

# Package ‘mediterraneancalculations’

February 16, 2023

**Type** Package

**Version** 0.2-2

**Date** 2023-01-10

**Title** Mediterranean Calculations

**Author** Sergio M. Vicente-Serrano [aut], Fergus Reig-Gracia [aut, cre]

**Maintainer** Fergus Reig-Gracia <fergusrg@gmail.com>

**Depends** R (>= 2.10)

**Imports** chron, sf, SpatialTools, lmom, SPEI, RobustLinearReg, hydroGOF, snowfall, Kendall, stats

**Description** .

**License** GPL (>= 3)

**URL** <https://lcsc.csic.es>

**LazyLoad** no

**Encoding** UTF-8

**Suggests** MASS,  
rmarkdown,  
knitr,  
testthat (>= 3.0.0)

**NeedsCompilation** no

**RoxygenNote** 7.2.3

**VignetteBuilder** knitr

**Collate** 'functions-mediterranean-calculations.r'

'mediterranean-calculations.r'

'mediterraneancalculations.R'

'mediterraneancalculationsNews.R'

## R topics documented:

mediterraneancalculations-package . . . . .	3
alexanderson_homogenize . . . . .	3
alexanderson_homogenize_data . . . . .	4

apply_ecdf . . . . .	5
apply_ecdf_month . . . . .	5
calculate_reconstruction_statistics . . . . .	6
calculate_statistics . . . . .	6
calculate_statistics_data . . . . .	7
calc_data_year . . . . .	7
calc_data_year_month_station . . . . .	8
calc_mkTrend_pval . . . . .	8
calc_mkTrend_slp . . . . .	9
calc_percentage . . . . .	9
ClimIndNews . . . . .	10
coef_var . . . . .	10
delete_zero . . . . .	11
delete_zones . . . . .	11
dry_spell_trend . . . . .	12
fill_one_series . . . . .	12
fill_series . . . . .	13
fill_unfillable_station . . . . .	13
main_mediterranean_calculations . . . . .	14
main_mediterranean_calculations_ . . . . .	15
mediterranean_calculations . . . . .	15
mkTrend . . . . .	16
mobile_trends . . . . .	16
near_correlations . . . . .	17
near_estations . . . . .	17
order_data . . . . .	18
overlap_station . . . . .	18
overlap_station_no_0 . . . . .	19
percentage_of_zeros . . . . .	19
quality_control . . . . .	20
read_data . . . . .	20
read_years . . . . .	21
save_csvs . . . . .	21
save_data . . . . .	22
save_delete_data . . . . .	22
second_data_fill . . . . .	23
second_data_fill_data . . . . .	23
select_data . . . . .	24
sum_no_nas . . . . .	24

---

`mediterraneancalculations-package`*mediterraneancalculations: Mediterranean Calculations*

---

## Description

.

## Details

Info

## See Also

Useful links:

- <https://lcsc.csic.es>

---

`alexanderson_homogenize`*Test de Alexanderson para todos los ficheros disponibles (que han pasado el segundo relleno con éxito)*

---

## Description

Test de Alexanderson para todos los ficheros disponibles (que han pasado el segundo relleno con éxito)

## Usage

```
alexanderson_homogenize(data, folder)
```

## Arguments

<code>data</code>	data y coor
<code>folder</code>	directorio para guardar los datos de salida

## Value

data y coor

---

alexanderson\_homogenize\_data

*Homogeneizar - test de Alexanderson Lo usamos en code\_web\_maps/snht\_functions.R Existe también librería snht de R – Serie de referencia, compara y corrige Para cada base de datos (1870, 1900...) Elegimos las 5 series más correlacionadas usando las serie de diferencias Con las 5 hacemos una media ponderada,  $(\text{correlación} * \text{dato1} + \dots) / \text{sum}(\text{correlaciones})$  y será la serie de referencia Alexanderson nos dará un punto de ruptura y un valor ratio por el que multiplicar la parte antigua... iterar mientras de puntos de ruptura Guardar estadísticos de inhomogeneidades. Básicamente número de datos cambiados en cada series y momento de la inhomogeneidad. CSV con número de datos cambiados y CSV con punto de inhomogeneidad - todo x 12 meses*

---

## Description

Homogeneizar - test de Alexanderson Lo usamos en code\_web\_maps/snht\_functions.R Existe también librería snht de R – Serie de referencia, compara y corrige Para cada base de datos (1870, 1900...) Elegimos las 5 series más correlacionadas usando las serie de diferencias Con las 5 hacemos una media ponderada,  $(\text{correlación} * \text{dato1} + \dots) / \text{sum}(\text{correlaciones})$  y será la serie de referencia Alexanderson nos dará un punto de ruptura y un valor ratio por el que multiplicar la parte antigua... iterar mientras de puntos de ruptura Guardar estadísticos de inhomogeneidades. Básicamente número de datos cambiados en cada series y momento de la inhomogeneidad. CSV con número de datos cambiados y CSV con punto de inhomogeneidad - todo x 12 meses

## Usage

```
alexanderson_homogenize_data(file_data, no_use_series = c())
```

## Arguments

file_data	ruta del fichero de datos
no_use_series	series que no se homogeneizarán

## Value

None

---

apply_ecdf	<i>Anomalías de los datos mensuales</i>
------------	---

---

### Description

Anomalías de los datos mensuales

### Usage

```
apply_ecdf(data)
```

### Arguments

data	datos mensuales
------	-----------------

### Value

anomalías de los datos

---

apply_ecdf_month	<i>Anomalías de los datos</i>
------------------	-------------------------------

---

### Description

Anomalías de los datos

### Usage

```
apply_ecdf_month(data)
```

### Arguments

data	datos mensuales
------	-----------------

### Value

anomalías de los datos

---

calculate_reconstruction_statistics	<i>Calcular estadísticos de la reconstrucción - hydroGOF – estadístico por estación - D / MAE / PBIAS / RMSE - por estación y mes</i>
-------------------------------------	---

---

**Description**

Calcular estadísticos de la reconstrucción - hydroGOF – estadístico por estación - D / MAE / PBIAS / RMSE - por estación y mes

**Usage**

calculate\_reconstruction\_statistics(sim, obs)

**Arguments**

sim	datos rellenados
obs	datos iniciales

**Value**

datos eliminados y datos de entrada por estación

---

calculate_statistics	<i>Salida final con todos los estadísticos, serie regional promedio, tendencias, SPI...</i>
----------------------	---

---

**Description**

Salida final con todos los estadísticos, serie regional promedio, tendencias, SPI...

**Usage**

calculate\_statistics(data, data\_ori)

**Arguments**

data	data y coor
data_ori	data original

**Value**

data and coor

---

`calculate_statistics_data`

*Calcula estadísticos de los datos Tendencia mensual, estacional y anual, paquete Trend, función sens.slope Sumar 1 a todo para evitar 0s Significación, paquete modifiedmk, función bbsmk Serie promedio de todo el país SPI a escalas 3, 12, y 24 de cada serie, importante que sea imposible invertir las operaciones Código Sergio para generar arrays y hacer figuras de tendencia*

---

### Description

Calcula estadísticos de los datos Tendencia mensual, estacional y anual, paquete Trend, función sens.slope Sumar 1 a todo para evitar 0s Significación, paquete modifiedmk, función bbsmk Serie promedio de todo el país SPI a escalas 3, 12, y 24 de cada serie, importante que sea imposible invertir las operaciones Código Sergio para generar arrays y hacer figuras de tendencia

### Usage

```
calculate_statistics_data(file_data, data_ori)
```

### Arguments

file_data	datos y coordenadas
data_ori	datos originales

### Value

None

---

calc_data_year	<i>Suma los datos de cada año, para devolver un solo dato anual</i>
----------------	---

---

### Description

Suma los datos de cada año, para devolver un solo dato anual

### Usage

```
calc_data_year(data)
```

### Arguments

data	matriz de datos
------	-----------------

### Value

un dato por año

---

`calc_data_year_month_station`*Devuelve el slope z por años y estaciones*

---

**Description**

Devuelve el slope z por años y estaciones

**Usage**

```
calc_data_year_month_station(data, calc_function)
```

**Arguments**

<code>data</code>	datos de las estaciones
<code>calc_function</code>	función a utilizar

**Value**

lista de resultados

---

`calc_mkTrend_pval`*Devuelve el pval calculado por mkTrend o el pval0 si el pval era NA*

---

**Description**

Devuelve el pval calculado por mkTrend o el pval0 si el pval era NA

**Usage**

```
calc_mkTrend_pval(data)
```

**Arguments**

<code>data</code>	matriz de datos
-------------------	-----------------

**Value**

pval



---

calc_mkTrend_slp	<i>regresión lineal de los datos contra los años</i>
------------------	--

---

**Description**

regresión lineal de los datos contra los años

**Usage**

```
calc_mkTrend_slp(data)
```

**Arguments**

data	index
------	-------

**Value**

lm

---

calc_percentage	<i>Diferencia en porcentaje</i>
-----------------	---------------------------------

---

**Description**

Diferencia en porcentaje

**Usage**

```
calc_percentage(datos, years = NA)
```

**Arguments**

datos	datos
years	años

**Value**

percentage

---

ClimIndNews	<i>mediterraneancalculationsNews</i>
-------------	--------------------------------------

---

### Description

Show the NEWS file of the **mediterraneancalculations** package.

### Usage

```
ClimIndNews()
```

### Details

(See description)

---

coef_var	<i>Coeficientes de variación, desviación estándar</i> <i><a href="https://fhernanb.github.io/Manual-de-R/varia.html">https://fhernanb.github.io/Manual-de-R/varia.html</a></i>
----------	---

---

### Description

Coeficientes de variación, desviación estándar <https://fhernanb.github.io/Manual-de-R/varia.html>

### Usage

```
coef_var(x, na.rm = FALSE)
```

### Arguments

x	datos
na.rm	Ignorara NAs

### Value

percentage

---

delete_zero	<i>Elimina datos si tenemos 5 meses o más seguidos de 0s, si uno de los meses implicados tiene menos del 70 por ciento de ceros</i>
-------------	---

---

**Description**

Elimina datos si tenemos 5 meses o más seguidos de 0s, si uno de los meses implicados tiene menos del 70 por ciento de ceros

**Usage**

```
delete_zero(data)
```

**Arguments**

data	datos
------	-------

**Value**

datos con los grupos de 0s eliminados

---

delete_zones	<i>Por debajo de 28 grados norte, eliminar estaciones</i>
--------------	---

---

**Description**

Por debajo de 28 grados norte, eliminar estaciones

**Usage**

```
delete_zones(data)
```

**Arguments**

data	data y coor
------	-------------

**Value**

data and coor

---

dry_spell_trend	<i>esta función calcula la tendencia. Hay que definirle un objeto de años (years) con el año correspondiente a cada caso.</i>
-----------------	---

---

**Description**

esta función calcula la tendencia. Hay que definirle un objeto de años (years) con el año correspondiente a cada caso.

**Usage**

```
dry_spell_trend(index, threshold)
```

**Arguments**

index	index
threshold	threshold

**Value**

output

---

fill_one_series	<i>Rellena la serie recibida utilizando las otras en el orden en el que están en other_series</i>
-----------------	---

---

**Description**

Rellena la serie recibida utilizando las otras en el orden en el que están en other\_series

**Usage**

```
fill_one_series(series, other_series)
```

**Arguments**

series	serie de datos a completar
other_series	series de datos con las que completar en el orden en el que se tienen que utilizar

**Value**

serie de datosd rellena

---

fill_series	<i>Rellenado mensual de las series Usamos estaciones a menos de 200km con correlación por encima de 0.7 Para junio, julio y agosto, rellenamos con la más cercana Utilizar el método que mejor correlaciona con la serie original</i>
-------------	---

---

### Description

Rellenado mensual de las series Usamos estaciones a menos de 200km con correlación por encima de 0.7 Para junio, julio y agosto, rellenamos con la más cercana Utilizar el método que mejor correlaciona con la serie original

### Usage

```
fill_series(control_data, min_correlation, max_dist)
```

### Arguments

control_data	datos de las estaciones y sus coordenadas
min_correlation	Correlación mínima para usar el dato en el relleno
max_dist	distancia máxima entre las series a utilizar

### Value

data y coord con los datos que no pasan el control eliminados

---

fill\_unfillable\_station

*En los países que no salgan series, vamos a permitir que hasta tres años de datos se rellenen con la media. Es decir, pongamos que si para un periodo concreto 1900-2020 no salen series pero saldrían porque hay un máximo de tres años de datos (es decir 36 meses), rellenamos esos datos con el promedio de los 15 datos más cercanos en el tiempo. Por ejemplo, si es 1900, pues con la media de 1900-1915, si es 1915, pues con la media de 1907 a 1922. Siempre y cuando esos cinco años no estén entre 2015 y 2020 o en los cinco primeros años de las series, que entonces tiramos la serie pues podría afectar a las tendencias. Si las series son las de 1981-2020, lo mismo, pero dejamos solamente dos años de datos perdidos.*

---

**Description**

En los países que no salgan series, vamos a permitir que hasta tres años de datos se rellenen con la media. Es decir, pongamos que si para un periodo concreto 1900-2020 no salen series pero saldrían porque hay un máximo de tres años de datos (es decir 36 meses), rellenos esos datos con el promedio de los 15 datos más cercanos en el tiempo. Por ejemplo, si es 1900, pues con la media de 1900-1915, si es 1915, pues con la media de 1907 a 1922. Siempre y cuando esos cinco años no estén entre 2015 y 2020 o en los cinco primeros años de las series, que entonces tiramos la serie pues podría afectar a las tendencias. Si las series son las de 1981-2020, lo mismo, pero dejamos solamente dos años de datos perdidos.

**Usage**

```
fill_unfillable_station(data, fillable_years)
```

**Arguments**

data                    datos de las estaciones que se intentarán rellenar  
 fillable\_years   años rellenables con la media mensual de la propia estación

**Value**

None

---

main\_mediterranean\_calculations

*Lee los ficheros de precipitación, calcula estadísticos y guarda los resultados Los ficheros de entrada son 2 CSVs uno de coordenadas en grados (filas las estaciones y columnas lat y lon y otro de datos mensuales con fechas en filas y las estaciones en las columnas)*

---

**Description**

Lee los ficheros de precipitación, calcula estadísticos y guarda los resultados Los ficheros de entrada son 2 CSVs uno de coordenadas en grados (filas las estaciones y columnas lat y lon y otro de datos mensuales con fechas en filas y las estaciones en las columnas)

**Usage**

```
main_mediterranean_calculations(file_data, file_coor)
```

**Arguments**

file\_data            ruta del fichero de datos  
 file\_coor            ruta del fichero de coordenadas

**Value**

None

---

`main_mediterranean_calculations_`*Calcula los estadísticos para un país*

---

**Description**

Calcula los estadísticos para un país

**Usage**

```
main_mediterranean_calculations_(read_all_data, folder, pb = NULL)
```

**Arguments**

<code>read_all_data</code>	datos de entrada
<code>folder</code>	carpeta donde guarda ficheros
<code>pb</code>	barra de progreso

**Value**

None

---

`mediterranean_calculations`*Hace el control de calidad Control de calidad: Se estaciones con menos de 20 años de datos y usando las 10 más correlacionadas a menos de 200 km, se desechan los datos con un percentil de diferencia de más de 0.6.*

---

**Description**

Hace el control de calidad Control de calidad: Se estaciones con menos de 20 años de datos y usando las 10 más correlacionadas a menos de 200 km, se desechan los datos con un percentil de diferencia de más de 0.6.

**Usage**

```
mediterranean_calculations(data, max_dist_eval)
```

**Arguments**

<code>data</code>	ruta del fichero de datos
<code>max_dist_eval</code>	máxima distancia entre 2 estaciones para usar una para evaluar una con la otra

**Value**

data y coor con los datos que no pasan el control eliminados

---

mkTrend	<i>Calcula pval (a veces no da resultado por temas de iteración) entonces coger pval0.</i>
---------	--

---

**Description**

Calcula pval (a veces no da resultado por temas de iteración) entonces coger pval0.

**Usage**

```
mkTrend(x, ci = 0.95)
```

**Arguments**

x	x
ci	ci

**Value**

list

---

mobile_trends	<i>esto te calcula unas tendencias móviles de una serie, en este caso que empieza en 1851 y termina en 2018,. Habría que hacerlo para cada base de datos y estación.</i>
---------------	--

---

**Description**

esto te calcula unas tendencias móviles de una serie, en este caso que empieza en 1851 y termina en 2018,. Habría que hacerlo para cada base de datos y estación.

**Usage**

```
mobile_trends(datos)
```

**Arguments**

datos	datos
-------	-------

**Value**

list



---

near_correlations	<i>Devuelve la correlación entre las estaciones Sin tener en cuenta las que están a más de 200 km (NA en esos casos)</i>
-------------------	--

---

**Description**

Devuelve la correlación entre las estaciones Sin tener en cuenta las que están a más de 200 km (NA en esos casos)

**Usage**

```
near_correlations(data, coor, max_dist)
```

**Arguments**

data	datos mensuales
coor	coordenadas de las estaciones que corresponden con data
max_dist	distancia máxima entre las series a utilizar

**Value**

correlación entre las estaciones

---

near_estations	<i>Devuelve las estaciones por orden de cercanía Sin tener en cuenta las que están a más de 200 km (NA en esos casos)</i>
----------------	---

---

**Description**

Devuelve las estaciones por orden de cercanía Sin tener en cuenta las que están a más de 200 km (NA en esos casos)

**Usage**

```
near_estations(data, coor, max_dist)
```

**Arguments**

data	datos mensuales
coor	coordenadas de las estaciones que corresponden con data
max_dist	distancia máxima entre las series a utilizar

**Value**

correlación entre las estaciones

---

order_data	<i>Ordena los datos y devuelve una lista con el orden</i>
------------	---

---

**Description**

Ordena los datos y devuelve una lista con el orden

**Usage**

order\_data(data)

**Arguments**

data                    datos

**Value**

list

---

overlap_station	<i>Calcula el tiempo de solape existente entre cada par de series</i>
-----------------	---

---

**Description**

Calcula el tiempo de solape existente entre cada par de series

**Usage**

overlap\_station(control\_data)

**Arguments**

control\_data      datos de las estaciones y sus coordenadas

**Value**

matriz con los meses que se solapan las estaciones entre si

---

overlap_station_no_0	<i>Calcula el tiempo de solape existente entre cada par de series sin contar 0s</i>
----------------------	---

---

**Description**

Calcula el tiempo de solape existente entre cada par de series sin contar 0s

**Usage**

```
overlap_station_no_0(control_data)
```

**Arguments**

control\_data      datos de las estaciones y sus coordenadas

**Value**

matriz con los meses que se solapan las estaciones entre si

---

percentage_of_zeros	<i>Devuelve el porcentaje de datos válidos que son 0s</i>
---------------------	---

---

**Description**

Devuelve el porcentaje de datos válidos que son 0s

**Usage**

```
percentage_of_zeros(data)
```

**Arguments**

data                  datos

**Value**

percentage

---

quality_control	<i>Control de calidad Estaciones con menos de 20 años de datos retirar Usando las 10 más cercanas a menos de 200 km, desechar si promedio de percentil se diferencia en más de 0.6 o en más de 0.5 para datos 0</i>
-----------------	---

---

### Description

Control de calidad Estaciones con menos de 20 años de datos retirar Usando las 10 más cercanas a menos de 200 km, desechar si promedio de percentil se diferencia en más de 0.6 o en más de 0.5 para datos 0

### Usage

```
quality_control(data, coor, max_dist, max_diff_anomaly, max_diff_anomaly_0)
```

### Arguments

data	datos
coor	coordenadas
max_dist	máxima distancia entre 2 estaciones para usar una para evaluar o completar la otra
max_diff_anomaly	máxima diferencia de anomalías para mantener dato en el control
max_diff_anomaly_0	máxima diferencia de anomalías para mantener dato en el control, si el dato es 0

### Value

data y coor con los datos que no pasan el control eliminados

---

read_data	<i>Leemos los datos desde los CSVs con el formato acordado Los ficheros de entrada son 2 CSVs uno de coordenadas en grados (filas las estaciones y columnas lat y lon y otro de datos mensuales con fechas en filas y las estaciones en las columnas)</i>
-----------	---

---

### Description

Leemos los datos desde los CSVs con el formato acordado Los ficheros de entrada son 2 CSVs uno de coordenadas en grados (filas las estaciones y columnas lat y lon y otro de datos mensuales con fechas en filas y las estaciones en las columnas)

### Usage

```
read_data(file_data, file_coor)
```

**Arguments**

file_data	ruta del fichero de datos
file_coor	ruta del fichero de coordenadas

**Value**

datos originales, datos de interes y coordenadas de las estaciones leidas

---

read_years	<i>Lee los años de cadenas de texto que terminan con los años</i>
------------	---

---

**Description**

Lee los años de cadenas de texto que terminan con los años

**Usage**

```
read_years(txt)
```

**Arguments**

txt	texto o vector de textos
-----	--------------------------

**Value**

list

---

save_csvs	<i>Guardamos los datos en CSVs</i>
-----------	------------------------------------

---

**Description**

Guardamos los datos en CSVs

**Usage**

```
save_csvs(i_ini, folder_name, data_save, coor_save)
```

**Arguments**

i_ini	identificador de los ficheros
folder_name	carpeta en la que guardar el fichero
data_save	datos de las estaciones a guardar
coor_save	datos de coordenadas a guardar

**Value**

None

---

save_data	<i>Guarda la salida en 5 fichero con los datos 5 ficheros que indican si cada datos es original o rellenado (1 dato no alterado, 0 dato alterado) y 5 ficheros de coordenadas para las estaciones de cada fichero de datos, que son: - 1870 a 2020 con más de 80 años originales - 1900 a 2020 con más de 80 años originales - 1930 a 2020 con más de 60 años originales - 1950 a 2020 con más de 40 años originales - 1990 a 2020 con más de 30 años originales</i>
-----------	--

---

### Description

Guarda la salida en 5 fichero con los datos 5 ficheros que indican si cada datos es original o rellenado (1 dato no alterado, 0 dato alterado) y 5 ficheros de coordenadas para las estaciones de cada fichero de datos, que son: - 1870 a 2020 con más de 80 años originales - 1900 a 2020 con más de 80 años originales - 1930 a 2020 con más de 60 años originales - 1950 a 2020 con más de 40 años originales - 1990 a 2020 con más de 30 años originales

### Usage

```
save_data(data_ori, control_data)
```

### Arguments

data_ori	datos originales leídos de los ficheros CSV
control_data	datos de las estaciones y sus coordenadas

### Value

data y coor con los datos que no pasan el control eliminados

---

save_delete_data	<i>Para cada estación guardar, número de datos de entrada y de datos eliminados</i>
------------------	---

---

### Description

Para cada estación guardar, número de datos de entrada y de datos eliminados

### Usage

```
save_delete_data(ori_data, process_data, folder)
```

### Arguments

ori_data	datos iniciales
process_data	datos procesados
folder	carpeta en la que se guarda el fichero resultante

**Value**

datos eliminados y datos de entrada por estación

---

second_data_fill	<i>Realiza un segundo relleno Para cada base de datos (1870, 1900...) Estaciones con más del 90 o 95 Ordenamos las estaciones por correlación (mínima 0.5) y rellenamos usando los 10 métodos... Las estaciones sin relleno total las tiramos</i>
------------------	---

---

**Description**

Realiza un segundo relleno Para cada base de datos (1870, 1900...) Estaciones con más del 90 o 95 Ordenamos las estaciones por correlación (mínima 0.5) y rellenamos usando los 10 métodos... Las estaciones sin relleno total las tiramos

**Usage**

```
second_data_fill(data, max_dist_eval = NA)
```

**Arguments**

data	data y coor
max_dist_eval	maxima distancia para el relleno

**Value**

data y coor

---

second_data_fill_data	<i>Realiza un segundo relleno Para cada base de datos (1870, 1900...) Estaciones con más del 90 o 95 Ordenamos las estaciones por correlación (mínima 0.5) y rellenamos usando los 10 métodos... Las estaciones sin relleno total las tiramos</i>
-----------------------	---

---

**Description**

Realiza un segundo relleno Para cada base de datos (1870, 1900...) Estaciones con más del 90 o 95 Ordenamos las estaciones por correlación (mínima 0.5) y rellenamos usando los 10 métodos... Las estaciones sin relleno total las tiramos

**Usage**

```
second_data_fill_data(file_data, fillable_years = 36, max_dist = NA)
```

**Arguments**

- file\_data            ruta del fichero de datos
- fillable\_years    años rellenables con la media mensual de la propia estación
- max\_dist           máxima distancia permitida para el relleno

**Value**

None

---

select_data	<i>Primeros datos válidos (no NAs)</i>
-------------	--

---

**Description**

Primeros datos válidos (no NAs)

**Usage**

select\_data(data, n\_reference\_stations)

**Arguments**

- data                datos
- n\_reference\_stations    numero de datos a devolver

**Value**

primeros datos distintos de NA

---

sum_no_nas	<i>Número de datos distintos de NA</i>
------------	--

---

**Description**

Número de datos distintos de NA

**Usage**

sum\_no\_nas(data)

**Arguments**

- data                datos

**Value**

número de datos no NAs



# Index

alexanderson\_homogenize, [3](#)  
alexanderson\_homogenize\_data, [4](#)  
apply\_ecdf, [5](#)  
apply\_ecdf\_month, [5](#)  
  
calc\_data\_year, [7](#)  
calc\_data\_year\_month\_station, [8](#)  
calc\_mkTrend\_pval, [8](#)  
calc\_mkTrend\_slp, [9](#)  
calc\_percentage, [9](#)  
calculate\_reconstruction\_statistics, [6](#)  
calculate\_statistics, [6](#)  
calculate\_statistics\_data, [7](#)  
ClimIndNews, [10](#)  
coef\_var, [10](#)  
  
delete\_zero, [11](#)  
delete\_zones, [11](#)  
dry\_spell\_trend, [12](#)  
  
fill\_one\_series, [12](#)  
fill\_series, [13](#)  
fill\_unfillable\_station, [13](#)  
  
main\_mediterranean\_calculations, [14](#)  
main\_mediterranean\_calculations\_, [15](#)  
mediterranean\_calculations, [15](#)  
mediterranean\_calculations  
    (mediterranean\_calculations-package),  
    [3](#)  
mediterranean\_calculations-package, [3](#)  
mkTrend, [16](#)  
mobile\_trends, [16](#)  
  
near\_correlations, [17](#)  
near\_estations, [17](#)  
  
order\_data, [18](#)  
overlap\_station, [18](#)  
overlap\_station\_no\_0, [19](#)  
  
percentage\_of\_zeros, [19](#)  
  
quality\_control, [20](#)  
  
read\_data, [20](#)  
read\_years, [21](#)  
  
save\_csvs, [21](#)  
save\_data, [22](#)  
save\_delete\_data, [22](#)  
second\_data\_fill, [23](#)  
second\_data\_fill\_data, [23](#)  
select\_data, [24](#)  
sum\_no\_nas, [24](#)