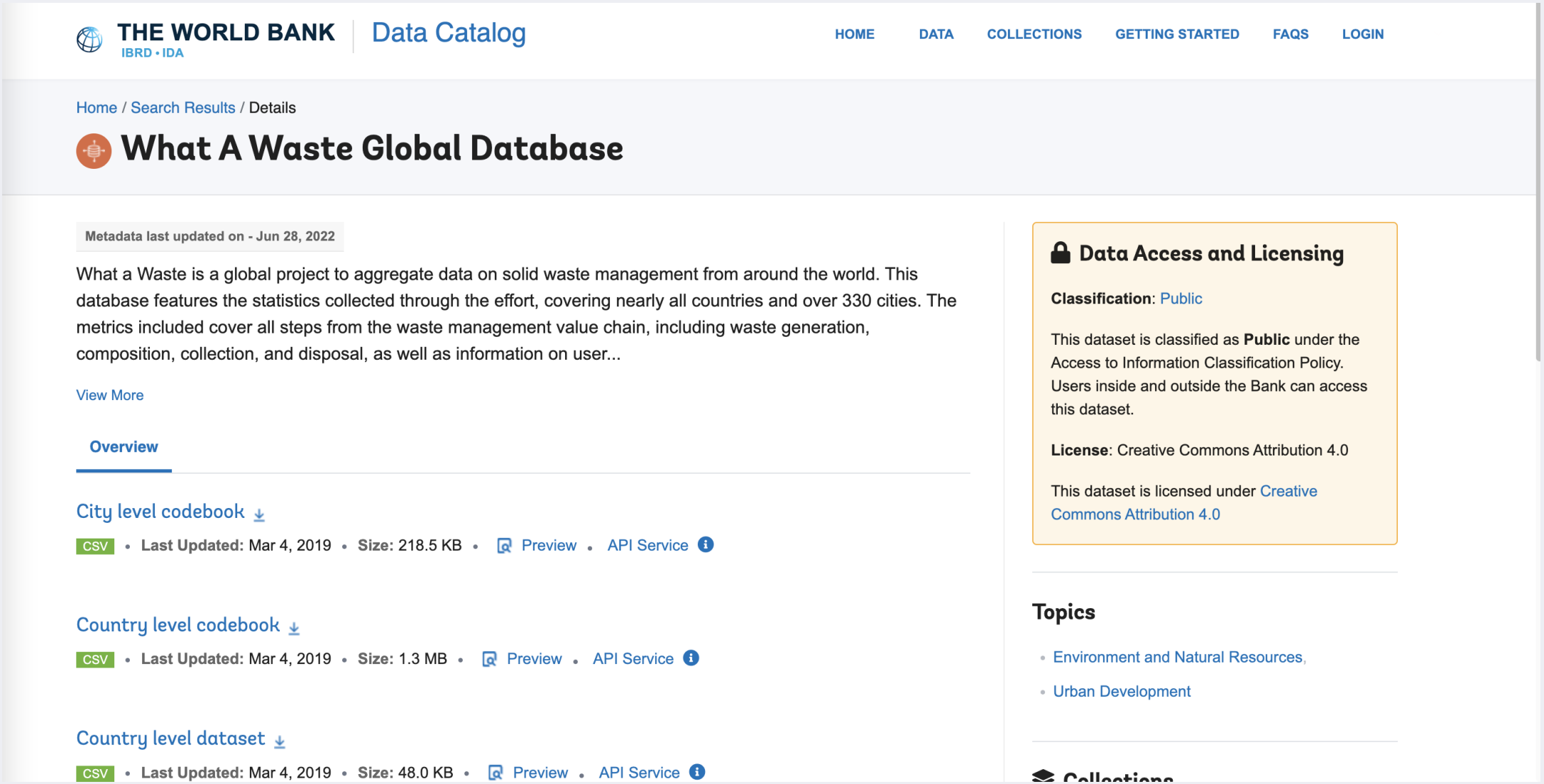
Formato tidy data:

Tres reglas para que los datos estén ordenado:

- Cada variable debe tener su propia columna
- Cada observación debe tener su propia fila
- Cada valor debe tener su propia celda



https://datacatalog.worldbank.org/search/dataset/0039597/What-a-Waste-Global-Database



THE WORLD BANK | Data Catalog
IBRD • IDA

HOME DATA COLLECTIONS GETTING STARTED FAQS LOGIN

Home / Search Results / Details

What A Waste Global Database

Metadata last updated on - Jun 28, 2022

What a Waste is a global project to aggregate data on solid waste management from around the world. This database features the statistics collected through the effort, covering nearly all countries and over 330 cities. The metrics included cover all steps from the waste management value chain, including waste generation, composition, collection, and disposal, as well as information on user...

View More

Overview

City level codebook
CSV • Last Updated: Mar 4, 2019 • Size: 218.5 KB • Preview • API Service

Country level codebook
CSV • Last Updated: Mar 4, 2019 • Size: 1.3 MB • Preview • API Service

Country level dataset
CSV • Last Updated: Mar 4, 2019 • Size: 48.0 KB • Preview • API Service

Data Access and Licensing

Classification: Public

This dataset is classified as **Public** under the Access to Information Classification Policy. Users inside and outside the Bank can access this dataset.

License: Creative Commons Attribution 4.0

This dataset is licensed under Creative Commons Attribution 4.0

Topics

- Environment and Natural Resources,
- Urban Development

Collections

Leer dataset

```
library(readr)
library(here)

waste <- read_csv(here("data_raw/country_level_data.csv"))
```

```
## Rows: 217 Columns: 28
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (5): iso3c, region_id, country_name, income_id, where_is_this_data_measured
## dbl (23): gdp, population_population_number_of_people, total_msw_total_msw_generated_tons_year, c...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#colnames(waste)
dplyr::glimpse(waste)
```

```
## Rows: 217
## Columns: 28
## $ iso3c                                          <chr> "ABW", "AFG", "AGO", "ALB", "…
## $ region_id                                      <chr> "LCN", "SAS", "SSF", "ECS", "…
## $ country_name                                   <chr> "Aruba", "Afghanistan", "Ango…
## $ income_id                                      <chr> "HIC", "LIC", "LMC", "UMC", "…
## $ gdp                                            <dbl> 35563.3125, 2057.0623, 8036.6…
## $ population_population_number_of_people         <dbl> 103187, 34656032, 25096150, 2…
## $ total_msw_total_msw_generated_tons_year        <dbl> 88132.02, 5628525.37, 4213643…
## $ composition_food_organic_waste_percent         <dbl> NA, NA, 51.80000, 51.40000, 3…
## $ composition_glass_percent                      <dbl> NA, NA, 6.700000, 4.500000, 8…
## $ composition_metal_percent                      <dbl> NA, NA, 4.400000, 4.800000, 2…
## $ composition_other_percent                      <dbl> NA, NA, 11.5000, 15.2100, 11.…
## $ composition_paper_cardboard_percent            <dbl> NA, NA, 11.900000, 9.900000, …
## $ composition_plastic_percent                    <dbl> NA, NA, 13.50000, 9.60000, 11…
## $ composition_rubber_leather_percent             <dbl> NA, NA, NA, NA, NA, NA, 1.26,…
## $ composition_wood_percent                       <dbl> NA, NA, NA, 4.60, NA, NA, 1.0…
## $ composition_yard_garden_green_waste_percent    <dbl> NA, NA, NA, NA, NA, NA, 9.95,…
## $ waste_treatment_anaerobic_digestion_percent    <dbl> NA, NA, NA, NA, NA, NA, NA, N…
## $ waste_treatment_compost_percent                <dbl> NA, NA, NA, NA, NA, 9.000, NA…
## $ waste_treatment_controlled_landfill_percent    <dbl> NA, NA, NA, NA, NA, NA, 8.900…
## $ waste_treatment_incineration_percent           <dbl> NA, NA, NA, NA, 52.10, NA, NA…
## $ waste_treatment_landfill_unspecified_percent   <dbl> NA, NA, NA, NA, NA, 9.00, NA,…
## $ waste_treatment_open_dump_percent              <dbl> NA, NA, NA, NA, NA, 62.00000,…
```

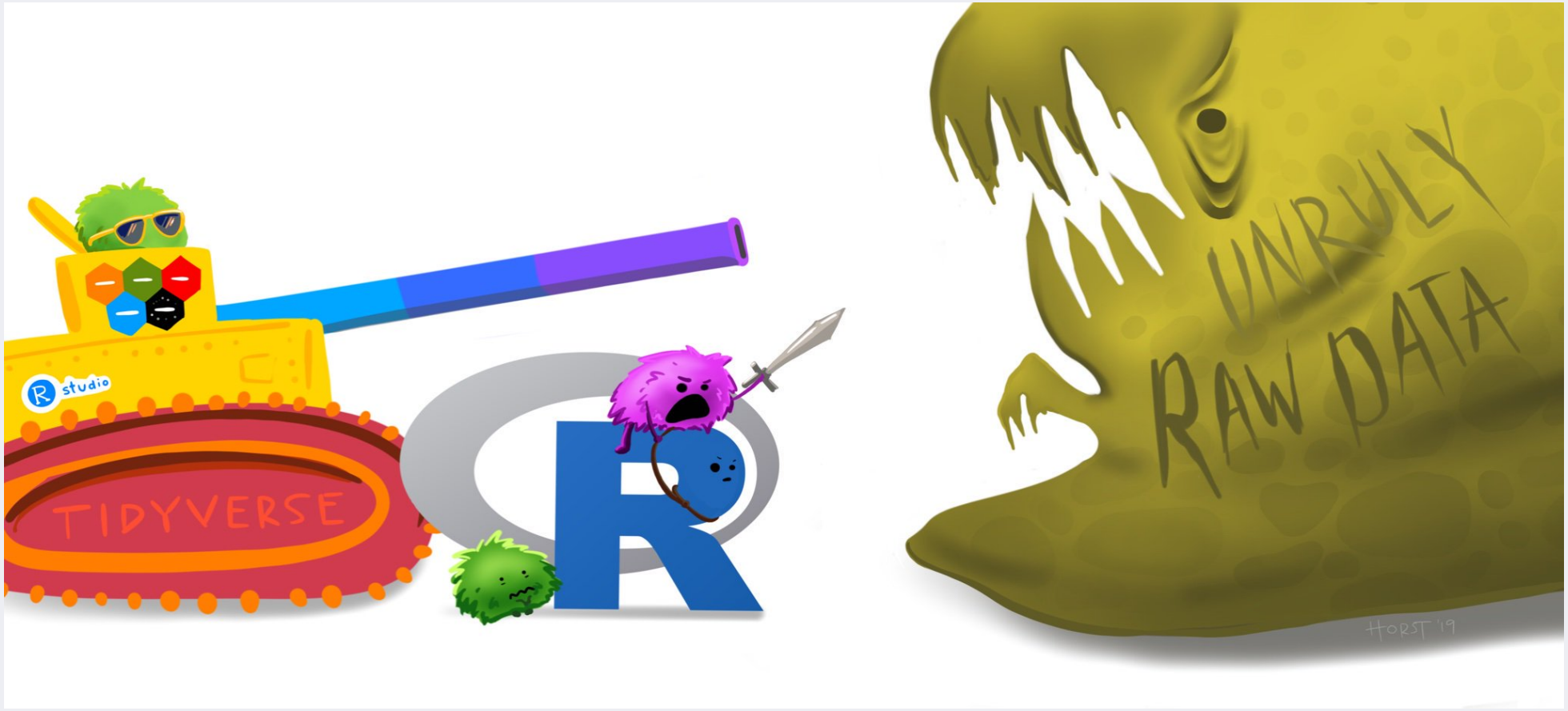
```
head(waste)
```

```
## # A tibble: 6 × 28
##   iso3c region_id country_name    income_id    gdp population_popu… total_msw_total… composition_foo…
##   <chr> <chr>     <chr>           <chr>      <dbl>             <dbl>             <dbl>             <dbl>
## 1 ABW   LCN       Aruba           HIC       35563.            103187            88132.              NA
## 2 AFG   SAS       Afghanistan     LIC        2057.          34656032          5628525.              NA
## 3 AGO   SSF       Angola          LMC        8037.          25096150          4213644.              51.8
## 4 ALB   ECS       Albania         UMC       13724.           2854191          1087447.              51.4
## 5 AND   ECS       Andorra         HIC       43712.             82431            43000               31.2
## 6 ARE   MEA       United Arab Em… HIC       67119.           9770529          5617682               39
## # … with 20 more variables: composition_glass_percent <dbl>, composition_metal_percent <dbl>,
## #   composition_other_percent <dbl>, composition_paper_cardboard_percent <dbl>,
## #   composition_plastic_percent <dbl>, composition_rubber_leather_percent <dbl>,
## #   composition_wood_percent <dbl>, composition_yard_garden_green_waste_percent <dbl>,
## #   waste_treatment_anaerobic_digestion_percent <dbl>, waste_treatment_compost_percent <dbl>,
## #   waste_treatment_controlled_landfill_percent <dbl>, waste_treatment_incineration_percent <dbl>,
## #   waste_treatment_landfill_unspecified_percent <dbl>, waste_treatment_open_dump_percent <dbl>, …
```

```
tail(waste)
```

```
## # A tibble: 6 × 28
##   iso3c region_id country_name income_id    gdp population_populat… total_msw_total… composition_foo…
##   <chr> <chr>     <chr>        <chr>      <dbl>                <dbl>             <dbl>             <dbl>
## 1 WSM   EAS       Samoa        UMC        6211.               187665            27399.              42.6
## 2 XKX   ECS       Kosovo       LMC        9724.              1801800           319000               42
## 3 YEM   MEA       Yemen, Rep.  LIC        8270.             27584212          4836820               65
## 4 ZAF   SSF       South Africa UMC       12667.             51729344         18457232               16.4
## 5 ZMB   SSF       Zambia       LMC        3201.             14264756          2608268               NA
## 6 ZWE   SSF       Zimbabwe     LIC        3191.             12500525          1449752               36
## # … with 20 more variables: composition_glass_percent <dbl>, composition_metal_percent <dbl>,
## #   composition_other_percent <dbl>, composition_paper_cardboard_percent <dbl>,
## #   composition_plastic_percent <dbl>, composition_rubber_leather_percent <dbl>,
## #   composition_wood_percent <dbl>, composition_yard_garden_green_waste_percent <dbl>,
## #   waste_treatment_anaerobic_digestion_percent <dbl>, waste_treatment_compost_percent <dbl>,
## #   waste_treatment_controlled_landfill_percent <dbl>, waste_treatment_incineration_percent <dbl>,
## #   waste_treatment_landfill_unspecified_percent <dbl>, waste_treatment_open_dump_percent <dbl>, …
```

Manejo de datos con tidyverse



Allison Horst Illustration

Manejo de datos con tidyverse

```
library(tidylog)
```

Da información de las operaciones que se realizan en el dataset.

El operador 'pipe'

```
library(magrittr)
```

Mecanismo para encadenar funciones.

```
data %>% function(...)
```

Ahora también implementado en R base como `|>`

```
data |> function(...)
```

Funciones de `dplyr`

- `arrange()` - Ordenar variable por casos
- `rename()` - Renombrar variables
- `relocate()` - Reordenar variables
- `select()` - Extraer variables

Ayudas de select:

Selecciona columnas que...

- contains() - *contienen ""*
- matches() - *coinciden con ""*
- starts_with() - *empiezan por ""*
- ends_with() - *acaban por ""*
- any_of() - *que estén en el set c("","")*

Ordernar datos por columnas:

```
waste %>%
  arrange(population_population_number_of_people)
```

```
## # A tibble: 217 × 28
##    iso3c region_id country_name   income_id    gdp population_popu… total_msw_total… composition_foo…
##    <chr> <chr>     <chr>          <chr>      <dbl>            <dbl>            <dbl>             <dbl>
##  1 TUV   EAS       Tuvalu         UMC        3793.            11097             3989.             43.6
##  2 NRU   EAS       Nauru          UMC       11167.            13049             6192.             43.6
##  3 VGB   LCN       British Virgi… HIC       24216.            20645            21099.              6.5
##  4 PLW   EAS       Palau          HIC       18275.            21503             9427.             26
##  5 MAF   LCN       St. Martin (F… HIC       30386.            30959            15480.             NA
##  6 SMR   ECS       San Marino     HIC       58806.            33203            17175.              5.35
##  7 GIB   ECS       Gibraltar      HIC       43712.            33623            16954              24.6
##  8 TCA   LCN       Turks and Cai… HIC       28174.            34900               NA              21.8
##  9 LIE   ECS       Liechtenstein  HIC       45727.            36545            32382              37.6
## 10 SXM   LCN       Sint Maarten … HIC          NA             37685               NA              46
## # … with 207 more rows, and 20 more variables: composition_glass_percent <dbl>,
## #   composition_metal_percent <dbl>, composition_other_percent <dbl>,
## #   composition_paper_cardboard_percent <dbl>, composition_plastic_percent <dbl>,
## #   composition_rubber_leather_percent <dbl>, composition_wood_percent <dbl>,
## #   composition_yard_garden_green_waste_percent <dbl>,
## #   waste_treatment_anaerobic_digestion_percent <dbl>, waste_treatment_compost_percent <dbl>,
## #   waste_treatment_controlled_landfill_percent <dbl>, waste_treatment_incineration_percent <dbl>, …
```

Ordernar datos por columnas

descendiente:

```
waste %>%
  arrange(desc(population_population_number_of_people))
```

```
## # A tibble: 217 × 28
##    iso3c region_id country_name   income_id    gdp population_popu… total_msw_total… composition_foo…
##    <chr> <chr>     <chr>          <chr>      <dbl>             <dbl>             <dbl>             <dbl>
##  1 CHN   EAS       China          UMC       16092.       1400050048       395081376              61.2
##  2 IND   SAS       India          LMC        6497.       1352617344       189750000              NA
##  3 USA   NAC       United States  HIC       61498.        326687488       265224528              14.9
##  4 IDN   EAS       Indonesia      LMC       10531.        261115456        65200000              53.8
##  5 BRA   LCN       Brazil         UMC       14596.        208494896        79069584              51.4
##  6 PAK   SAS       Pakistan       LMC        4571.        193203472        30760000              30
##  7 BGD   SAS       Bangladesh     LMC        3196.        155727056        14778497.             80.6
##  8 NGA   SSF       Nigeria        LMC        4690.        154402176        27614830.             NA
##  9 RUS   ECS       Russian Feder… UMC       26013.        143201680        60000000              28.4
## 10 JPN   EAS       Japan          HIC       41310.        126529104        42720000              36
## # … with 207 more rows, and 20 more variables: composition_glass_percent <dbl>,
## #   composition_metal_percent <dbl>, composition_other_percent <dbl>,
## #   composition_paper_cardboard_percent <dbl>, composition_plastic_percent <dbl>,
## #   composition_rubber_leather_percent <dbl>, composition_wood_percent <dbl>,
## #   composition_yard_garden_green_waste_percent <dbl>,
## #   waste_treatment_anaerobic_digestion_percent <dbl>, waste_treatment_compost_percent <dbl>,
## #   waste_treatment_controlled_landfill_percent <dbl>, waste_treatment_incineration_percent <dbl>,…
```

Ordernar datos por orden jerárquico:

```
waste %>%
  arrange(region_id, country_name)
```

```
## # A tibble: 217 × 28
##    iso3c region_id country_name   income_id    gdp population_popu… total_msw_total… composition_foo…
##    <chr> <chr>     <chr>          <chr>      <dbl>             <dbl>            <dbl>             <dbl>
##  1 ASM   EAS       American Samoa UMC       11113.             55599           18989.              19.7
##  2 AUS   EAS       Australia      HIC       47784.          23789338        13345000               48.4
##  3 BRN   EAS       Brunei Daruss… HIC       60866.            423196          216253.              36
##  4 KHM   EAS       Cambodia       LMC        3364.          15270790         1089000               66.6
##  5 CHN   EAS       China          UMC       16092.        1400050048       395081376               61.2
##  6 FJI   EAS       Fiji           UMC       10788.            867086          189390.              33.2
##  7 PYF   EAS       French Polyne… HIC       60956.            273528          147000               47
##  8 GUM   EAS       Guam           HIC       59075.            159973          141500               27.3
##  9 HKG   EAS       Hong Kong SAR… HIC       57216.           7305700         5679816.              35
## 10 IDN   EAS       Indonesia      LMC       10531.         261115456        65200000               53.8
## # … with 207 more rows, and 20 more variables: composition_glass_percent <dbl>,
## #   composition_metal_percent <dbl>, composition_other_percent <dbl>,
## #   composition_paper_cardboard_percent <dbl>, composition_plastic_percent <dbl>,
## #   composition_rubber_leather_percent <dbl>, composition_wood_percent <dbl>,
## #   composition_yard_garden_green_waste_percent <dbl>,
## #   waste_treatment_anaerobic_digestion_percent <dbl>, waste_treatment_compost_percent <dbl>,
## #   waste_treatment_controlled_landfill_percent <dbl>, waste_treatment_incineration_percent <dbl>, …
```

Cambiar nombre columnas:

```
waste %>%
  rename(population = population_population_number_of_people,
         total_waste = total_msw_total_msw_generated_tons_year)
```

```
## rename: renamed 2 variables (population, total_waste)
```

```
## # A tibble: 217 × 28
##    iso3c region_id country_name         income_id    gdp population total_waste composition_food_org…
##    <chr> <chr>     <chr>                <chr>      <dbl>      <dbl>       <dbl>                  <dbl>
##  1 ABW   LCN       Aruba                HIC       35563.     103187      88132.                   NA
##  2 AFG   SAS       Afghanistan          LIC        2057.   34656032    5628525.                   NA
##  3 AGO   SSF       Angola               LMC        8037.   25096150    4213644.                   51.8
##  4 ALB   ECS       Albania              UMC       13724.    2854191    1087447.                   51.4
##  5 AND   ECS       Andorra              HIC       43712.      82431      43000                    31.2
##  6 ARE   MEA       United Arab Emirates HIC       67119.    9770529    5617682                    39
##  7 ARG   LCN       Argentina            HIC       23550.   42981516   17910550                    38.7
##  8 ARM   ECS       Armenia              UMC       11020.    2906220     492800                    57
##  9 ASM   EAS       American Samoa       UMC       11113.      55599      18989.                   19.7
## 10 ATG   LCN       Antigua and Barbuda  HIC       17966.      96777      30585                    46
## # … with 207 more rows, and 20 more variables: composition_glass_percent <dbl>,
## #   composition_metal_percent <dbl>, composition_other_percent <dbl>,
## #   composition_paper_cardboard_percent <dbl>, composition_plastic_percent <dbl>,
## #   composition_rubber_leather_percent <dbl>, composition_wood_percent <dbl>,
## #   composition_yard_garden_green_waste_percent <dbl>,
## #   waste_treatment_anaerobic_digestion_percent <dbl>, waste_treatment_compost_percent <dbl>,
```

Seleccionar sólo variables de interés:

```
waste_select <- waste %>%
  select(iso3c,
         region_id,
         country = country_name,
         income_id,
         gdp,
         population = population_population_number_of_people,
         total_waste = total_msw_total_msw_generated_tons_year,
         starts_with("composition"))
```

```
## select: renamed 3 variables (country, population, total_waste) and dropped 12 variables
```

```
glimpse(waste_select)
```

```
## Rows: 217
## Columns: 16
## $ iso3c                                      <chr> "ABW", "AFG", "AGO", "ALB", "AND", "ARE", "ARG"…
## $ region_id                                  <chr> "LCN", "SAS", "SSF", "ECS", "ECS", "MEA", "LCN"…
## $ country                                    <chr> "Aruba", "Afghanistan", "Angola", "Albania", "A…
## $ income_id                                  <chr> "HIC", "LIC", "LMC", "UMC", "HIC", "HIC", "HIC"…
## $ gdp                                        <dbl> 35563.3125, 2057.0623, 8036.6904, 13724.0586, 4…
## $ population                                 <dbl> 103187, 34656032, 25096150, 2854191, 82431, 977…
## $ total_waste                                <dbl> 88132.02, 5628525.37, 4213643.58, 1087446.75, 4…
## $ composition_food_organic_waste_percent     <dbl> NA, NA, 51.80000, 51.40000, 31.20000, 39.00000,…
## $ composition_glass_percent                  <dbl> NA, NA, 6.700000, 4.500000, 8.200000, 4.000000,…
## $ composition_metal_percent                  <dbl> NA, NA, 4.400000, 4.800000, 2.600000, 3.000000,…
## $ composition_other_percent                  <dbl> NA, NA, 11.5000, 15.2100, 11.6000, 10.0000, 15.…
## $ composition_paper_cardboard_percent        <dbl> NA, NA, 11.900000, 9.900000, 35.100000, 25.0000…
## $ composition_plastic_percent                <dbl> NA, NA, 13.50000, 9.60000, 11.30000, 19.00000, …
## $ composition_rubber_leather_percent         <dbl> NA, NA, NA, NA, NA, NA, 1.26, 0.30, 4.20, NA, N…
## $ composition_wood_percent                   <dbl> NA, NA, NA, 4.60, NA, NA, 1.09, 0.60, NA, NA, N…
## $ composition_yard_garden_green_waste_percent <dbl> NA, NA, NA, NA, NA, NA, 9.95, NA, NA, NA, NA, N…
```

(Des)seleccionar variables:

```
waste_select %>%
  select(-gdp)
```

```
## select: dropped one variable (gdp)

## # A tibble: 217 × 15
##    iso3c region_id country         income_id population total_waste composition_foo… composition_gla…
##    <chr> <chr>     <chr>           <chr>           <dbl>       <dbl>             <dbl>             <dbl>
##  1 ABW   LCN       Aruba           HIC            103187      88132.              NA                NA
##  2 AFG   SAS       Afghanistan     LIC          34656032    5628525.              NA                NA
##  3 AGO   SSF       Angola          LMC          25096150    4213644.              51.8               6.7
##  4 ALB   ECS       Albania         UMC           2854191    1087447.              51.4               4.5
##  5 AND   ECS       Andorra         HIC             82431      43000               31.2               8.2
##  6 ARE   MEA       United Arab Em… HIC           9770529    5617682               39                 4
##  7 ARG   LCN       Argentina       HIC          42981516   17910550               38.7               3.16
##  8 ARM   ECS       Armenia         UMC           2906220     492800               57                 3.2
##  9 ASM   EAS       American Samoa  UMC             55599      18989.              19.7               3.4
## 10 ATG   LCN       Antigua and Ba… HIC             96777      30585               46                 7
## # … with 207 more rows, and 7 more variables: composition_metal_percent <dbl>,
## #   composition_other_percent <dbl>, composition_paper_cardboard_percent <dbl>,
## #   composition_plastic_percent <dbl>, composition_rubber_leather_percent <dbl>,
## #   composition_wood_percent <dbl>, composition_yard_garden_green_waste_percent <dbl>
```

Organizar columnas:

```
waste_select %>%
  relocate(country, .before = iso3c)
```

```
## relocate: columns reordered (country, iso3c, region_id, income_id, gdp, …)
```

```
## # A tibble: 217 × 16
##    country  iso3c region_id income_id    gdp population total_waste composition_foo… composition_gla…
##    <chr>    <chr> <chr>     <chr>      <dbl>      <dbl>       <dbl>             <dbl>             <dbl>
##  1 Aruba    ABW   LCN       HIC       35563.     103187      88132.              NA                NA
##  2 Afghani… AFG   SAS       LIC        2057.   34656032    5628525.              NA                NA
##  3 Angola   AGO   SSF       LMC        8037.   25096150    4213644.              51.8               6.7
##  4 Albania  ALB   ECS       UMC       13724.    2854191    1087447.              51.4               4.5
##  5 Andorra  AND   ECS       HIC       43712.      82431      43000               31.2               8.2
##  6 United … ARE   MEA       HIC       67119.    9770529    5617682               39                 4
##  7 Argenti… ARG   LCN       HIC       23550.   42981516   17910550               38.7               3.16
##  8 Armenia  ARM   ECS       UMC       11020.    2906220     492800               57                 3.2
##  9 America… ASM   EAS       UMC       11113.      55599      18989.              19.7               3.4
## 10 Antigua… ATG   LCN       HIC       17966.      96777      30585               46                 7
## # … with 207 more rows, and 7 more variables: composition_metal_percent <dbl>,
## #   composition_other_percent <dbl>, composition_paper_cardboard_percent <dbl>,
## #   composition_plastic_percent <dbl>, composition_rubber_leather_percent <dbl>,
## #   composition_wood_percent <dbl>, composition_yard_garden_green_waste_percent <dbl>
```

Tablas de resumen de datos:

```
library(summarytools)
dfSummary(waste_select$region_id)
```

```
## waste_select$region_id was converted to a data frame

## Data Frame Summary
## waste_select
## Dimensions: 217 x 1
## Duplicates: 210
##
## ------------------------------------------------------------------------------
## No   Variable      Stats / Values   Freqs (% of Valid)   Graph     Valid      Missing
## ---- ------------ ---------------- -------------------- --------- ---------- ---------
## 1    region_id     1. EAS           37 (17.1%)           III       217        0
##      [character]   2. ECS           58 (26.7%)           IIIII     (100.0%)   (0.0%)
##                    3. LCN           42 (19.4%)           III
##                    4. MEA           21 ( 9.7%)           I
##                    5. NAC            3 ( 1.4%)
##                    6. SAS            8 ( 3.7%)
##                    7. SSF           48 (22.1%)           IIII
## ------------------------------------------------------------------------------
```

```
waste_select %>%
  select(population, gdp) %>%
  dfSummary()
```

```
## select: dropped 14 variables (iso3c, region_id, country, income_id, total_waste, …)

## Data Frame Summary
## waste_select
## Dimensions: 217 x 2
## Duplicates: 0
##
## ------------------------------------------------------------------------------------------
## No   Variable     Stats / Values                    Freqs (% of Valid)    Graph         Valid      Mis
## ---- ------------ --------------------------------- --------------------- ------------- ---------- ---
## 1    population   Mean (sd) : 33643890 (136583825)   217 distinct values   :             217        0
##      [numeric]    min < med < max:                                        :             (100.0%)   (0.
##                   11097 < 5737723 < 1400050048                            :
##                   IQR (CV) : 20456779 (4.1)                               :
##                                                                           :
##
## 2    gdp          Mean (sd) : 22645.9 (22663.6)     213 distinct values   :             216        1
##      [numeric]    min < med < max:                                        :             (99.5%)    (0.
##                   822.6 < 13465.9 < 117335.6                              :
##                   IQR (CV) : 31111.8 (1)                                  : :
##                                                                           : : : : : .
## ------------------------------------------------------------------------------------------
```

Más funciones de `dplyr`

- `distinct()` - Extraer valores únicos
- `recode()` - Recodificar casos de una variable
- `group_by()` - Agrupar datos por casos
- `summarise()` - Resumir datos por casos
- `mutate()` - Crear nuevas variables
- `filter()` - Filtrar datos por casos
- `case_when()` - Filtrar datos por casos

Extraer valores únicos (niveles) de una(s) variable(s):

```
waste_select %>%
  distinct(income_id)
```

```
## distinct: removed 213 rows (98%), 4 rows remaining
```

```
## # A tibble: 4 × 1
##   income_id
##   <chr>
## 1 HIC
## 2 LIC
## 3 LMC
## 4 UMC
```

Igual a:

```
base::unique(waste_select$income_id)
```

```
## [1] "HIC" "LIC" "LMC" "UMC"
```

LIC = Low income; LMC = Lower middle income; UMC = Upper middle income; HIC = High income

Recodificar niveles de una variable:

```
waste_select %>%
  distinct(region_id)
```

```
## distinct: removed 210 rows (97%), 7 rows remaining
```

```
## # A tibble: 7 × 1
##   region_id
##   <chr>
## 1 LCN
## 2 SAS
## 3 SSF
## 4 ECS
## 5 MEA
## 6 EAS
## 7 NAC
```

- LCN: Latin America & Caribbean
- SAS: South Asia
- SSF: Sub-Saharan Africa
- ECS: Europe & Central Asia
- MEA: Middle East & North Africa
- EAS: East Asia & Pacific
- NAC: North America

Recodificar niveles de una variable:

```
waste_regions <- waste_select %>%
  mutate(region_id = recode(region_id,
                            "LCN" = "Latin_America",
                            "SAS" = "South_Asia",
                            "SSF" = "Sub-Saharan_Africa",
                            "ECS" = "Europe_Central_Asia",
                            "MEA" = "Middle_East_North_Africa",
                            "EAS" = "East_Asia_Pacific",
                            "NAC" = "North_America"))
```

```
## mutate: changed 217 values (100%) of 'region_id' (0 new NA)
```

```
distinct(waste_regions, region_id)
```

```
## # A tibble: 7 × 1
##   region_id
##   <chr>
## 1 Latin_America
## 2 South_Asia
## 3 Sub-Saharan_Africa
## 4 Europe_Central_Asia
## 5 Middle_East_North_Africa
## 6 East_Asia_Pacific
## 7 North_America
```

Agrupar datos y resumir:

```
waste_regions %>%
  group_by(region_id) %>%
  summarise(total_waste = sum(total_waste, na.rm = TRUE))
```

```
## group_by: one grouping variable (region_id)

## summarise: now 7 rows and 2 columns, ungrouped

## # A tibble: 7 × 2
##   region_id                total_waste
##   <chr>                          <dbl>
## 1 East_Asia_Pacific          630827137.
## 2 Europe_Central_Asia        407170316.
## 3 Latin_America              224893129.
## 4 Middle_East_North_Africa   124442193.
## 5 North_America              290409562
## 6 South_Asia                 245640470.
## 7 Sub-Saharan_Africa         149000010.
```

Crear nueva variable - Ej: transformar basura a millones de toneladas

```
waste_regions %>%
  group_by(region_id) %>%
  summarise(total_waste = sum(total_waste, na.rm = TRUE)) %>%
  mutate(waste_mtons = total_waste/1000000)
```

```
## group_by: one grouping variable (region_id)

## summarise: now 7 rows and 2 columns, ungrouped

## mutate: new variable 'waste_mtons' (double) with 7 unique values and 0% NA

## # A tibble: 7 × 3
##   region_id                total_waste waste_mtons
##   <chr>                          <dbl>       <dbl>
## 1 East_Asia_Pacific         630827137.        631.
## 2 Europe_Central_Asia       407170316.        407.
## 3 Latin_America             224893129.        225.
## 4 Middle_East_North_Africa  124442193.        124.
## 5 North_America             290409562         290.
## 6 South_Asia                245640470.        246.
## 7 Sub-Saharan_Africa        149000010.        149.
```

Filtrar datos:

```
waste_regions %>%
  filter(region_id == "Latin_America")
```

```
## filter: removed 175 rows (81%), 42 rows remaining
```

```
## # A tibble: 42 × 16
##    iso3c region_id  country income_id    gdp population total_waste composition_foo… composition_gla…
##    <chr> <chr>      <chr>   <chr>      <dbl>      <dbl>       <dbl>             <dbl>             <dbl>
##  1 ABW   Latin_Ame… Aruba   HIC       35563.     103187      88132.              NA                NA
##  2 ARG   Latin_Ame… Argent… HIC       23550.   42981516   17910550              38.7               3.16
##  3 ATG   Latin_Ame… Antigu… HIC       17966.      96777      30585              46                 7
##  4 BHS   Latin_Ame… Bahama… HIC       35400.     386838     264000              46                 7
##  5 BLZ   Latin_Ame… Belize  UMC        7259.     359288     101379.             47                 8
##  6 BOL   Latin_Ame… Bolivia LMC        7984.   10724705    2219052              55.2               2.90
##  7 BRA   Latin_Ame… Brazil  UMC       14596.  208494896   79069584              51.4               2.4
##  8 BRB   Latin_Ame… Barbad… HIC       15445.     280601     174815.             18.3               3.7
##  9 CHL   Latin_Ame… Chile   HIC       20362.   16829442    6517000              53.3               6.6
## 10 COL   Latin_Ame… Colomb… UMC       12523.   46406648   12150120              59.6               2.35
## # … with 32 more rows, and 7 more variables: composition_metal_percent <dbl>,
## #   composition_other_percent <dbl>, composition_paper_cardboard_percent <dbl>,
## #   composition_plastic_percent <dbl>, composition_rubber_leather_percent <dbl>,
## #   composition_wood_percent <dbl>, composition_yard_garden_green_waste_percent <dbl>
```

Filtrar datos:

```
waste_regions %>%
  filter(region_id == "Europe_Central_Asia" & population <= 1000000)
```

```
## filter: removed 205 rows (94%), 12 rows remaining
```

```
## # A tibble: 12 × 16
##    iso3c region_id  country income_id    gdp population total_waste composition_foo… composition_gla…
##    <chr> <chr>      <chr>   <chr>      <dbl>      <dbl>       <dbl>             <dbl>             <dbl>
##  1 AND   Europe_Ce… Andorra HIC       4.37e4      82431      43000              31.2               8.2
##  2 CHI   Europe_Ce… Channe… HIC       4.67e4     164541     178933              NA                NA
##  3 FRO   Europe_Ce… Faeroe… HIC       4.44e4      48842      61000              NA                NA
##  4 GIB   Europe_Ce… Gibral… HIC       4.37e4      33623      16954              24.6               4.42
##  5 GRL   Europe_Ce… Greenl… HIC       4.39e4      56905      50000              42.8               7.1
##  6 IMN   Europe_Ce… Isle o… HIC       4.42e4      80759      50551              NA                NA
##  7 ISL   Europe_Ce… Iceland HIC       5.53e4     343400     225270.             10                NA
##  8 LIE   Europe_Ce… Liecht… HIC       4.57e4      36545      32382              37.6               5
##  9 LUX   Europe_Ce… Luxemb… HIC       1.14e5     619896     490338.             30                 4
## 10 MCO   Europe_Ce… Monaco  HIC       4.37e4      37783      46000              NA                 3
## 11 MNE   Europe_Ce… Monten… UMC       2.08e4     622227     329780.             33.8               8.97
## 12 SMR   Europe_Ce… San Ma… HIC       5.88e4      33203      17175.              5.35              5.61
## # … with 7 more variables: composition_metal_percent <dbl>, composition_other_percent <dbl>,
## #   composition_paper_cardboard_percent <dbl>, composition_plastic_percent <dbl>,
## #   composition_rubber_leather_percent <dbl>, composition_wood_percent <dbl>,
## #   composition_yard_garden_green_waste_percent <dbl>
```

Clasificar variables en nuevo factor:

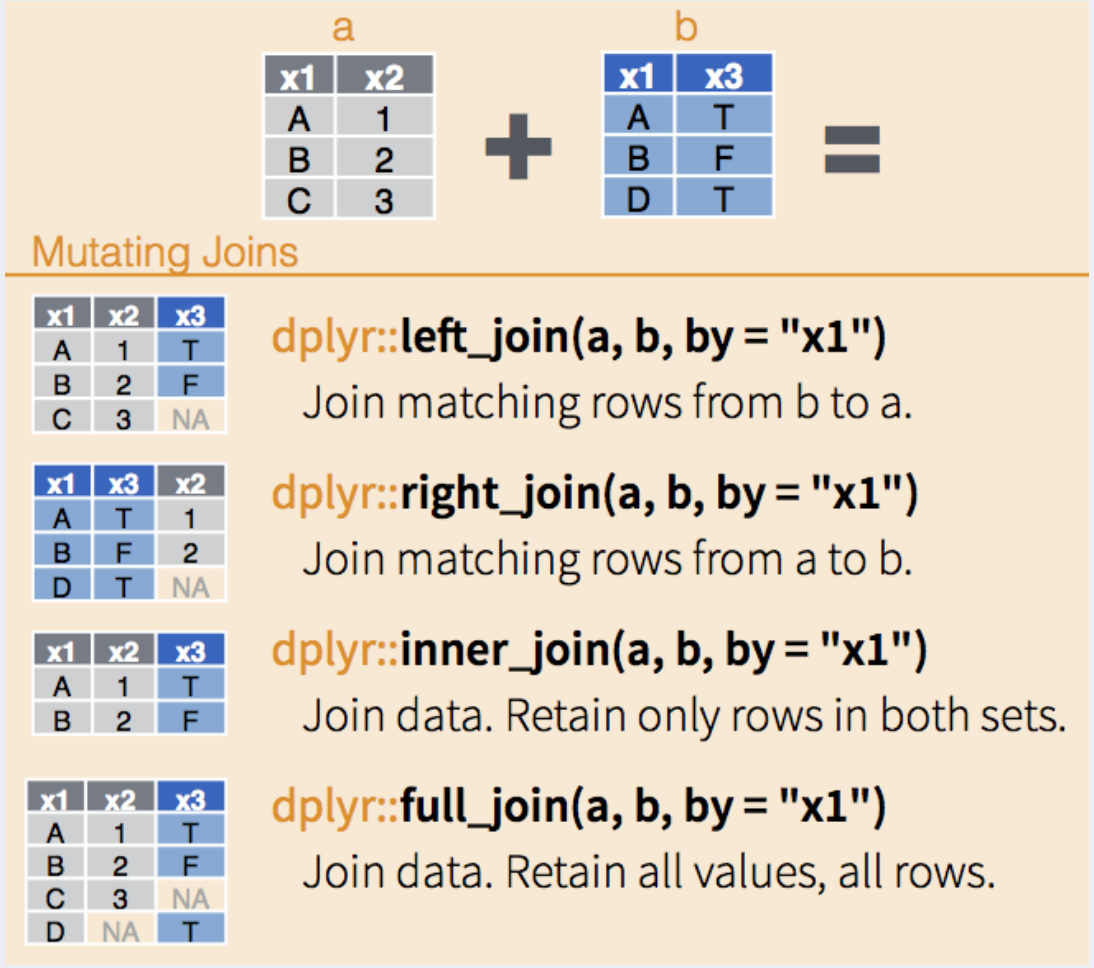
```
waste_regions %>%
  mutate(pop_size = case_when(
    population >= 1000000 ~ "big",
    population < 1000000 & population > 500000 ~ "medium",
    population <= 500000 ~ "small")) %>%
  relocate(pop_size, .before = population)
```

```
## mutate: new variable 'pop_size' (character) with 3 unique values and 0% NA

## relocate: columns reordered (iso3c, region_id, country, income_id, gdp, …)

## # A tibble: 217 × 17
##    iso3c region_id          country income_id    gdp pop_size population total_waste composition_foo…
##    <chr> <chr>              <chr>   <chr>      <dbl> <chr>         <dbl>       <dbl>             <dbl>
##  1 ABW   Latin_America      Aruba   HIC       35563. small        103187      88132.              NA
##  2 AFG   South_Asia         Afghan… LIC        2057. big         34656032    5628525.              NA
##  3 AGO   Sub-Saharan_Africa Angola  LMC        8037. big         25096150    4213644.              51.8
##  4 ALB   Europe_Central_As… Albania UMC       13724. big          2854191    1087447.              51.4
##  5 AND   Europe_Central_As… Andorra HIC       43712. small          82431      43000               31.2
##  6 ARE   Middle_East_North… United… HIC       67119. big          9770529    5617682               39
##  7 ARG   Latin_America      Argent… HIC       23550. big         42981516   17910550               38.7
##  8 ARM   Europe_Central_As… Armenia UMC       11020. big          2906220     492800               57
##  9 ASM   East_Asia_Pacific  Americ… UMC       11113. small          55599      18989.              19.7
## 10 ATG   Latin_America      Antigu… HIC       17966. small          96777      30585               46
## # … with 207 more rows, and 8 more variables: composition_glass_percent <dbl>,
```

Combinar bases de datos con `join`:



Data Wrangling CheatSheet

Combinar bases de datos con `join`:

Vamos a darle a nuestro dataset información sobre el continente donde se encuentra cada pais y su localización.

Leer nuevo dataset con información sobre el continente:

```
world_data <- read_csv2(here("data_raw/world_data.csv"))
```

```
glimpse(world_data)
```

```
## Rows: 241
## Columns: 16
## $ name       <chr> "Aruba", "Afghanistan", "Angola", "Anguilla", "Albania", "Aland", "Andorra", "Un…
## $ name_long  <chr> "Aruba", "Afghanistan", "Angola", "Anguilla", "Albania", "Aland Islands", "Andor…
## $ sovereignt <chr> "Netherlands", "Afghanistan", "Angola", "United Kingdom", "Albania", "Finland", …
## $ type       <chr> "Country", "Sovereign country", "Sovereign country", "Dependency", "Sovereign co…
## $ abbrev     <chr> "Aruba", "Afg.", "Ang.", "Ang.", "Alb.", "Aland", "And.", "U.A.E.", "Arg.", "Arm…
## $ continent  <chr> "North America", "Asia", "Africa", "North America", "Europe", "Europe", "Europe"…
## $ formal_en  <chr> "Aruba", "Islamic State of Afghanistan", "People's Republic of Angola", NA, "Rep…
## $ pop_est    <dbl> 103065, 28400000, 12799293, 14436, 3639453, 27153, 83888, 4798491, 40913584, 296…
## $ gdp_md_est <chr> "2258", "22270", "110300", "108.9", "21810", "1563", "3660", "184300", "573900",…
## $ pop_year   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
## $ lastcensus <dbl> 2010, 1979, 1970, NA, 2001, NA, 1989, 2010, 2010, 2001, 2010, NA, NA, NA, 2011, …
## $ gdp_year   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, …
## $ economy    <chr> "6. Developing region", "7. Least developed region", "7. Least developed region"…
## $ income_grp <chr> "2. High income: nonOECD", "5. Low income", "3. Upper middle income", "3. Upper …
## $ iso_a3     <chr> "ABW", "AFG", "AGO", "AIA", "ALB", "ALA", "AND", "ARE", "ARG", "ARM", "ASM", "AT…
## $ region_un  <chr> "Americas", "Asia", "Africa", "Americas", "Europe", "Europe", "Europe", "Asia", …
```

Seleccionar variables de interés:

```
continent <- world_data %>%
  select(iso_a3,
         country_name = name_long,
         continent)
```

```
## select: renamed one variable (country_name) and dropped 13 variables
```

```
glimpse(continent)
```

```
## Rows: 241
## Columns: 3
## $ iso_a3       <chr> "ABW", "AFG", "AGO", "AIA", "ALB", "ALA", "AND", "ARE", "ARG", "ARM", "ASM", "…
## $ country_name <chr> "Aruba", "Afghanistan", "Angola", "Anguilla", "Albania", "Aland Islands", "And…
## $ continent    <chr> "North America", "Asia", "Africa", "North America", "Europe", "Europe", "Europ…
```

Combinar datasets:

Usando `full_join()`

```
waste_world <- waste_regions %>%
  rename(iso_a3 = iso3c) %>%
  full_join(continent, by = "iso_a3")
```

```
## rename: renamed one variable (iso_a3)

## full_join: added 2 columns (country_name, continent)

##            > rows only in x     5

##            > rows only in y    29

##            > matched rows     212

##            >                 =====

##            > rows total       246
```

Combinar datasets:

Usando `left_join()`

```
waste_world <- waste_regions %>%
  rename(iso_a3 = iso3c) %>%
  left_join(continent, by = "iso_a3")
```

```
## rename: renamed one variable (iso_a3)

## left_join: added 2 columns (country_name, continent)

##            > rows only in x     5

##            > rows only in y  ( 29)

##            > matched rows     212

##            >                 =====

##            > rows total       217
```

¿Qué paises se han quedado sin identificar?

```
waste_world %>%
  filter(is.na(continent)) %>%
  pull(country, iso_a3)
```

```
## filter: removed 212 rows (98%), 5 rows remaining
```

```
##                CHI                GIB                TUV                TWN                XKX
## "Channel Islands"       "Gibraltar"          "Tuvalu"                 NA          "Kosovo"
```

Buscar los paises que faltan en el dataset de continente:

```
continent %>%
  filter(country_name %in%
           c("Channel Islands", "Gibraltar", "Tuvalu", "Kosovo", "Taiwan"))
```

```
## filter: removed 239 rows (99%), 2 rows remaining
```

```
## # A tibble: 2 × 3
##   iso_a3 country_name continent
##   <chr>  <chr>        <chr>
## 1 <NA>   Kosovo       <NA>
## 2 <NA>   Taiwan       <NA>
```

library(stringr)

Allison Horst Illustration

Utilidades de `library(stringr)`

- `str_length()` - Longitud de una cadena
- `str_detect()` - Detecta un determinado patrón
- `str_extract()` - Extrae un determinado patrón
- `str_c()` - Encadena caracteres (similar a `paste0()`)
- `str_sub()` - Extrae sub-caracteres de una cadena
- `str_replace()` - Reemplaza caracter(es) por otro(s)
- `str_to_lower()`, `str_to_upper()`, `str_to_title()` - transformar en mayúsculas o minúsculas

Usando `library(stringr)`

Alternativa para buscar los paises que faltan en el dataset de continente:

```
continent %>%
  filter(str_detect(country_name, "Kosovo|Gibraltar|Tuvalu|Channel Islands|Taiwan"))
```

```
## filter: removed 239 rows (99%), 2 rows remaining
```

```
## # A tibble: 2 × 3
##   iso_a3 country_name continent
##   <chr>  <chr>        <chr>
## 1 <NA>   Kosovo       <NA>
## 2 <NA>   Taiwan       <NA>
```

Otro ejemplo - buscar Islas:

```
continent %>%
  filter(str_detect(country_name, "Island"))
```

```
## filter: removed 224 rows (93%), 17 rows remaining
```

```
## # A tibble: 17 × 3
##    iso_a3 country_name                          continent
##    <chr>  <chr>                                 <chr>
##  1 ALA    Aland Islands                         Europe
##  2 <NA>   Ashmore and Cartier Islands           Oceania
##  3 COK    Cook Islands                          Oceania
##  4 CYM    Cayman Islands                        North America
##  5 <NA>   Falkland Islands                      <NA>
##  6 FRO    Faeroe Islands                        Europe
##  7 HMD    Heard I. and McDonald Islands         Seven seas (open ocean)
##  8 MHL    Marshall Islands                      Oceania
##  9 MNP    Northern Mariana Islands              Oceania
## 10 NFK    Norfolk Island                        Oceania
## 11 PCN    Pitcairn Islands                      Oceania
## 12 SGS    South Georgia and South Sandwich Islands Seven seas (open ocean)
## 13 SLB    Solomon Islands                       Oceania
## 14 TCA    Turks and Caicos Islands              North America
## 15 VGB    British Virgin Islands                North America
```

Corregir un dato:

```
continent_corrected <- continent %>%
  mutate(iso_a3 = ifelse(country_name == "Kosovo", "XKX", iso_a3)) %>%
  mutate(iso_a3 = ifelse(country_name == "Taiwan", "TWN", iso_a3))
```

```
## mutate: changed one value (<1%) of 'iso_a3' (1 fewer NA)
## mutate: changed one value (<1%) of 'iso_a3' (1 fewer NA)
```

Volver a combinar dataset:

```
waste_world <- waste_regions %>%
  rename(iso_a3 = iso3c) %>%
  left_join(continent_corrected, by="iso_a3")
```

```
## rename: renamed one variable (iso_a3)

## left_join: added 2 columns (country_name, continent)

##            > rows only in x     3

##            > rows only in y  ( 27)

##            > matched rows     214

##            >                 =====
```

Ver que paises se han quedado fuera:

```
setdiff(waste_world$iso_a3, continent_corrected$iso_a3)
```

```
## [1] "CHI" "GIB" "TUV"
```

Vemos que países están en waste_world y que no están en continent_corrected

Ver que paises se han quedado fuera:

Si cambiamos el orden de los datos, vemos que países están en continent_corrected que no están en waste_world

```
setdiff(continent_corrected$iso_a3, waste_world$iso_a3)
```

```
##  [1] "AIA" "ALA" "ATA" NA    "ATF" "BLM" "COK" "GGY" "HMD" "JEY" "MSR" "NFK" "NIU" "PCN" "PRK" "SGS"
## [17] "SHN" "SPM" "VAT" "WLF"
```

Ver que paises hay en común:

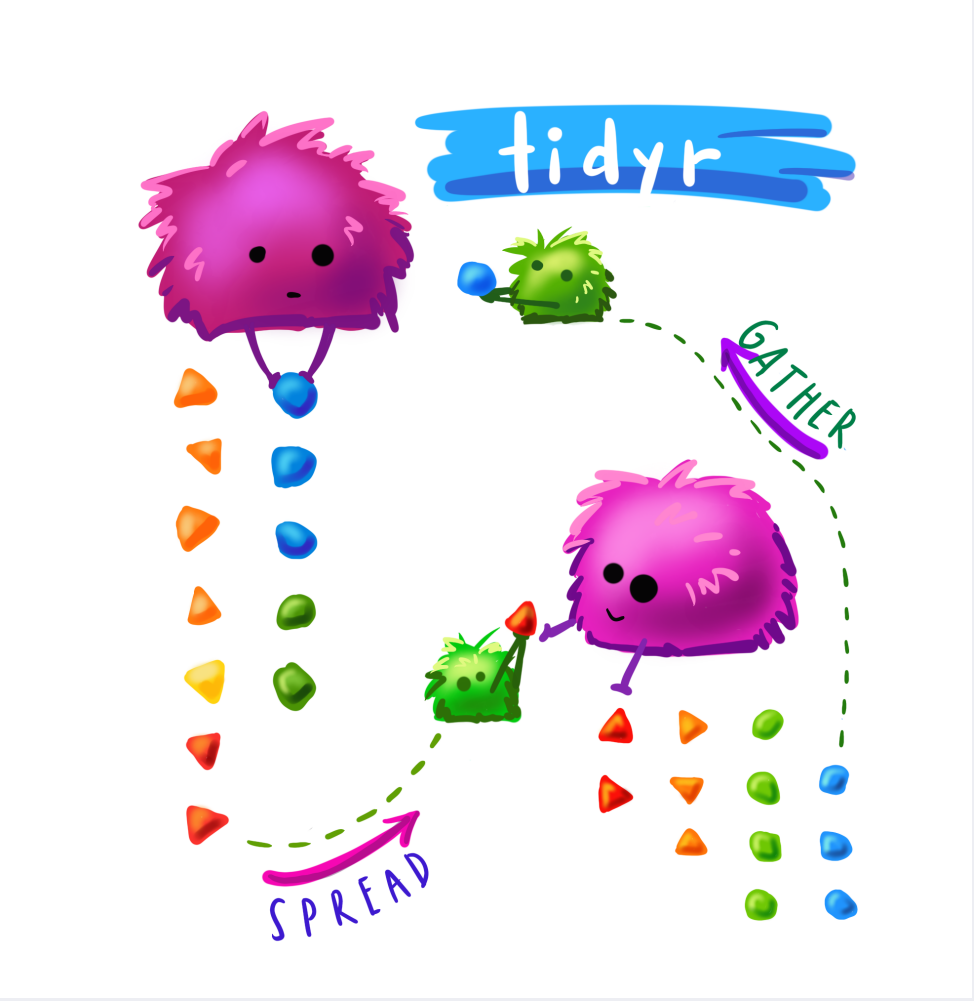
```
intersect(waste_world$iso_a3, continent_corrected$iso_a3)
```

```
##   [1] "ABW" "AFG" "AGO" "ALB" "AND" "ARE" "ARG" "ARM" "ASM" "ATG" "AUS" "AUT" "AZE" "BDI" "BEL" "BEN"
##  [17] "BFA" "BGD" "BGR" "BHR" "BHS" "BIH" "BLR" "BLZ" "BMU" "BOL" "BRA" "BRB" "BRN" "BTN" "BWA" "CAF"
##  [33] "CAN" "CHE" "CHL" "CHN" "CIV" "CMR" "COD" "COG" "COL" "COM" "CPV" "CRI" "CUB" "CUW" "CYM" "CYP"
##  [49] "CZE" "DEU" "DJI" "DMA" "DNK" "DOM" "DZA" "ECU" "EGY" "ERI" "ESP" "EST" "ETH" "FIN" "FJI" "FRA"
##  [65] "FRO" "FSM" "GAB" "GBR" "GEO" "GHA" "GIN" "GMB" "GNB" "GNQ" "GRC" "GRD" "GRL" "GTM" "GUM" "GUY"
##  [81] "HKG" "HND" "HRV" "HTI" "HUN" "IDN" "IMN" "IND" "IRL" "IRN" "IRQ" "ISL" "ISR" "ITA" "JAM" "JOR"
##  [97] "JPN" "KAZ" "KEN" "KGZ" "KHM" "KIR" "KNA" "KOR" "KWT" "LAO" "LBN" "LBR" "LBY" "LCA" "LIE" "LKA"
## [113] "LSO" "LTU" "LUX" "LVA" "MAC" "MAF" "MAR" "MCO" "MDA" "MDG" "MDV" "MEX" "MHL" "MKD" "MLI" "MLT"
## [129] "MMR" "MNE" "MNG" "MNP" "MOZ" "MRT" "MUS" "MWI" "MYS" "NAM" "NCL" "NER" "NGA" "NIC" "NLD" "NOR"
## [145] "NPL" "NRU" "NZL" "OMN" "PAK" "PAN" "PER" "PHL" "PLW" "PNG" "POL" "PRI" "PRT" "PRY" "PSE" "PYF"
## [161] "QAT" "ROU" "RUS" "RWA" "SAU" "SDN" "SEN" "SGP" "SLB" "SLE" "SLV" "SMR" "SOM" "SRB" "SSD" "STP"
## [177] "SUR" "SVK" "SVN" "SWE" "SWZ" "SXM" "SYC" "SYR" "TCA" "TCD" "TGO" "THA" "TJK" "TKM" "TLS" "TON"
## [193] "TTO" "TUN" "TUR" "TWN" "TZA" "UGA" "UKR" "URY" "USA" "UZB" "VCT" "VEN" "VGB" "VIR" "VNM" "VUT"
## [209] "WSM" "XKX" "YEM" "ZAF" "ZMB" "ZWE"
```

```
glimpse(waste_world)
```

```
## Rows: 217
## Columns: 18
## $ iso_a3                                      <chr> "ABW", "AFG", "AGO", "ALB", "AND", "ARE", "ARG"…
## $ region_id                                   <chr> "Latin_America", "South_Asia", "Sub-Saharan_Afr…
## $ country                                     <chr> "Aruba", "Afghanistan", "Angola", "Albania", "A…
## $ income_id                                   <chr> "HIC", "LIC", "LMC", "UMC", "HIC", "HIC", "HIC"…
## $ gdp                                         <dbl> 35563.3125, 2057.0623, 8036.6904, 13724.0586, 4…
## $ population                                  <dbl> 103187, 34656032, 25096150, 2854191, 82431, 977…
## $ total_waste                                 <dbl> 88132.02, 5628525.37, 4213643.58, 1087446.75, 4…
## $ composition_food_organic_waste_percent      <dbl> NA, NA, 51.80000, 51.40000, 31.20000, 39.00000,…
## $ composition_glass_percent                   <dbl> NA, NA, 6.700000, 4.500000, 8.200000, 4.000000,…
## $ composition_metal_percent                   <dbl> NA, NA, 4.400000, 4.800000, 2.600000, 3.000000,…
## $ composition_other_percent                   <dbl> NA, NA, 11.5000, 15.2100, 11.6000, 10.0000, 15.…
## $ composition_paper_cardboard_percent         <dbl> NA, NA, 11.900000, 9.900000, 35.100000, 25.0000…
## $ composition_plastic_percent                 <dbl> NA, NA, 13.50000, 9.60000, 11.30000, 19.00000, …
## $ composition_rubber_leather_percent          <dbl> NA, NA, NA, NA, NA, NA, 1.26, 0.30, 4.20, NA, N…
## $ composition_wood_percent                    <dbl> NA, NA, NA, 4.60, NA, NA, 1.09, 0.60, NA, NA, N…
## $ composition_yard_garden_green_waste_percent <dbl> NA, NA, NA, NA, NA, NA, 9.95, NA, NA, NA, NA, N…
## $ country_name                                <chr> "Aruba", "Afghanistan", "Angola", "Albania", "A…
## $ continent                                   <chr> "North America", "Asia", "Africa", "Europe", "E…
```

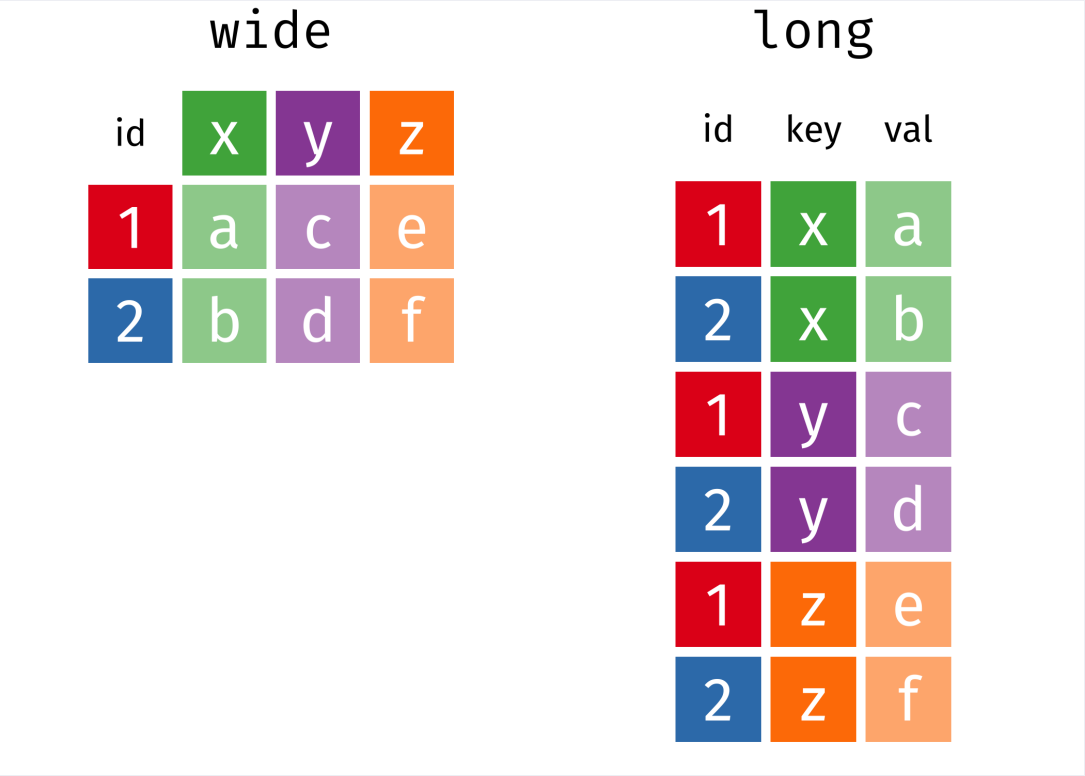
library(tidyr)



Allison Horst Illustration

Reestructurar dataset con `library(tidyr)`

- `pivot_wider()` o `spread()`
- `pivot_longer()` o `gather()`



Fuente: Garrick Aden-Buie's - Tidyexplained Verbs

Reestructurar el dataset con `library(tidyr)`

```
glimpse(waste_world)
```

```
## Rows: 217
## Columns: 18
## $ iso_a3                                      <chr> "ABW", "AFG", "AGO", "ALB", "AND", "ARE", "ARG"…
## $ region_id                                   <chr> "Latin_America", "South_Asia", "Sub-Saharan_Afr…
## $ country                                     <chr> "Aruba", "Afghanistan", "Angola", "Albania", "A…
## $ income_id                                   <chr> "HIC", "LIC", "LMC", "UMC", "HIC", "HIC", "HIC"…
## $ gdp                                         <dbl> 35563.3125, 2057.0623, 8036.6904, 13724.0586, 4…
## $ population                                  <dbl> 103187, 34656032, 25096150, 2854191, 82431, 977…
## $ total_waste                                 <dbl> 88132.02, 5628525.37, 4213643.58, 1087446.75, 4…
## $ composition_food_organic_waste_percent      <dbl> NA, NA, 51.80000, 51.40000, 31.20000, 39.00000,…
## $ composition_glass_percent                   <dbl> NA, NA, 6.700000, 4.500000, 8.200000, 4.000000,…
## $ composition_metal_percent                   <dbl> NA, NA, 4.400000, 4.800000, 2.600000, 3.000000,…
## $ composition_other_percent                   <dbl> NA, NA, 11.5000, 15.2100, 11.6000, 10.0000, 15.…
## $ composition_paper_cardboard_percent         <dbl> NA, NA, 11.900000, 9.900000, 35.100000, 25.0000…
## $ composition_plastic_percent                 <dbl> NA, NA, 13.50000, 9.60000, 11.30000, 19.00000, …
## $ composition_rubber_leather_percent          <dbl> NA, NA, NA, NA, NA, NA, 1.26, 0.30, 4.20, NA, N…
## $ composition_wood_percent                    <dbl> NA, NA, NA, 4.60, NA, NA, 1.09, 0.60, NA, NA, N…
## $ composition_yard_garden_green_waste_percent <dbl> NA, NA, NA, NA, NA, NA, 9.95, NA, NA, NA, NA, N…
## $ country_name                                <chr> "Aruba", "Afghanistan", "Angola", "Albania", "A…
## $ continent                                   <chr> "North America", "Asia", "Africa", "Europe", "E…
```

Reestructurar el dataset con `library(tidyr)`

```
composition <- waste_world %>%
  pivot_longer(cols = starts_with("composition"), names_to = "composition", values_to = "percent")
```

```
## pivot_longer: reorganized (composition_food_organic_waste_percent, composition_glass_percent, compositi
```

```
composition %>%
  select(country, composition, percent)
```

```
## select: dropped 8 variables (iso_a3, region_id, income_id, gdp, population, …)

## # A tibble: 1,953 × 3
##    country     composition                                  percent
##    <chr>       <chr>                                          <dbl>
##  1 Aruba       composition_food_organic_waste_percent            NA
##  2 Aruba       composition_glass_percent                         NA
##  3 Aruba       composition_metal_percent                         NA
##  4 Aruba       composition_other_percent                         NA
##  5 Aruba       composition_paper_cardboard_percent               NA
##  6 Aruba       composition_plastic_percent                       NA
##  7 Aruba       composition_rubber_leather_percent                NA
##  8 Aruba       composition_wood_percent                          NA
##  9 Aruba       composition_yard_garden_green_waste_percent       NA
## 10 Afghanistan composition_food_organic_waste_percent            NA
## # … with 1,943 more rows
```

```
composition %>%
  select(country, composition, percent) %>%
  arrange(desc(country))
```

```
## select: dropped 8 variables (iso_a3, region_id, income_id, gdp, population, …)

## # A tibble: 1,953 × 3
##    country  composition                                  percent
##    <chr>    <chr>                                          <dbl>
##  1 Zimbabwe composition_food_organic_waste_percent            36
##  2 Zimbabwe composition_glass_percent                          5
##  3 Zimbabwe composition_metal_percent                          6
##  4 Zimbabwe composition_other_percent                          3
##  5 Zimbabwe composition_paper_cardboard_percent               27
##  6 Zimbabwe composition_plastic_percent                       23
##  7 Zimbabwe composition_rubber_leather_percent                NA
##  8 Zimbabwe composition_wood_percent                          NA
##  9 Zimbabwe composition_yard_garden_green_waste_percent       NA
## 10 Zambia   composition_food_organic_waste_percent            NA
## # … with 1,943 more rows
```

Simplificar variables con `library(stringr)`:

```
composition_fix <- composition %>%
  mutate(composition=str_remove(composition, "composition_")) %>%
  mutate(composition=str_remove(composition, "_percent"))
```

```
## mutate: changed 1,953 values (100%) of 'composition' (0 new NA)
## mutate: changed 1,953 values (100%) of 'composition' (0 new NA)
```

```
distinct(composition_fix, composition)
```

```
## distinct: removed 1,944 rows (>99%), 9 rows remaining

## # A tibble: 9 × 1
##   composition
##   <chr>
## 1 food_organic_waste
## 2 glass
## 3 metal
## 4 other
## 5 paper_cardboard
## 6 plastic
## 7 rubber_leather
## 8 wood
## 9 yard_garden_green_waste
```

```
glimpse(composition_fix)
```

```
## Rows: 1,953
## Columns: 11
## $ iso_a3       <chr> "ABW", "ABW", "ABW", "ABW", "ABW", "ABW", "ABW", "ABW", "ABW", "AFG", "AFG", "…
## $ region_id    <chr> "Latin_America", "Latin_America", "Latin_America", "Latin_America", "Latin_Ame…
## $ country      <chr> "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba…
## $ income_id    <chr> "HIC", "HIC", "HIC", "HIC", "HIC", "HIC", "HIC", "HIC", "HIC", "LIC", "LIC", "…
## $ gdp          <dbl> 35563.312, 35563.312, 35563.312, 35563.312, 35563.312, 35563.312, 35563.312, 3…
## $ population   <dbl> 103187, 103187, 103187, 103187, 103187, 103187, 103187, 103187, 103187, 346560…
## $ total_waste  <dbl> 88132.02, 88132.02, 88132.02, 88132.02, 88132.02, 88132.02, 88132.02, 88132.02…
## $ country_name <chr> "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba…
## $ continent    <chr> "North America", "North America", "North America", "North America", "North Ame…
## $ composition  <chr> "food_organic_waste", "glass", "metal", "other", "paper_cardboard", "plastic",…
## $ percent      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 51.80,…
```

```r
composition_clean <- composition_fix %>%
  select(-country_name) %>%
  relocate(continent, .before = region_id) %>%
  arrange(continent, country)
```

```
## select: dropped one variable (country_name)

## relocate: columns reordered (iso_a3, continent, region_id, country, income_id, …)
```

```
glimpse(composition_clean)
```

```
## Rows: 1,953
## Columns: 10
## $ iso_a3      <chr> "DZA", "DZA", "DZA", "DZA", "DZA", "DZA", "DZA", "DZA", "DZA", "AGO", "AGO", "A…
## $ continent   <chr> "Africa", "Africa", "Africa", "Africa", "Africa", "Africa", "Africa", "Africa",…
## $ region_id   <chr> "Middle_East_North_Africa", "Middle_East_North_Africa", "Middle_East_North_Afri…
## $ country     <chr> "Algeria", "Algeria", "Algeria", "Algeria", "Algeria", "Algeria", "Algeria", "A…
## $ income_id   <chr> "UMC", "UMC", "UMC", "UMC", "UMC", "UMC", "UMC", "UMC", "UMC", "LMC", "LMC", "L…
## $ gdp         <dbl> 11826.165, 11826.165, 11826.165, 11826.165, 11826.165, 11826.165, 11826.165, 11…
## $ population  <dbl> 40606052, 40606052, 40606052, 40606052, 40606052, 40606052, 40606052, 40606052,…
## $ total_waste <dbl> 12378740.0, 12378740.0, 12378740.0, 12378740.0, 12378740.0, 12378740.0, 1237874…
## $ composition <chr> "food_organic_waste", "glass", "metal", "other", "paper_cardboard", "plastic", …
## $ percent     <dbl> 54.4, 1.2, 2.8, 0.8, 9.8, 16.9, 12.6, 1.5, NA, 51.8, 6.7, 4.4, 11.5, 11.9, 13.5…
```

Guardar dataset para el próximo día:

```
write_csv(composition_clean, here("data/waste_world_composition.csv"))
```

Comprobar datos: asegurar que los porcentajes suman 100%

```
composition_clean %>%
  group_by(country) %>%
  summarise(per_sum = sum(percent, na.rm = TRUE))
```

```
## group_by: one grouping variable (country)

## summarise: now 217 rows and 2 columns, ungrouped

## # A tibble: 217 × 2
##    country             per_sum
##    <chr>                 <dbl>
##  1 Afghanistan             0
##  2 Albania               100.
##  3 Algeria               100
##  4 American Samoa        100
##  5 Andorra               100
##  6 Angola                 99.8
##  7 Antigua and Barbuda   100
##  8 Argentina             100.
##  9 Armenia               100
## 10 Aruba                   0
## # … with 207 more rows
```

Comprobar datos: Lista de paises cuyos porcentajes no suman 100%

```
composition_clean %>%
  group_by(country) %>%
  summarise(per_sum = sum(percent, na.rm = TRUE)) %>%
  filter(per_sum < 99.9 | per_sum > 100.1)  %>% # arrange(per_sum) %>%
  pull(per_sum, country)
```

```
##              Afghanistan                   Angola                    Aruba 
##                     0.00                    99.80                     0.00 
##                  Belgium                 Botswana               Cabo Verde 
##                    99.86                   100.10                     0.00 
## Central African Republic          Channel Islands          Congo, Dem. Rep. 
##                     0.00                     0.00                     0.00 
##            Côte d'Ivoire                  Curacao                 Djibouti 
##                     0.00                     0.00                     0.00 
##                  Eritrea                 Eswatini           Faeroe Islands 
##                     0.00                     0.00                     0.00 
##                Gibraltar           Guinea-Bissau                    India 
##                    97.25                     0.00                     0.00 
##      Iran, Islamic Rep.                     Iraq             Isle of Man 
##                   100.30                    99.74                     0.00 
##              Korea, Rep.                   Kuwait          Kyrgyz Republic 
##                   100.40                   101.00                     0.00 
##                   Latvia                  Lesotho                  Liberia 
##                   100.20                     0.00                     0.00 
##                Lithuania       Macao SAR, China               Madagascar 
```

Crear dataset solo para los paises con información completa:

```
composition_complete <- composition_clean %>%
  group_by(country) %>%
  mutate(per_sum = sum(percent, na.rm = TRUE)) %>%
  filter(per_sum < 99.9 | per_sum > 100.1)
```

```
## group_by: one grouping variable (country)

## mutate (grouped): new variable 'per_sum' (double) with 42 unique values and 0% NA

## filter (grouped): removed 1,341 rows (69%), 612 rows remaining
```

Atención, en este caso para filtrar el dataset usamos `mutate()` y no `summarise()`, ya que queremos toda la informacion desagregada.

Ejercicio 1:

Calcular composición de basura en España:

Ejercicio 1:

Calcular composición de basura en España:

```
composition_complete %>%
  filter(country == "Spain") %>%
  select(composition, percent)
```

```
## # A tibble: 0 × 3
## # Groups:   country [0]
## # … with 3 variables: country <chr>, composition <chr>, percent <dbl>
```

Ejercicio 2:

Usando el dataset waste_world - calcular la población media por regiones en millones de habitantes.

Ejercicio 2:

Usando el dataset waste_world - calcular la población media por regiones en millones de habitantes.

```
waste_world %>%
  group_by(region_id) %>%
  summarise(pop = mean(population, na.rm = TRUE)) %>%
  mutate(pop = pop / 1000000)
```

```
## # A tibble: 7 × 2
##   region_id                 pop
##   <chr>                   <dbl>
## 1 East_Asia_Pacific        61.6
## 2 Europe_Central_Asia      15.7
## 3 Latin_America            15.0
## 4 Middle_East_North_Africa 20.2
## 5 North_America           121.
## 6 South_Asia              223.
## 7 Sub-Saharan_Africa       18.9
```

Ejercicio 3:

Crear una variable basada en el nivel de basura per cápita y contar el numero de paises en cada grupo.

Ejemplo:

- high_waste = >0.6 toneladas de basura por persona al año
- medium_waste = 0.2 a 0.6 toneladas de basura por persona al año
- low_waste = <0.2 toneladas de basura por persona al año

Ejercicio 3:

Crear una variable basada en el nivel de basura per cápita y contar el número de paises en cada grupo.

- high_waste = >0.6 toneladas de basura por persona al año
- medium_waste = 0.2 a 0.6 toneladas de basura por persona al año
- low_waste = <0.2 toneladas de basura por persona al año

```
waste_world %>%
  select(country, population, total_waste) %>%
  mutate(waste_per_pers = total_waste / population) %>%
  mutate(waste_levels = case_when(
              waste_per_pers >= 0.6 ~ "high_waste",
              waste_per_pers <= 0.2 ~ "low_waste",
              waste_per_pers < 0.6 & waste_per_pers > 0.2 ~ "medium_waste")) %>%
  group_by(waste_levels) %>%
  summarise(n_countries = n())
```

```
## # A tibble: 4 × 2
##   waste_levels n_countries
##   <chr>              <int>
## 1 high_waste            34
## 2 low_waste             65
## 3 medium_waste         116
## 4 <NA>                   2
```

Ejercicio 4:

Usando la nueva categoria de niveles de basura, contar paises por region y crear una tabla como esta:

region_id	high_waste	low_waste	medium_waste
East_Asia_Pacific	6	10	21
Europe_Central_Asia	14	5	39
Latin_America	8	4	28
Middle_East_North_Africa	3	3	15
North_America	3	0	0
South_Asia	0	7	1
Sub-Saharan_Africa	0	36	12

Ejercicio 4:

Usando la nueva categoria de niveles de basura, contar paises por region y crear una tabla como esta:

```
waste_world %>%
  select(region_id, country, population, total_waste) %>%
  mutate(waste_per_pers = total_waste/population) %>%
  filter(!is.na(waste_per_pers)) %>%
  mutate(waste_levels = case_when(
              waste_per_pers >= 0.6 ~ "high_waste",
              waste_per_pers <= 0.2 ~ "low_waste",
              waste_per_pers < 0.6 & waste_per_pers > 0.2 ~ "medium_waste")) %>%
  group_by(region_id, waste_levels) %>%
  summarise(n_countries=n()) %>%
  pivot_wider(names_from=waste_levels, values_from=n_countries) %>%
  replace(is.na(.), 0) %>%
  kable()
```

Ejercicio 5:

Usando el dataset waste_world - calcular el residuo de basura plástica en millones de toneladas (composition_plastic_percent * total_waste) por continente y ordenarlo de mayor a menor.

Ejercicio 5:

Usando el dataset waste_world - calcular el residuo de basura plástica en millones de toneladas (composition_plastic_percent * total_waste) por continente y ordenarlo de mayor a menor.

```
waste_world %>%
  select(continent,
         total_waste,
         plastic_per=composition_plastic_percent) %>%
  mutate(plastic_waste=total_waste*plastic_per) %>%
  group_by(continent) %>%
  summarise(plastic = sum(plastic_waste, na.rm=T)/1000000) %>%
  arrange(desc(plastic))
```

```
## # A tibble: 8 × 2
##   continent               plastic
##   <chr>                     <dbl>
## 1 Asia                    8417.
## 2 North America           4387.
## 3 Europe                  4258.
## 4 South America           1735.
## 5 Africa                  1266.
## 6 <NA>                     152.
## 7 Oceania                  152.
## 8 Seven seas (open ocean)    6.54
```

Recursos

- Tidyverse packages
- R for Data Science Book - Wrangle Chapter
- RStudio CheatSheets
    - Data import with readr, readxl, and googlesheets4
    - Data Transformation with dplyr
    - Data tidying with tidyr
    - String manipulation with stringr
    - Factors with forcats
    - Dates and times with lubridate