

Importare i dati con protocollo SDMX

L'Istat e altri provider di statistiche pubbliche diffondono i dati anche in modalità *machine-to-machine*. In pratica, con R, è possibile scaricare i dati mediante *query* utilizzando il pacchetto *rsdmx*.

Installiamo il pacchetto:

```
install.packages("rsdmx")
library(rsdmx)
library(tidyverse)
```

Lista dei dataset disponibili

Per avere la lista dei dataset (flussi di dati, per l'esattezza) disponibili, dobbiamo conoscere l'identificativo del provider:

```
istat.fl <- readSDMX(providerId = "ISTAT", resource = "dataflow") %>%
  as.data.frame()
```

Cerchiamo i dati che ci interessano, ad esempio con una ricerca per parola chiave all'interno del testo (vedi [Ricerca e sostituzione di testo](#)):

```
istat.fl %>%
  filter(str_detect(Name.it, "coppi"))
```

	id	agencyID	Name.en
			Name.it ...
1	174_65	IT1	Distribution of household work of couples Distribuzione del carico di lavoro nelle coppie ...
2	174_80	IT1	Daily activities - people in couple Attività quotidiane - persone in coppia ...

L'identificativo di Eurostat è "ESTAT", quello dell'Oecd è "OECD".

Scaricare il dataset

Il dataset che mi interessa, *Distribuzione del carico di lavoro nelle coppie*, ha l'identificativo 174_65.

Lo scarico (`readSDMX()`) con i relativi metadati (argomento `dsd = TRUE`) e lo trasformo in

dataframe con le etichette delle variabili e delle modalità (`as.data.frame(labels = TRUE)`):

```
contr.genere <- readSDMX(providerId = "ISTAT", resource = "data",
  flowRef = "174_65",
  # Data Structure Definition (metadati)
  dsd = TRUE) %>%
  # dataframe con le etichette, importate con dsd
  as.data.frame(labels = TRUE)
```

Visualizziamo il dataframe in forma compatta, come [tibble](#):

```
as_tibble(contr.genere)
```

```
# A tibble: 16,122 x 35
  FREQ  FREQ_label.en  FREQ_label.it  POSPROF_D  POSPROF_D_label.en
POSPROF_D_label~
  <chr> <chr>          <chr>          <chr>      <chr>
<chr>
1 A    annual        annuale        1          employed person
occupato
2 A    annual        annuale        1          employed person
occupato
3 A    annual        annuale        1          employed person
occupato
4 A    annual        annuale        1          employed person
occupato
5 A    annual        annuale        1          employed person
occupato
6 A    annual        annuale        1          employed person
occupato
7 A    annual        annuale        1          employed person
occupato
8 A    annual        annuale        1          employed person
occupato
9 A    annual        annuale        1          employed person
occupato
10 A   annual        annuale        1          employed person
occupato
# ... with 16,112 more rows, and 29 more variables: CONDLAV_CO <chr>,
#   CONDLAV_CO_label.en <chr>, CONDLAV_CO_label.it <chr>, ETA_DONNA
<chr>,
#   ETA_DONNA_label.en <chr>, ETA_DONNA_label.it <chr>, ETA_FIGLIOPIC
<chr>,
#   ETA_FIGLIOPIC_label.en <chr>, ETA_FIGLIOPIC_label.it <chr>,
ITTER107 <chr>,
```

```
# ITTER107_label.en <chr>, ITTER107_label.it <chr>, NUM_FIGLI
<chr>,
# NUM_FIGLI_label.en <chr>, NUM_FIGLI_label.it <chr>,
# COPPIA_CONSENZA_FIGLI <chr>, COPPIA_CONSENZA_FIGLI_label.en
<chr>, ...
```

Filtrare i dati

E' possibile filtrare i dati, utilizzando l'argomento `key` nella funzione `readSDMX()`. A tale scopo, dobbiamo conoscere lo schema dei dati, a prescindere dalle etichette, che moltiplicano i campi.

```
readSDMX(providerId = "ISTAT", resource = "data",
          flowRef = "174_65",
          dsd = T) %>% as.data.frame() %>% names()
```

```
names(contr.genere.s)
[1] "FREQ"           "POSPROF_D"      "CONDLAV_CO"
[4] "ETA_DONNA"      "ETA_FIGLIOPIC"  "ITTER107"
[7] "NUM_FIGLI"      "COPPIA_CONSENZA_FIGLI" "TIPO_DATO"
[10] "TITOLO_STUDIO_U" "TITOLO_STUDIO_D" "obsTime"
[13] "obsValue"
```

Volendo scaricare solo i dati del 2013, indicheremo il valore 2013 al dodicesimo posto, in questo modo:

```
key = ".....2013."
```

Il valore di `key` è una stringa con tanti punti quanti sono i campi (senza le etichette): al posto del punto, nei campi che interessano, viene inserito un valore (2013) della variabile ("obsTime": dodicesimo posto). Volendo solo i dati relativi alle coppie con figli (valore 11 per la variabile "COPPIA_CONSENZA_FIGLI", ottavo posto), nel 2013:

```
key = ".....11...2013."
```

Quindi

```
readSDMX(providerId = "ISTAT", resource = "data",
          flowRef = "174_65",
          key = ".....11...2013.",
          dsd = T)
```

Ho caricato una funzione [su GitHub](https://www.agnesevardanega.eu/wiki/) per recuperare le informazioni sui flussi di dati ed organizzarle in due dataframe: uno per i campi e uno per le etichette (`cod_sdmx()`); cfr.

anche il  [post sul blog](#)).

Vedi anche

- [SDMX Istat: Tutti i dataflows \(flussi di dati\)](#)
- [rsdmx quickstart guide](#)
- [Datalab: Leggere la statistica pubblica con R](#)
- [Datalab: Lanciare una query SDMX sui server Istat](#)
- [Ondata: Guida all'uso delle API REST di ISTAT](#)
- [Istat: Web service SDMX per l'accesso ai dati di I.Stat](#)
- [Eurostat: Brief explanation about the Eurostat data and the formats of data](#)

Script di esempio

[rsdmx_istat.R](#)

```
# installazione
install.packages("rsdmx")

# pacchetti
library(rsdmx)
library(tidyverse)

# elenco dei dataset disponibili su Istat
istat.fl <- readSDMX(providerId = "ISTAT", resource = "dataflow")
%>%
  as.data.frame()

# ricerca per nome del dataset
istat.fl %>%
  filter(str_detect(Name.it, "coppi"))

# download del dataset
sdmx.data <- readSDMX(providerId = "ISTAT", resource = "data",
                      flowRef = "174_65",
                      dsd = T)

# dataframe con etichette
contr.genere <- as.data.frame(sdmx.data,
                              labels = TRUE)

# tibble senza etichette
contr.genere.s <- as_tibble(sdmx.data)
```

```
# con filtro  
readSDMX(providerId = "ISTAT", resource = "data",  
          flowRef   = "174_65",  
          key       = ".....11...2013.",  
          dsd       = T)
```

Gestione dei dati

From:

<https://www.agnesevardanega.eu/wiki/> - **Ricerca Sociale con R**

Permanent link:

https://www.agnesevardanega.eu/wiki/r/gestione_dei_dati/importare_sdmx

Last update: **10/09/2025 06:30**

