

# Package: epiCo (via r-universe)

February 7, 2025

**Title** Statistical and Viz Tools for Vector-Borne Diseases in Colombia

**Version** 1.0.1

**Description** Provides statistical and visualization tools for the analysis of demographic indicators, and spatio-temporal behavior and characterization of outbreaks of vector-borne diseases (VBDs) in Colombia. It implements travel times estimated in Bravo-Vega C., Santos-Vega M., & Cordovez J.M. (2022), and the endemic channel method (Bortman, M. (1999) <<https://iris.paho.org/handle/10665.2/8562>>).

**License** MIT + file LICENSE

**URL** <https://epiverse-trace.github.io/epiCo/>,  
<https://github.com/epiverse-trace/epiCo>

**BugReports** <https://github.com/epiverse-trace/epiCo/issues>

**Depends** R (>= 4.0.0)

**Imports** dplyr, ggplot2, ggraph, grDevices, igraph, incidence, leaflet, lubridate, magrittr, rlang, scales, spdep, stats, treemapify, utils

**Suggests** checkmate, covr, knitr, rmarkdown, spelling, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/Needs/website** epiverse-trace/epiversetheme

**Config/testthat/edition** 3

**Config/potools/style** explicit

**Encoding** UTF-8

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Language** en-US

**Config/pak/sysreqs** libfontconfig1-dev libfreetype6-dev libgdal-dev  
 gdal-bin libgeos-dev libglpk-dev make libicu-dev libjpeg-dev  
 libpng-dev libxml2-dev libssl-dev libproj-dev libsqlite3-dev  
 libudunits2-dev

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

**RemoteUrl** <https://github.com/epiverse-trace/epiCo>

**RemoteRef** HEAD

**RemoteSha** fec74a100872cb3a449c86ee89e3c2e9f947ee81

## Contents

age_risk . . . . .	2
describe_ethnicity . . . . .	3
describe_occupation . . . . .	4
divipola_table . . . . .	4
endemic_channel . . . . .	5
epi_calendar . . . . .	6
epi_data . . . . .	7
geometric_mean . . . . .	7
geometric_sd . . . . .	8
incidence_rate . . . . .	9
isco88_table . . . . .	10
morans_index . . . . .	10
neighborhoods . . . . .	11
population_pyramid . . . . .	12
<b>Index</b>	<b>13</b>

---

age_risk	<i>Returns the specific rates associated with being infected given age and sex</i>
----------	--

---

## Description

Function that returns the specific rates of being infected given age and sex

## Usage

```
age_risk(age, population_pyramid, sex = NULL, plot = FALSE)
```

**Arguments**

age	A vector with the ages of cases in years from 0 to 100 years
population_pyramid	A dataframe with the count of individuals with the columns age, population and sex
sex	A vector with the sex of cases 'F' and 'M'. The default value is NULL
plot	A boolean for displaying a plot. The default value is FALSE

**Value**

A dataframe with the proportion or total count of individuals

**Examples**

```
pop_pyramid <- population_pyramid("15001", 2015,
  sex = TRUE, total = TRUE,
  plot = FALSE
)
ages <- round(runif(150, 0, 100))
sex <- c(rep("M", 70), rep("F", 80))
age_risk(
  age = ages, sex = sex, population_pyramid = pop_pyramid,
  plot = TRUE
)
```

---

describe\_ethnicity      *Provides the sociological description of ethnicities in Colombia*

---

**Description**

Function that returns the description of the consulted ethnicities

**Usage**

```
describe_ethnicity(ethnic_codes)
```

**Arguments**

ethnic_codes	A numeric vector with the codes of ethnicities to consult
--------------	---

**Value**

A printed message with ethnicities descriptions

**Examples**

```
describe_ethnicity(round(runif(n = 150, min = 1, max = 4)))
```

---

describe\_occupation    *Get ISCO-88 occupation labels from codes*

---

### Description

Function that translates a vector of ISCO-88 occupation codes into a vector of labels

### Usage

```
describe_occupation(isco_codes, sex = NULL, plot = NULL)
```

### Arguments

isco_codes	A numeric vector of ISCO-88 occupation codes (major, submajor, minor, or unit)
sex	A vector with the respective sex for isco_codes vector. The default value is NULL
plot	A type of plot between treemap and circular packing. The default value is NULL

### Value

A string vector of ISCO-88 labels

### Examples

```
demog_data <- data.frame(
  occupation_label =
    c(6111, 3221, 5113, 5133, 6111, 23, 25),
  sex = c("F", "M", "F", "F", "M", "M", "F")
)
describe_occupation(
  isco_codes = demog_data$occupation_label,
  sex = demog_data$sex, plot = "treemap"
)
```

---

divipola\_table    *divipola\_table*

---

### Description

Political and administrative distribution of Colombia's municipalities

### Usage

```
data(divipola_table)
```

**Format**

An object of class `data.frame` with 1121 rows and 8 columns.

**Details**

DIVIPOLA table

---

endemic_channel	<i>Create and return the endemic channel of a disease from an incidence object</i>
-----------------	--

---

**Description**

Function that builds the endemic channel of a disease time series based on the selected method and windows of observation

**Usage**

```
endemic_channel(
  incidence_historic,
  observations = NULL,
  method = c("geometric", "median", "mean", "unusual_behavior"),
  geometric_method = "shifted",
  outlier_years = NULL,
  outliers_handling = c("ignored", "included", "replaced_by_median", "replaced_by_mean",
    "replaced_by_geometric_mean"),
  ci = 0.95,
  plot = FALSE
)
```

**Arguments**

incidence_historic	An incidence object with the historic weekly observations
observations	A numeric vector with the current observations
method	A string with the mean calculation method of preference (median, mean, or geometric) or to use the unusual behavior method (Poisson Distribution Test for Hypoendemic settings)
geometric_method	A string with the selected method for geometric mean calculation; see: <code>geometric_mean</code>
outlier_years	A numeric vector with the outlier years
outliers_handling	A string with the handling decision regarding outlier years, see: <code>outliers_handling</code> function

`ci` = 0.95 A numeric value to specify the confidence interval to use with the geometric method

`plot` A boolean for displaying a plot

**Value**

A dataframe with the observation, historical mean, and confidence intervals (or risk areas)

**Examples**

```
data_event <- epiCo::epi_data
data_ibague <- data_event[data_event$cod_mun_o == 73001, ]
incidence_historic <- incidence::incidence(data_ibague$fec_not,
  interval = "1 epiweek"
)
endemic_channel(incidence_historic,
  method = "geometric", plot = TRUE
)
```

---

epi\_calendar

*Get the epidemiological calendar of a consulted year.*

---

**Description**

Function that returns the starting date of the epidemiological weeks in a year of interest.

**Usage**

```
epi_calendar(year, jan_days = 4)
```

**Arguments**

`year` A numeric value for the year of interest.

`jan_days` Number of January days that the first epidemiological week must contains.

**Value**

A character array with the starting dates of the epidemiological weeks of the given year.

**Examples**

```
epi_calendar(2016)
```

---

epi_data	<i>epi_data</i>
----------	-----------------

---

**Description**

Epidemiological data for the Tolima department for the years 2012 to 2022

**Usage**

```
data(epi_data)
```

**Format**

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 66747 rows and 16 columns.

**Details**

Epidemiological data

---

geometric_mean	<i>Returns the geometric mean of a vector of real numbers.</i>
----------------	--

---

**Description**

Function that returns the geometric mean of a vector of real numbers according to the selected method.

**Usage**

```
geometric_mean(
  x,
  method = c("positive", "shifted", "optimized", "weighted"),
  shift = 1,
  epsilon = 0.001
)
```

**Arguments**

x	A numeric vector of real values
method	Description of methods: <ul style="list-style-type: none"> <li>• positive = only positive values within x are used in the calculation.</li> <li>• shifted = positive and zero values within x are used by adding a shift value before the calculation and subtracting it to the final result.</li> </ul>

- optimized = optimized shifted method. See: De La Cruz, R., & Kreft, J. U. (2018). Geometric mean extension for data sets with zeros. arXiv preprint arXiv:1806.06403.
- weighted = a probability weighted calculation of gm for negative, positive, and zero values. See: Habib, E. A. (2012). Geometric mean for negative and zero values. International Journal of Research and Reviews in Applied Sciences, 11(3), 419-432.

shift = 1 (default) a positive value to use in the shifted method  
 epsilon = 1e-5 (default) the minimum positive value to consider in the optimized method.

### Value

The geometric mean of the x vector, and the epsilon value if optimized method is used.

### Examples

```
x <- c(4, 5, 3, 7, 8)
geometric_mean(x, method = "optimized")
```

---

geometric\_sd                      *Returns the geometric standard deviation of a vector of real numbers.*

---

### Description

Function that returns the geometric standard deviation of a vector of real numbers according to the selected method.

### Usage

```
geometric_sd(
  x,
  method = c("positive", "shifted", "optimized", "weighted"),
  shift = 1,
  delta = 0.001
)
```

### Arguments

x	A numeric vector of real values
method	Description of methods: <ul style="list-style-type: none"> <li>• positive = only positive values within x are used in the calculation.</li> <li>• shifted = positive and zero values within x are used by adding a shift value before the calculation and subtracting it to the final result.</li> <li>• optimized = optimized shifted method. See: De La Cruz, R., &amp; Kreft, J. U. (2018). Geometric mean extension for data sets with zeros. arXiv preprint arXiv:1806.06403.</li> </ul>



- weighted = a probability weighted calculation of gm for negative, positive, and zero values. See: Habib, E. A. (2012). Geometric mean for negative and zero values. International Journal of Research and Reviews in Applied Sciences, 11(3), 419-432.

shift            a positive value to use in the shifted method  
 delta            an positive value (shift) used in the optimized method.

### Value

The geometric mean of the x vector, and the epsilon value if optimized method is used.

### Examples

```
x <- c(4, 5, 3, 7, 8)
geometric_sd(x, method = "optimized")
```

---

incidence\_rate            *Extends an incidence class object with incidence rates estimations.*

---

### Description

Function that estimates incidence rates from a incidence class object and population projections.

### Usage

```
incidence_rate(incidence_object, level, scale = 1e+05)
```

### Arguments

incidence\_object            An incidence object.  
 level            Administration level at which incidence counts are grouped (0 = national, 1 = state/department, 2 = city/municipality).  
 scale            Scale to consider when calculating the incidence\_rate.

### Value

A modified incidence object where counts are normalized with the population.

**Examples**

```

data_event <- epiCo::epi_data
incidence_historic <- incidence::incidence(data_event$fec_not,
  groups = data_event$cod_mun_o,
  interval = "1 year"
)
incidence_object <- subset(incidence_historic,
  from = "2015-01-04",
  to = "2018-12-27"
)
inc_rate <- incidence_rate(incidence_object, level = 2, scale = 100000)
inc_rate$rates

```

---

isco88_table	<i>isco88_table</i>
--------------	---------------------

---

**Description**

ISCO88 description of occupations

**Usage**

```
data(isco88_table)
```

**Format**

An object of class `data.frame` with 390 rows and 8 columns.

**Details**

ISCO88 occupation table

---

morans_index	<i>Calculate spatial correlation of given municipalities in an incidence_rate object.</i>
--------------	---

---

**Description**

Function to calculate spatial autocorrelation via Moran's Index from a given `incidence_rate` object grouped by municipality.

**Usage**

```
morans_index(incidence_object, scale = 1e+05, threshold = 2, plot = TRUE)
```

**Arguments**

incidence_object	An incidence object with one observation for the different locations (groups).
scale	Scale to consider when calculating the incidence_rate.
threshold	Maximum traveling time around each municipality.
plot	if TRUE, returns a plot of influential observations in the Moran's plot.

**Value**

List of Moran's I clustering analysis, giving the quadrant of each observation, influential values.

**Examples**

```
data_event <- epiCo::epi_data
incidence_historic <- incidence::incidence(data_event$fec_not,
  groups = data_event$cod_mun_o,
  interval = "4 year"
)
incidence_object <- subset(incidence_historic,
  from = "2015-01-04",
  to = "2018-12-27"
)
morans_index(incidence_object, scale = 100000, threshold = 2, plot = TRUE)
```

---

neighborhoods

*Neighborhoods from real travel distances in Colombia*


---

**Description**

Function to build neighborhoods from real travel distances inside Colombia by land or river transportation.

**Usage**

```
neighborhoods(query_vector, threshold = 2)
```

**Arguments**

query_vector	Codes of the municipalities to consider for the neighborhoods.
threshold	Maximum traveling time around each municipality.

**Value**

neighborhood object according to the introduced threshold.

### Examples

```
query_vector <- c("05001", "05002", "05004", "05021", "05030", "05615")
neighborhoods(query_vector, 2)
```

---

population\_pyramid      *Returns the population pyramid of the consulted region*

---

### Description

Function that returns the population pyramid of the municipality or department of a specific year

### Usage

```
population_pyramid(  
  divipola_code,  
  year,  
  sex = TRUE,  
  range = 5,  
  total = TRUE,  
  plot = FALSE  
)
```

### Arguments

divipola_code	A code from the divipola table representing a department or municipality. To obtain values at the national level, code '0' is used
year	A numeric input for the year of interest
sex	A boolean to consult data disaggregated by sex. The default value is TRUE
range	A numeric value from 1 to 100 for the age range to use. The default value is 5
total	A boolean for returning the total number rather than the proportion of the country's population. The default value is TRUE
plot	A boolean for displaying a plot. The default value is TRUE

### Value

A dataframe with the proportion or total count of individuals

### Examples

```
population_pyramid("15001", 2015, sex = TRUE, total = TRUE, plot = TRUE)
```

# Index

## \* datasets

divipola\_table, [4](#)

epi\_data, [7](#)

isco88\_table, [10](#)

age\_risk, [2](#)

describe\_ethnicity, [3](#)

describe\_occupation, [4](#)

divipola\_table, [4](#)

endemic\_channel, [5](#)

epi\_calendar, [6](#)

epi\_data, [7](#)

geometric\_mean, [7](#)

geometric\_sd, [8](#)

incidence\_rate, [9](#)

isco88\_table, [10](#)

morans\_index, [10](#)

neighborhoods, [11](#)

population\_pyramid, [12](#)