

# Package: istatR (via r-universe)

June 3, 2026

**Title** Interface to the Italian National Institute of Statistics (ISTAT') API

**Version** 0.1.0

**Description** Provides an interface to the 'ISTAT' 'SDMX' RESTful API <<https://esploradati.istat.it/SDMXWS/rest/>>. Allows users to discover available datasets, explore their structure and dimensions, and retrieve statistical data from the Italian National Institute of Statistics. Based on the Python 'istatapi' package by Jacopo Attolini.

**License** Apache License (>= 2)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.3

**Imports** httr2, xml2, dplyr, tibble, stringr, purrr, readr, rlang

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**URL** <https://github.com/jfulponi/istatR>

**BugReports** <https://github.com/jfulponi/istatR/issues>

**Config/pak/sysreqs** libicu-dev libxml2-dev libssl-dev libx11-dev

**Repository** <https://jfulponi.r-universe.dev>

**Date/Publication** 2026-02-03 16:31:57 UTC

**RemoteUrl** <https://github.com/jfulponi/istatr>

**RemoteRef** HEAD

**RemoteSha** 21669834770c8d2a873eb135d42f9587a3d1ac62

## Contents

all_available . . . . .	2
base . . . . .	3

dimensions_info . . . . .	3
discovery . . . . .	3
get_available_values . . . . .	4
get_data . . . . .	4
get_dimension_values . . . . .	5
istat_dataset . . . . .	6
istat_get . . . . .	7
istat_timeout . . . . .	8
print.istat_dataset . . . . .	9
reset_filters . . . . .	9
retrieval . . . . .	10
search_dataset . . . . .	10
set_filters . . . . .	11
utils . . . . .	11
<b>Index</b>	<b>12</b>

---

all_available	<i>List all available ISTAT datasets</i>
---------------	--

---

## Description

Retrieves a list of all available datasets (dataflows) from the ISTAT API.

## Usage

```
all_available()
```

## Value

A tibble with the following columns:

**df\_id** Dataflow ID  
**version** Dataset version  
**df\_description** English description of the dataset  
**df\_structure\_id** Data structure definition ID

## Examples

```
## Not run:
# Get all available datasets
datasets <- all_available()
head(datasets)

## End(Not run)
```

---

base	<i>ISTAT API Base Functions</i>
------	---------------------------------

---

**Description**

Core functions for communicating with the ISTAT SDMX REST API

---

dimensions_info	<i>Get information about dataset dimensions</i>
-----------------	---

---

**Description**

Returns information about the dimensions of a dataset, including their positions and associated codelists.

**Usage**

```
dimensions_info(dataset, include_descriptions = TRUE)
```

**Arguments**

dataset	An <code>istat_dataset</code> object
include_descriptions	Logical; whether to include dimension descriptions (default: TRUE)

**Value**

A tibble with dimension information

**Examples**

```
## Not run:  
ds <- istat_dataset("139_176")  
dimensions_info(ds)  
  
## End(Not run)
```

---

discovery	<i>ISTAT Dataset Discovery Functions</i>
-----------	--

---

**Description**

Functions for discovering and exploring ISTAT datasets

get\_available\_values *Get all available values for all dimensions*

---

### Description

Uses the availableconstraint endpoint to get all valid values for each dimension. This is more accurate than getting values from codelists as it reflects actual data availability.

### Usage

```
get_available_values(dataset)
```

### Arguments

dataset            An istat\_dataset object

### Value

A named list where each element contains a tibble of available values for that dimension

### Examples

```
## Not run:
ds <- istat_dataset("139_176")
available <- get_available_values(ds)
available$FREQ # Available frequency values

## End(Not run)
```

---

get\_data            *Retrieve data from an ISTAT dataset*

---

### Description

Fetches data from an ISTAT dataset using the currently set filters. The data is returned as a tibble with the TIME\_PERIOD column converted to Date format and sorted in ascending order.

### Usage

```
get_data(
  dataset,
  start_period = NULL,
  end_period = NULL,
  last_n_observations = NULL
)
```

**Arguments**

**dataset** An `istat_dataset` object with filters set  
**start\_period** Optional start date for filtering (format: YYYY-MM-DD or YYYY)  
**end\_period** Optional end date for filtering (format: YYYY-MM-DD or YYYY)  
**last\_n\_observations** Optional integer to get only the last N observations

**Value**

A tibble containing the requested data with columns including:

**DATAFLOW** Dataset identifier  
**FREQ** Frequency  
**TIME\_PERIOD** Time period (as Date)  
**OBS\_VALUE** Observation value  
 ... Additional dimension and metadata columns

**Examples**

```

## Not run:
# Create and configure dataset
ds <- istat_dataset("139_176")
ds <- set_filters(ds,
  FREQ = "M",
  TIPO_DATO = c("ISAV", "ESAV"),
  PAESE_PARTNER = "WORLD"
)

# Get all data
data <- get_data(ds)

# Get data for a specific time range
data <- get_data(ds, start_period = "2020-01-01", end_period = "2023-12-31")

# Get only the last 12 observations
data <- get_data(ds, last_n_observations = 12)

## End(Not run)

```

---

get\_dimension\_values *Get available values for a dimension*

---

**Description**

Retrieves all available values for a specific dimension of a dataset.

**Usage**

```
get_dimension_values(dataset, dimension_id)
```

**Arguments**

```
dataset      An istat_dataset object
dimension_id The ID of the dimension
```

**Value**

A tibble with columns:

```
id Value ID/code
name Human-readable name (English)
```

**Examples**

```
## Not run:
ds <- istat_dataset("139_176")
get_dimension_values(ds, "TIPO_DATO")

## End(Not run)
```

---

istat_dataset	<i>Create an ISTAT dataset object</i>
---------------	---------------------------------------

---

**Description**

Creates a dataset object for a specific ISTAT dataflow. This object can be used to explore the dataset's structure, dimensions, and available values, and to set filters before retrieving data.

**Usage**

```
istat_dataset(dataflow_identifier)
```

**Arguments**

```
dataflow_identifier
  Either a dataflow ID (e.g., "139_176"), a structure ID, or an exact dataset description
```

**Value**

A list with class "istat\_dataset" containing:

**df\_id** Dataflow ID

**version** Dataset version

**df\_description** Dataset description

**df\_structure\_id** Data structure definition ID

**dimensions** Named list of dimension information

**filters** Named list of current filters (initialized to "." for all)

**Examples**

```
## Not run:
# Create dataset by ID
ds <- istat_dataset("139_176")

# View dimensions
dimensions_info(ds)

# Get available values for a dimension
get_dimension_values(ds, "TIPO_DATO")

# Set filters
ds <- set_filters(ds, FREQ = "M", TIPO_DATO = c("ISAV", "ESAV"))

## End(Not run)
```

---

istat\_get

*Quick data retrieval*

---

**Description**

A convenience function that combines creating a dataset, setting filters, and retrieving data in one call.

**Usage**

```
istat_get(
  dataflow_id,
  ...,
  start_period = NULL,
  end_period = NULL,
  last_n_observations = NULL
)
```

**Arguments**

dataflow\_id      Dataflow ID (e.g., "139\_176")  
 ...                Named filter arguments (dimension\_id = value)  
 start\_period      Optional start date  
 end\_period        Optional end date  
 last\_n\_observations  
                     Optional integer to get only the last N observations

**Value**

A tibble containing the requested data

**Examples**

```

## Not run:
# Quick retrieval with filters
data <- istat_get(
  "139_176",
  FREQ = "M",
  TIPO_DATO = "ISAV",
  PAESE_PARTNER = "WORLD",
  start_period = "2020-01-01"
)

## End(Not run)

```

---

istat_timeout	<i>Get or set the API timeout</i>
---------------	-----------------------------------

---

**Description**

The ISTAT API can be slow to respond, especially for large queries. This function allows you to get or set the timeout value in seconds. The default timeout is 300 seconds (5 minutes).

**Usage**

```
istat_timeout(seconds = NULL)
```

**Arguments**

seconds            Optional. If provided, sets the timeout to this value in seconds. If NULL (default), returns the current timeout value.

**Value**

If seconds is NULL, returns the current timeout value. If seconds is provided, invisibly returns the previous timeout value.

### Examples

```
# Get current timeout
istat_timeout()

# Set timeout to 10 minutes
istat_timeout(600)

# Set timeout back to default
istat_timeout(300)
```

---

`print.istat_dataset`     *Print method for istat\_dataset*

---

### Description

Print method for istat\_dataset

### Usage

```
## S3 method for class 'istat_dataset'
print(x, ...)
```

### Arguments

<code>x</code>	An istat_dataset object
<code>...</code>	Additional arguments (ignored)

### Value

Invisibly returns the input

---

`reset_filters`     *Reset all filters to default (all values)*

---

### Description

Reset all filters to default (all values)

### Usage

```
reset_filters(dataset)
```

### Arguments

<code>dataset</code>	An istat_dataset object
----------------------	-------------------------

**Value**

The modified `istat_dataset` object with all filters reset to "."

**Examples**

```
## Not run:
ds <- istat_dataset("139_176")
ds <- set_filters(ds, FREQ = "M")
ds <- reset_filters(ds) # All filters back to "."

## End(Not run)
```

---

retrieval	<i>ISTAT Data Retrieval Functions</i>
-----------	---------------------------------------

---

**Description**

Functions for retrieving data from ISTAT datasets

---

search_dataset	<i>Search for datasets by keyword</i>
----------------	---------------------------------------

---

**Description**

Searches available ISTAT datasets by keyword in their description. The search is case-insensitive.

**Usage**

```
search_dataset(keyword)
```

**Arguments**

keyword            Character string to search for in dataset descriptions

**Value**

A tibble with matching datasets (same columns as `all_available()`)

**Examples**

```
## Not run:
# Search for datasets related to imports
import_datasets <- search_dataset("import")

# Search for population datasets
pop_datasets <- search_dataset("population")

## End(Not run)
```

---

set_filters	<i>Set filters for a dataset</i>
-------------	----------------------------------

---

**Description**

Sets dimension filters that will be used when retrieving data. Filter names should match dimension IDs (case-insensitive).

**Usage**

```
set_filters(dataset, ...)
```

**Arguments**

dataset	An <code>istat_dataset</code> object
...	Named arguments where names are dimension IDs and values are either single values or character vectors for multiple values. Use "." to select all values for a dimension.

**Value**

The modified `istat_dataset` object

**Examples**

```
## Not run:  
ds <- istat_dataset("139_176")  
  
# Set single values  
ds <- set_filters(ds, FREQ = "M", PAESE_PARTNER = "WORLD")  
  
# Set multiple values  
ds <- set_filters(ds, TIPO_DATO = c("ISAV", "ESAV"))  
  
## End(Not run)
```

---

utils	<i>ISTAT API Utility Functions</i>
-------	------------------------------------

---

**Description**

XML parsing and helper functions

# Index

`all_available`, [2](#)

`base`, [3](#)

`dimensions_info`, [3](#)

`discovery`, [3](#)

`get_available_values`, [4](#)

`get_data`, [4](#)

`get_dimension_values`, [5](#)

`istat_dataset`, [6](#)

`istat_get`, [7](#)

`istat_timeout`, [8](#)

`print.istat_dataset`, [9](#)

`reset_filters`, [9](#)

`retrieval`, [10](#)

`search_dataset`, [10](#)

`set_filters`, [11](#)

`utils`, [11](#)