User's guide to **climatol**

An R contributed package for homogenization of climatological series
(and functions for drawing wind-rose and Walter&Lieth diagrams)
Version 2.1, distributed under the GPL license, version 2 or newer

By José A. Guijarro *(http://webs.ono.com/climatol/climatol.html)*
State Meteorological Agency, Balearic Islands Office, Spain

This guide's version: 1.1 (June, 2011)

Foreword

The "Climatol" R contributed package is mostly devoted to the problem of homogenizing climatological series, that is to say, remove the perturbations produced by changes in the conditions of observation or in the nearby environment to allow the series to reflect only (as far as possible) the climatic variations.

The R standard documentation of the package provides descriptions of the functions and their parameters, and users should refer to it whenever needed. This guide, on the other hand, has been written as a complement, trying to focus more on explaining the methodology underlying the algorithms of the package, how to call its functions, and how to interpret and use their results.

This guide is structured in two parts: a *Quick start* (in the following few pages) for those anxious to begin homogenizing their data, and an *Extended guide* where the different aspects of the package are treated more thoroughly.

Most examples of this guide can be reproduced with data files contained in `climatol-dat.zip`, downloadable from *http://webs.ono.com/climatol/climatol-dat.zip*, which contains real series from a Mediterranean area, although the names and coordinates of the stations are fictitious.

Acknowledgements

This package has greatly benefited from fruitful discussions in the frame of COST Action ES0601, entitled *Advances in homogenisation methods of climate series: an integrated approach (HOME)*. My acknowledgments to all the participants, and to the European Science Foundation for promoting and funding this enriching meetings. I must also acknowledge the Spanish State Meteorological Agency *(AEMET)* for its continuous support to my participation in this Action.

Quick start

First we need to prepare the input data in two plain text files with adequate formats. In one of them you must provide the coordinates and names of the stations, containing a line of the form

```
X Y Z CODE NAME
```

for each station, where the coordinates `X` and `Y` may be in km (from e.g., an UTM projection) or in geographical degrees (longitude and latitude, in this order) with their fractional part in decimals (not in the degrees, minutes and seconds form). The other parameters are the altitude `Z` in m, an identification `CODE` of the station, and the `NAME` of the station itself, that must be enclosed in quotes if it contains more than one word. (It is advisable to put all names between quotes to avoid errors). The name of this file must be `VAR_FIRSTY-LASTY.est`, where `VAR` is an abbreviation of the climatic variable being analyzed, and `FIRSTY` and `LASTY` are the first and last years of the studied period.

The data must be arranged in another single file containing station data blocks in the same order as they appear in the station file. The file base name will be that of the station file, using the extension `dat`.

Example: Suppose you are going to homogenize monthly average minimum temperatures from 1956 to 2005, and you choose `Tmin` as a short name for that variable. The stations file would be `Tmin_1956-2005.est`, and could begin, as in the accompanying example data, with:

```
27.0 53.9  456 S03  "La Perla"
31.8 26.5  123 S08  "El Palmeral"
49.2 30.0  154 S11  "Miraflores"
43.4 29.6  156 S13  "Torremar"
... (etc)
```

And the data file should be named `Tmin_1956-2005.dat`, and their first lines could be:

```
NA NA NA NA NA NA NA NA NA NA NA NA
-0.4 1.8 5.5 6.5 15.1 17.4 16.7 16.4 12.2 6.0 2.6 2.3
1.5 4.0 6.5 8.7 12.4 12.1 20.3 NA 14.7 11.0 3.2 0.5
... (etc)
```

This would be the data for the first station of your network[1], in chronological order: January to December of 1956, the same for 1957 in the second line, 1958 in the third line, etc. In this example, data from 1956 and August 1958 are missing, and are replaced by `NA` (Not Available), which is the standard missing data code in R (though others may be used). When all the data from the first station are listed, data from the second station follow, an so on until all station data are reported. It is important to note that all station must report data for every month of the study period (1956-2005 in our example), and hence the need of including missing codes to fill any

[1] Actually, these are not the first lines of our example data, which do not have any missing data in the first three lines. That is why they have been replaced by these other in the text, in order to illustrate how to proceed when missing data are present, which is the usual case.

missing data. For convenience, 12 values (a whole year) have been placed in each line, but this is not compulsory; data may be placed in a free space separated format with any number (even variable) of data items in each line, because they will be read sequentially. (Important note: no month must be simultaneously void of data in all the stations of the file, since this would result in an abnormal process termination).

All you have to do to homogenize your data is to start R in your working directory (where your data and station files are located), load the homogeneity functions, either with the command

```
library(climatol)
```

if you made a regular installation of the package, or with

```
source("depurdat.R")
```

if you have this file[2] in your working directory, and issue the automatic homogenization command, that in our example would be:

```
homogen("Tmin", 1956, 2005)
```

This command accepts other optional parameters, the more important being the following:

nm Number of data per year in each station (12 by default: monthly values. Set to `nm=1` if you are analyzing annual data, `nm=4` for seasonal data, etc).

deg Set to `TRUE` if coordinates are in geographical degrees, or left in its default `FALSE` value if they are in km (the distance unit used internally in the package).

std Type of normalization. By default, data will be studentized using both the mean and the standard deviation, but if your variable has a natural zero (e.g., precipitation), `std=2` can be preferable (data will be normalized just as ratios to the mean values). Another option is `std=1`, for only applying differences to the mean values.

rtrans Root transformation to apply to the data: `2` for square root, `3` for cubic root, etc (fractional numbers are allowed). Useful if your variable distribution is far from normal, as with wind speeds or precipitations from arid regions).

na.strings Character string to be treated as a missing value. It defaults to the R standard `"NA"`, but can be set to any other string as, e.g.: `na.strings="-999.0"`.

Another example to homogenize seasonal precipitations (four data per year) for the period 1961-2005, with station coordinates in geographical degrees, applying a cubic root transformation to the data (no example file provided):

```
homogen("SsPrp", 1961, 2005, nm=4, deg=TRUE, std=2, rtrans=3)
```

The command of the first example would generate the following files (in the same working directory):

Tmin_1956-2005.esh Station file after the homogenization. It has the same structure of the input file `Tmin_1956-2005.est`, but with additional columns (see the extended

[2]The file `depurdat.R` holds the homogenization functions of the package

guide) and, probably, lines (when the process detects an abrupt shift in the mean, the series will be split, creating a new one with the same coordinates and adding an incremental number to the name and code of the station).

Tmin_1956-2005.dah Homogenized data file with missing data filled, analogous to the input data file `Tmin_1956-2005.dat`.

Tmin_1956-2005.txt Log file of the process, with all messages issued to the screen (including the final summaries).

Tmin_1956-2005.pdf File with a (potentially long) collection of diagnostic graphics generated during the process.

The log and graphic files may suggest to re-run the process with different parametrizations (see the extended guide for an explanation), while the homogenized data files may be post-processed with the function `dahstat`. For example, if we want a listing of average values for the period 1971-2000 from the above homogenized temperatures, we can get it in a file named `Tmin_1971-2000.med` with the command:

```
dahstat("Tmin", 1956, 2005, 1971, 2000)
```

As you can see, the parameters are the name of the variable, the first and last years of the study period, and the first and last years of the period for which we want the means to be computed (that can be omitted if the period is the same). Other parameters of the function are:

out Type of output (the file name will have the corresponding extension):
`"med"` for means of the data (the default).
`"mdn"` for medians.
`"max"` for maximum values.
`"min"` for minimum values.
`"std"` for standard deviations.
`"q"` for quantiles (see the `prob` parameter).
`"tnd"` for trends.
Any unrecognized option will just read the homogenized data, allowing you to apply your own analysis on them.

vala Annual value computed in the listing. Can be set to `0` (no annual value will be computed), `1` (sum of the monthly or other sub-annual data), `2` (mean of the data; the default), `3` (maximum) or `4` (minimum).

prob Probability for the computation of the quantiles (if option `out="q"` is used. Default value: `0.5`, which is the same as the median).

eshcol Columns of the homogenized station file `"*.esh"` to be included in the output file. Its default value is `4`, indicating that only the code of the station (the fourth column) will precede the computed statistics.

The output files will have the base name with an extension equal to the chosen `out` option, with the exception of the quantiles, which will have an extension `qPP`, where `PP` will be replaced by the probability set with the `prob` option (in %).

Therefore, if we want to obtain 1971-2000 monthly normals from the previously homogenized minimum temperatures, we could issue the following command:

```
dahstat("Tmin", 1956, 2005, 1971, 2000)
```

But if we try to compute the trends for the whole period of study 1956-2005, including the coordinates of the stations (columns 1 and 2 in the `Tmin_1956-2005.esh` output file) after the station codes, we should do:

```
dahstat("Tmin", 1956, 2005, out="tnd", vala=1, eshcol=c(4,1,2))
```
[3]

and in this way we would obtain the list of the trends in a text file called `Tmin_1956-2005.tnd` that, by including the site coordinates, would be suitable to produce a map (either within R or importing it into a GIS).

(End of the quick guide)

[3] Note the use of the concatenation R function `c` to provide a vector of numbers.

Extended Guide

Contents

1. Introduction

As the reader most probably knows, meteorological stations are not only recording the local climate variations, but rather their measurements are also affected by changes in instrumentation, methods of observation, relocations and changes in the environment (e.g. urban growth or land use changes). Hence, we call homogenization to the statistical process that tries to remove this unwanted perturbations and let the climatological series to reveal climate variations only.

This problem has been recognized since many decades ago. Some old methods relayed on tests to check the non stationarity of a single climatological series. This *absolute* methods must be avoided, since they assume a climate stability that has proved unrealistic. The alternative is to use *relative* homogenization methods, in which the stationarity test are applied to series of ratios or differences between the problem station and one or more well correlated series from neighbor stations. Peterson *et al.* (1998) and Aguilar *et al.* (2003) provide reviews of the different approaches developed by climatologists so far, while the next section explains the strategy followed in this package.

2. Methodology

2.1. Type II regression

As in many other methods, homogeneity tests are applied here on a difference series between the problem station and a reference series constructed as an (optionally) weighted average of series from nearby stations. But unlike most of them, the selection of the these stations is based on proximity only, disregarding the correlation criterion, in order to be able to use the nearest stations even if they have a too short (or none) common period of observation for correlations to be safely computed. Therefore, while the use of correlations is usually constrained to selected long series, we are able to use as much information as possible from our climatological network. This implies, however, that the region under study should be climatically homogeneous[4], since the presence of sharp geographical boundaries can lead to the use of nearby badly correlated stations to compute the reference series. In this case, the region should be subdivided and the homogeneity process independently applied to every sub-region.

This approach was inspired by the method used by Paulhus and Kohler (1952) to fill missing daily precipitation data, consisting in a spatial interpolation of the rate to normal precipitation of neighbor stations. This proportion method is extended in the *climatol* package with options to use differences and full studentization to normalize the data. Proportions (or ratios) to normal climatological values are appropriate for precipitation and other zero-limited variables with L-shape probability distributions, while differences to normals (or studentizations, if this differences are further divided by the standard deviation) are most suited to temperature and other (near) normally distributed variables.

From the statistical point of view, this is equivalent to apply a type II linear regression model, instead of the far more known type I. The latter is normally computed by a least squares adjustment, minimizing the deviations between the points (observations) to the regression line in

[4] Or, at least, that the climate varies smoothly throughout the studied region.

the Y axis direction (vertically, as if figure 1-left). The underlying assumption is that the independent variable X is either controlled by the investigator or measured with neglectable errors (Sokal and Rohlf, 1969) . But this is not the case when adjusting regression lines to pairs of series of a climatological network, where the errors are *a priory* similar in all stations. In this case, the deviations to minimize should be computed perpendicularly to the regression line, as in figure 1-right.

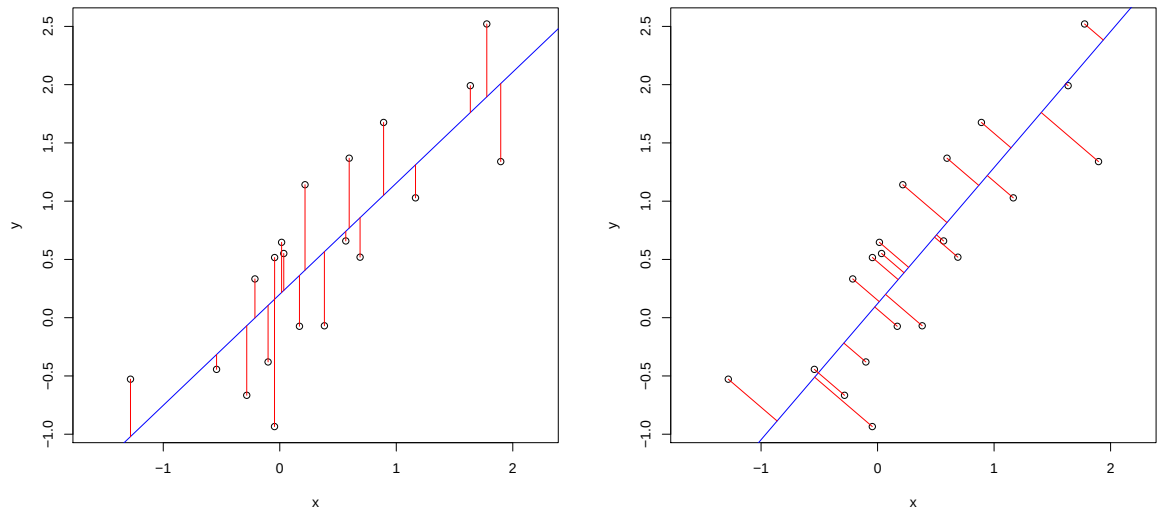

Figure 1: Deviations minimized by Ordinary Least Squares (type I regression, left) and Orthogonal Regression (type II regression, right).

There is a least squares analytical expression for computing this type II orthogonal regression line (Daget, 1979), but there are a few alternatives that provide a very close approximation. The simplest is called *reduced major axis* which, if we name x and y the standardized versions of the dependent and independent variable ($x = (X - m_X)/s_X$ and $y = (Y - m_Y)/s_Y$, where m and s stand for the mean and standard deviation respectively), has the form:

$$\hat{y} = x$$

(Or $\hat{y} = -x$ when the relation is inverse, which is not the case when dealing with the same variable in a climatically homogeneous region).

A characteristic of this type II regression is that the variance of the estimated variable is the same as that of the original variable, since this line does not tend to the horizontal when the coefficient of determination (r^2, equal to the fraction of explained variance) tends to zero. It can be argued that, when this fraction is lower than one, the extra variance provided by the type II regression with respect to the Ordinary Least Squares (OLS) counterpart is spurious. But we expect high values of r^2 if the observational network is dense enough, and on the other hand we will avoid the undesired effect of a reduced variance when the assessment of the variability of the series is the final goal of the climatic study. In addition, this approach provides a means to not only adjust for changes in the average of a series, but for changes in variance also[5].

[5] Although changes in the variance of the series are not searched in this package.

2.2. Data estimates

Once the original data are normalized, we estimate every term of each series as a weighted average of a prescribed number of the nearest available data. The weights to be applied to the reference data can be all the same (plain average) or be computed as an inverse function of the distance d between the observing sites. The function originally chosen for this was $1/(1+d^2/a)$, where the parameter a allows the investigator to modulate the relative weight of nearby stations to the more distant ones, but it is more conveniently formulated as $1/(1+d^2/h^2)$, since in this way the new parameter h becomes the distance at which the weight is half that of a station placed in the same location of the data being estimated[6]. In figure 2 this function is plotted for different values of h. (The parameter h is called *weight distance*, `wd`, in the parameter list of the homogenization function of this package).

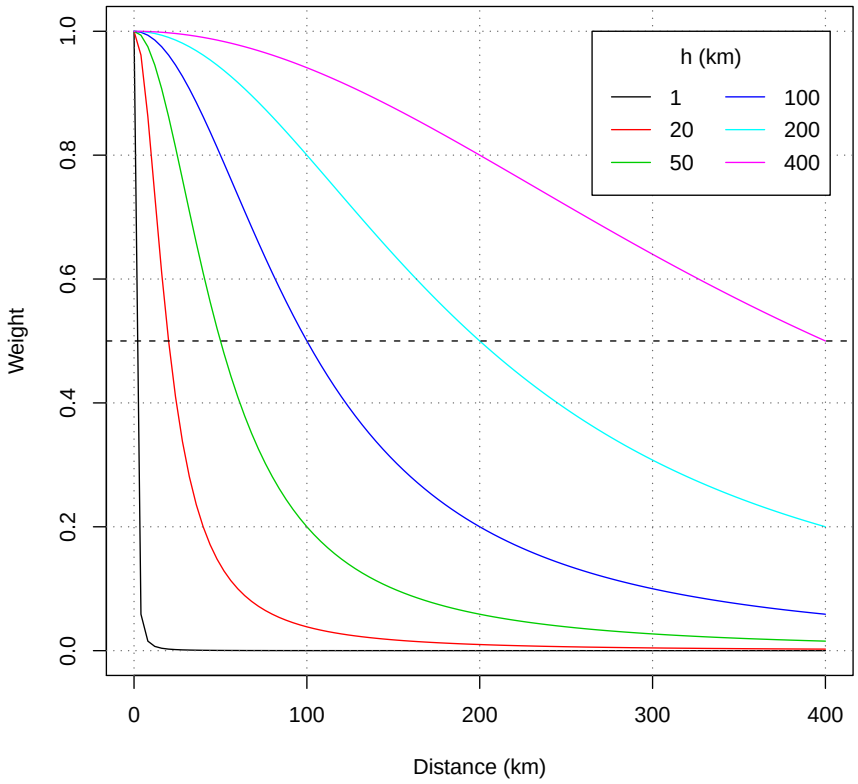

Figure 2: Different shapes of the weighting function according to the weight distance h (parameter `wd` of the `homogen` function).

But the first problem we must face is that, unless the series are complete, we cannot compute their means and standard deviations for the whole study period. We must then begin by computing this parameters from the available data only, use the estimated series (after undoing normalization) to fill the missing data, recompute the means and standard deviations, re-normalize the data, and obtain new estimates of the series. This process is repeated until the maximum change in a mean is less than a chosen amount (0.005 units by default).

[6] Thanks to Victor Venema for this suggestion.

2.3. Outlier and sharp shift detection and correction

After having estimated all the data, for every original series we can compute a series of anomalies (differences between the normalized original and estimated data), and apply to them tests for the detection of:

1. Outliers: The series of anomalies is standardized, and anomalies greater than 5 (by default) standard deviations will result in the deletion of their corresponding original data.
2. Shifts in the mean: The Standard Normal Homogeneity Test (SNHT, by Alexandersson, 1986) is applied to the anomaly series in two stages:
   a) On windows of 120 terms moved forward in steps of 60 terms (default values).
   b) On the whole series.

The maximum SNHT test values (called `tV` in this package) and their locations for every series are retained, and the series with the greatest value, if higher than the default threshold, is split at the point where this maximum has been computed. Values after this break point are transferred to a new series (with the same coordinates) and deleted from the original series.

Ideally, after the first split of a series, the whole process should be repeated, since that inhomogeneity may have influenced the homogeneity assessment of its nearby series. But this can lead to a very long process when dealing with a big number of stations with many inhomogeneities, and therefore a tolerance factor is provided to allow several splits at a time.

When all inhomogeneities detected over the prescribed threshold in the stepped SNHT test have been removed through the split process, the SNHT is applied again to the whole series, possibly generating more breaks in the series.

The stepped test has been implemented to prevent multiple shifts in the mean from yielding misleadingly low SNHT results, while the application to the whole series is more powerful for detecting smaller shifts that may have passed inadvertently to the stepped test. Anyway, the default threshold of SNHT for the whole series has been set greater than for the stepped application, to avoid the series been split due to a smooth trend instead of an abrupt shift in the mean (although steep local trends will be detected and treated as if they were shifts).

After all inhomogeneities over the set thresholds have been eliminated, a final stage is performed, devoted entirely to missing data recalculation (including the data removed by the outliers and shifts detection stages). This applies to all the series, either original (not split series or first fragments of the split series) or derived (new series created by the split process). Despite the number of reference data, the last missing data filling of the fragmented series is computed using only the reference of the other fragments.

3. Application

3.1. Preparing the data

Climatological and station coordinates data must be provided in the way explained in the quick guide in order to be properly read by the homogenization function. Alternatively, you can read them with your own R functions, allowing you to read them from files with a different structure or to take advantage of the R procedures to access Relational Data Bases. The only precaution is that your data must end in the R memory space in the two objects:

dat Numerical matrix containing the data, with dimensions `nd, ne` (where `nd` and `ne` stand for number of data per station and number of stations, respectively). Missing data must be assigned the standard R `NA` value.

est.c Data frame with five columns `X Y Z Code Name`, containing the coordinates (`X` and `Y` may be expressed either in geographic degrees or in km, and `Z` in m), codes and names of the stations. The ordering of these lines must be consistent with that of the data blocks in the `dat` object.

3.2. Homogenizing the series

The homogenisation function of this package is called `homogen`, and must be provided, at least, with three parameters:

varcli Acronym of the name of the climatic variable under study.

anyi Initial year of the study period.

anyf Final year of the study period.

This three parameters have no default values, and they will be used by the function to build the base name of the input and output files of the process, as explained in the quick guide. The other (optional) parameters accepted in the call to the function are the following:

nm Number of data per year in each station (`12` by default: monthly values. Set to `nm=1` if you are analyzing annual data, `nm=4` for seasonal data, etc).

nref Maximum number of reference data to be used in the estimation of each data. As explained in the methodology section, all data are estimated as if they were all missing (in order to compute the anomalies), as a weighted average of the nearest data[7]. This parameter sets the maximum number of data to be used, if more are available. (`10` by default).

[7]Note that we are talking about *nearest data* and not *nearest stations*, since the available data will likely be changing along the study period.

dz.max Threshold of outlier tolerance. By default, anomalies greater than 5 standard deviations (of the anomaly series itself) will be rejected (conservative figure).

wd Distance (in km) at which data will have half the weight of a station located at the same site of the series been estimated. The default values are 0 for the first two stages (meaning that all the reference data will have the same weight), and 100 for the last stage of final missing data re-computation. You can provide a vector of three values, one for each stage, as in `wd=c(0, 200, 50)`. Any additional value will be disregarded, while the last value will be repeated if the vector has less than three elements.

tVt Threshold value of the stepped SNHT window test (25 by default).

tVf Tolerance factor to split several series at a time. The default is 0.02, meaning that a 2% will be allowed for every reference data. (E.g.: if the maximum SNHT test value in a series is 30 and 10 references were used to compute the anomalies, the series will be split if the maximum test of the reference series is lower than 30*(1+0.02*10)=36. (Set `tVf=0` to disable a split if any reference series has already been split at the same iteration).

swa Size of the step forward to be applied to the windowed application of SNHT. The default value is 60, meaning that the test will be applied to the first 2*60 available terms of the series, and then this 120 window will be skipped 60 terms forward for another test, and so forth until the end of the series is reached. This default value is suitable to monthly series, but too big for annual and possibly to low for daily series.

snhtt Threshold value for the SNHT test when applied to the complete series. Defaults to 50 (a conservative value), and can be set to 0 to skip this test.

mxdif Maximum data difference in consecutive iterations. The iterative computation of means (and, optionally, standard deviations) of the series will stop when the maximum difference of any data is at most equal to this parameter, set by default to 0.05.

force Boolean parameter to force the split of series even when only one reference station is available. Defaults to FALSE.

a Constant to be added to the data just after reading them from the input file. Provided, in combination with the following b parameter, as a means to apply a linear transformation to the data. E.g., if the original data are in a different unit than the desired working unit. (Defaults to 0).

b Factor to be applied to the data (1 by default).

wz Factor to apply to the station altitudes before computing the matrix of euclidean distances. By default it has a value of 0.001, to give to the vertical coordinate (in m) the same weight than to the horizontal coordinates (in km).

deg Set to TRUE if the input coordinates are in geographical degrees, or left in its default FALSE value if they are in km (the distance units used internally by the package).

rtrans Root transformation to apply to the data: 2 for square root, 3 for cubic root, etc. (Fractional numbers are allowed; useful if your variable distribution is far from normal, as with wind speeds or precipitations from arid regions).

std Type of normalization. By default (3), both the mean and the standard deviation will be applied, but if your variable has a natural zero (as is the case with precipitation), `std=2` can be preferable (data will be normalized just as ratios to the mean values). Another option is `std=1`, for only applying differences to the mean values.

ndec Number of decimal digits to which the homogenized data must be rounded (`1` by default).

mndat Minimum number of data for a split fragment to become a new series. If leaved to its `0` default value, it will be set to half the value of the `swa` parameter when applied to daily data, and to the value of `nm` otherwise, with an absolute minimum of 5. (If this value is too low, the means and standard deviations of the series will be very poorly estimated, and the same will happen to the reconstruction of that series).

gp Graphic parameter. Set it to:

- `0`, to prevent any graphic output.
- `1`, to have only descriptive graphics of the input data (no homogenization will be performed).
- `2`, to produce also the diagnostic graphics of anomalies.
- `3` (the default), to get also the graphics of running annual means and applied corrections.
- `4`: as with `3`, but running annual totals (instead of means) will be plotted. (To be preferred when working with precipitation data).

leer [8] Set to `FALSE` if you read your data with your own R routines.

na.strings Character string to be treated as a missing value. It defaults to the R standard `"NA"`, but can be set to any other string as, e.g., `na.strings="-999.0"`, or even a vector of strings, as in `na.strings=c("-999", "-999.0", "-999.0")`.

nclust Maximum number of stations for the cluster analysis. By default, if the number of input series is greater than `100`, only a random sample of this size will be used for these descriptive initial graphics.

maxite Maximum number of iterations when computing the means of the series. Defaults to `50`, to avoid a too long processing time when convergence is very slow.

ini Initial date. Void by default, if set (with format `'YYYY-MM-DD'`) it will be assumed that the series contain daily data (see section 7 for a discussion on the limitations of such an application).

vmin Minimum possible value (lower limit) of the studied variable. Unset by default, but note that `vmin=0` will be applied if `std` is set to `2` (e.g., in precipitation or wind speed analysis).

vmax Maximum possible value (upper limit) of the studied variable. Unset by default, but useful to homogenize, e.g., relative humidity or relative sunshine hours (set `vmax=100` and `vmin=0` if these data are expressed as percentages).

[8] Spanish for *to read*

verb Verbosity. `TRUE` by default, may be set to `FALSE` to avoid the long output sent to the console. (It will be sent to the log file anyway, as explained in the following section).

As explained in the quick guide, the most trivial homogenization example with this function is:

```
homogen("Tmin", 1956, 2005)
```

You can reproduce this example after putting the appropriate data and station files in your R working directory. These files, named `Tmin_1956-2005.dat` and `Tmin_1956-2005.est`, are archived in `climatol-dat.zip`, available from http://webs.ono.com/climatol/climatol.html. The outputs of this example will be explained in the following section.

4. Outputs

The example application command `homogen("Tmin", 1956, 2005)` generates four output files, stored in the working directory:

Tmin_1956-2005.txt A text file that logs all the processing output to the console.

Tmin_1956-2005.pdf A PDF file with a collection of diagnostic graphics.

Tmin_1956-2005.dah A text file containing the homogenized data (with missing data filled). It has the same structure as the input data file `Tmin_1956-2005.dat`.

Tmin_1956-2005.esh A text file with the coordinates and names of the stations of the homogenized data file.

4.1. `*.txt` file

The log text file is meant to be self-explanatory. It begins by recording how the function was called, with all the parameter values (including the unmodified defaults), for future reference.

Then the convergent iterative computation of means and missing data filling follows, displaying the maximum mean difference to the previous iteration and identifying the code of the corresponding station. Outliers rejected during this process appear in lines as the following:

```
S63(7) 1966 7:  21.1 -> 14.3 (stan=6.42)
```

These lines begin with the code of the station and, between parenthesis, its rank in the station list input file. Then, the year and month of the outlier follow, and, after a colon, the value of the original observation, an arrow, and the suggested correct value. At the end of the line, between parenthesis also, the standardized anomaly (standard deviation of the anomaly of the normalized observation) is given. Note that the suggested correct values appearing in this outlier rejection lines are only provisional estimations, since the final estimation of the missing values (including the rejected outliers) will have been computed at the final stage of the process.

After the iterative computations of series averages (and standard deviations, if the default `std=3` setting is unchanged), the shift analysis results are presented. For every series, identified by its

ordinal number, the maximum value of the SNHT test (`tV`) is shown. And when all series have undergone their tests, the one (or more) that scored the maximum value is split, and the record of this process is reflected in this file in lines like, e.g.:

```
M56(10) breaks at 1976 7 (95.1)
```

The code and ordinal number of the station appears as in the outlier rejection lines, followed by the year and month of the split point, and the value of `tV`, within parenthesis. The given break point here is always the first term after the shift, and from this term until the last one of the series are moved to a new series attributed to a new station, with identical coordinates as the original series and code and name formed by appending an increasing number to the primary ones.

These blocks of iterative computation of means (with possible outlier removals) and break analysis is repeated several times as the process goes through stages 1 (stepped forward window SNHT tests) and 2 (classical whole series SNHT tests), and a final stage 3 is undergone to compute the final estimation of missing data (this time without shift analysis).

The log file ends with a set of final computations, including:

ACmx Maximum absolute auto-correlations. The R `acm` auto-correlation function is applied to the anomaly series, and the maximum absolute value of all lags is retained for every series. High auto correlation values may give an indication of lack of randomness, and attention should be given to those series.

SNHT Standard Normal Homogeneity Test of the final series of anomalies. This is to evaluate the remaining inhomogeneities of the output series of the process.

RMSE Root Mean Squared Error of the estimated data. This is computed form the differences between the observed and estimated data, when both are available. It serves to give an idea of the errors involved in the estimation of the missing data, and may help to choose the best parameters when different applications of the `homogen` function are performed. On the other hand, high RMSE values may indicate either a bad quality of the original series or a singularity of the site of that station, in the sense that it could be placed in a location with a special micro-climate that is not affecting their neighbors.

PD Percentage of original Data. When a series is split in two or more fragments, this value helps in identifying which one is retaining most of the original data (the longest fragment).

Summaries of these four magnitudes are given first, and then their values are displayed for every series (primaries and derived).

4.2. `*.pdf` file

A potentially long (depending on the `gp` setting) series of diagnostic graphics are also produced by this function. The first figures are dedicated to a description of the input data: overall number of available data (figure 3), box-plots (monthly if applicable, as the figure 4 January example), and a histogram (figure 5). Big outliers or any major problems in the input data revealed by these graphics may suggest a corrective action prior to repeating the homogenization process.

The following figure is a plot of correlation coefficients versus distance (figure 6). The correlation coefficients are computed from the first differences of the series to avoid the impact of inhomogeneities, and all available pairs of observations have been used. Only computed correlations of 1 and -1 have been removed from the correlation matrix, since they must come from series having only two pairs of common observations, but be aware that some of this correlations may have been computed from as few as three data points. Although this coefficients are not going to be used in the homogenization process, this plot is useful to assess the smoothness of space-climate variations, or otherwise the existence of possible factors that compromise this correlation-distance relationship. In the example of figure 6, high and low correlations co-exist at short distances, indicating the impact of the different topography of the sites on the minimum temperatures in calm and clear sky nights.

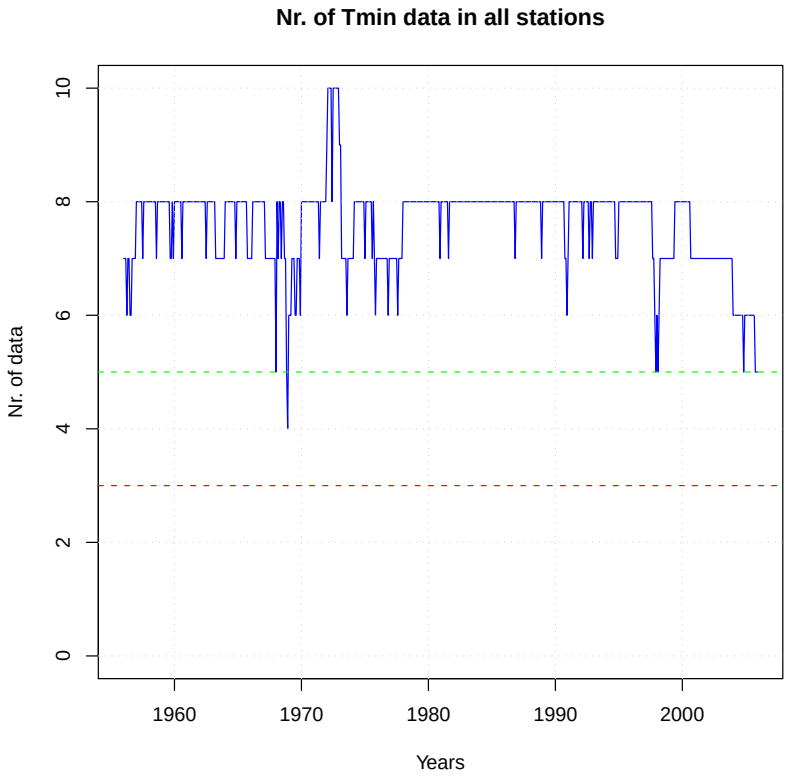

Figure 3: Overall number of available data.

A cluster analysis is then performed, based on the correlation matrix, that serves to produce two more figures: A dendrogram, where you can see the stations grouped by similarity of their data regimes, and a map locating the sites of the stations, identified by their ordinal numbers and in different colors according to their clusters. This is intended as a first approximation to a climatic classification of the stations, although the number of clusters, automatically chosen by the dashed red horizontal line in the middle of the dendrogram, will probably be not the best. If the clusters are very different (are connected by high dissimilarity distances in the dendrogram) and their spatial location depicts clearly delimited areas, the climate of the study area may be subject to strong discontinuities, and hence the investigator should consider doing separate homogenizations for each climatic subarea.

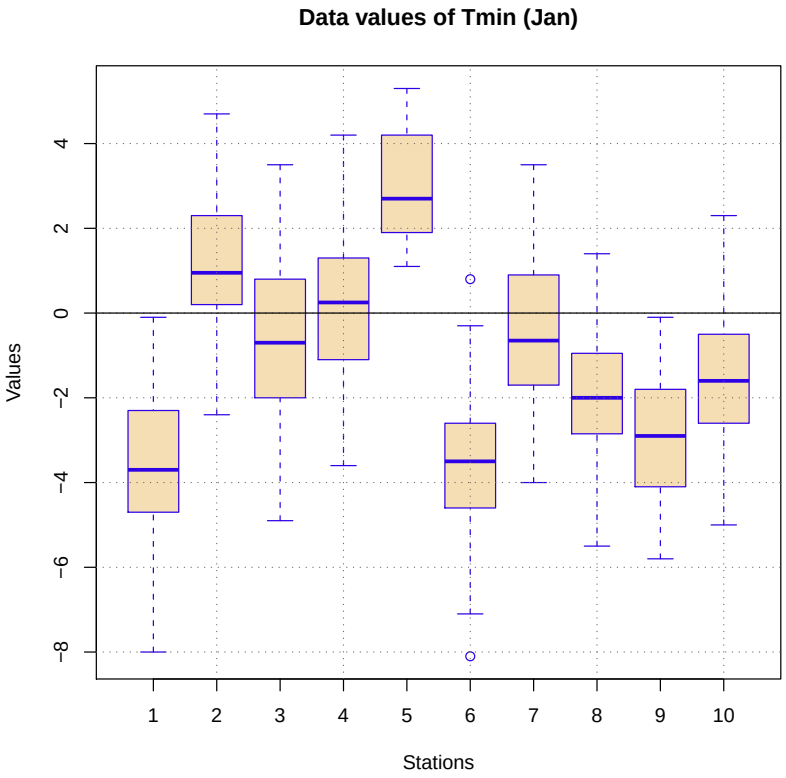

Figure 4: Example of monthly box-plots of the data.

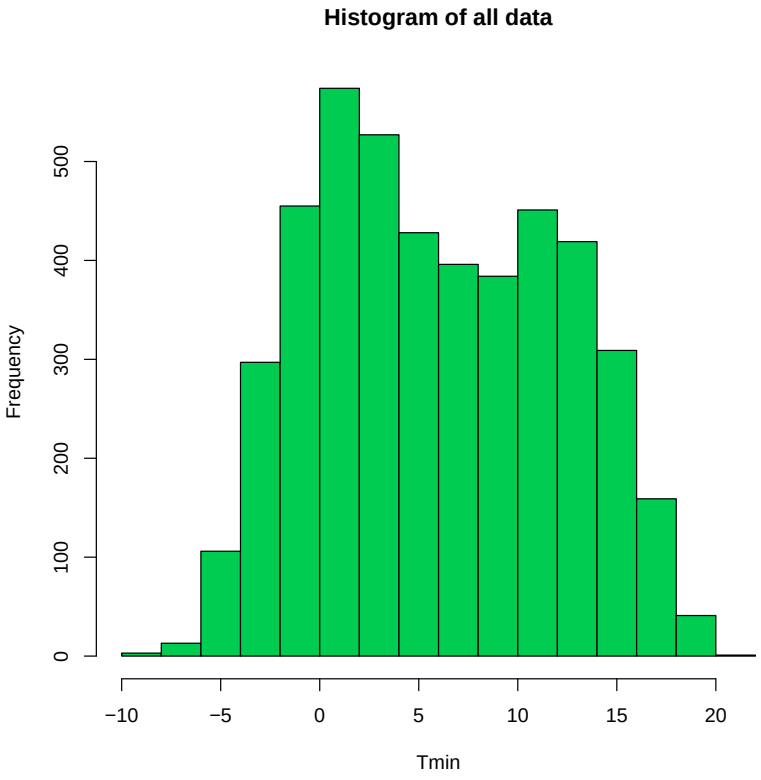

Figure 5: Histogram of all data.

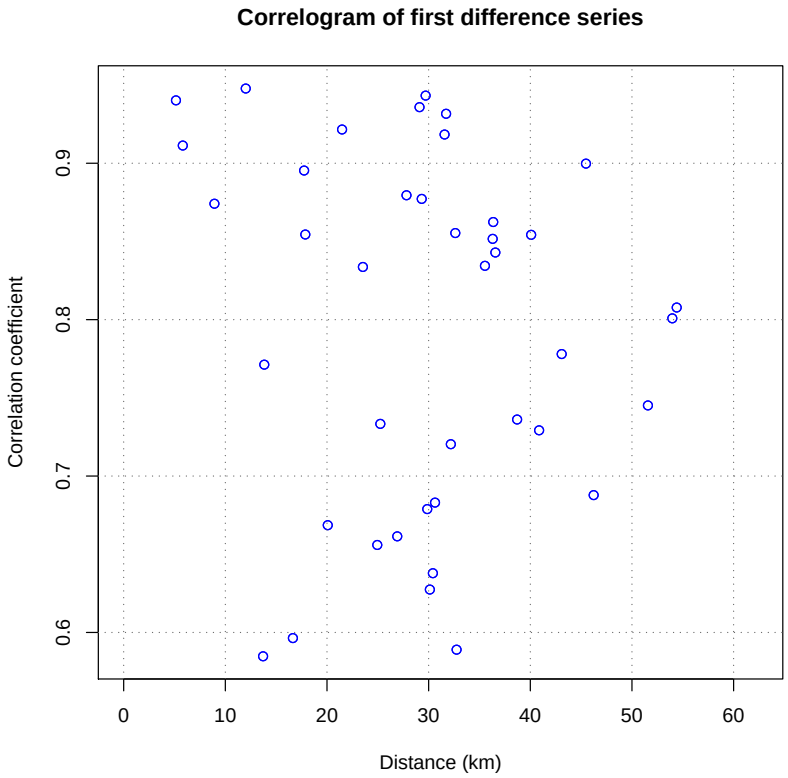

Figure 6: Correlation–distance plot.

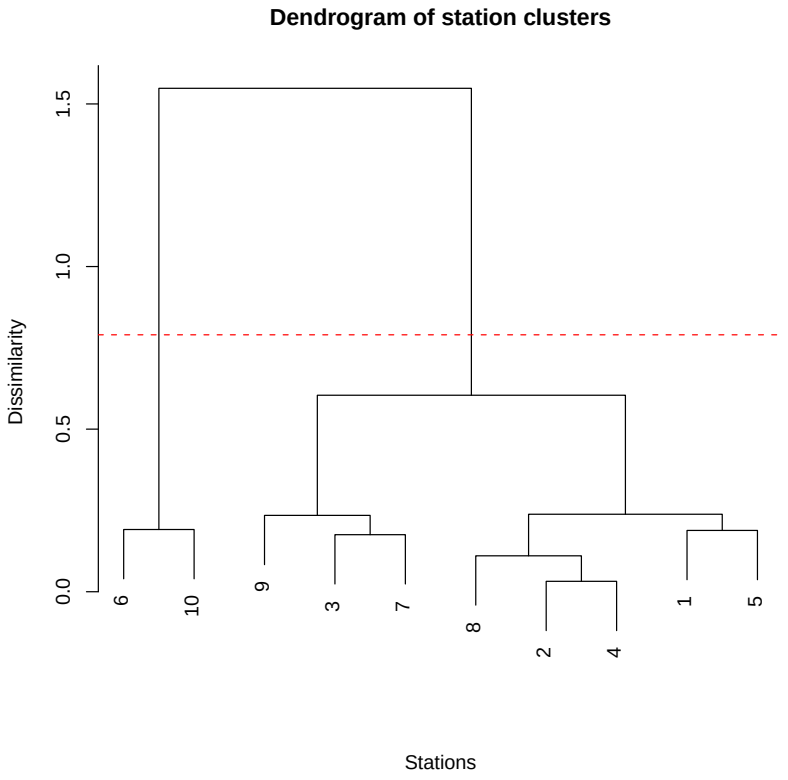

Figure 7: Dendrogram built from the correlation matrix.

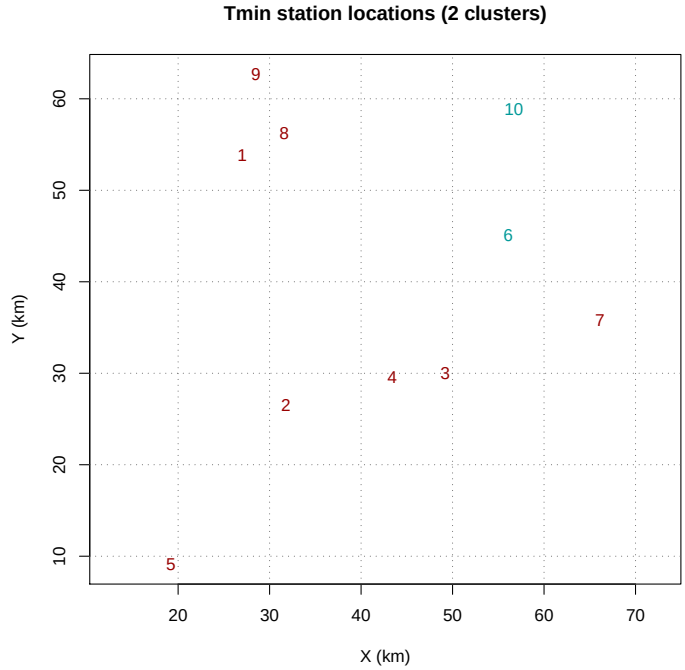

Figure 8: Map of the stations, colored according to their clusters.

After these descriptive figures, we enter those describing the analysis of the anomaly series, as if figure 9, with anomalies plotted as vertical blue bars. When the maximum value of the shift in the mean test is over the prescribed threshold, the location where the series is split is marked by a vertical red dashed line, and a number at the top shows the (floor rounded) value of the test. In the lower part of the figure, the minimum distance to the nearest reference data is graphed in green, in a logarithmic scale.

All split series are shown in a similar figure, allowing a quick visual inspection of the homogenization process and a subjective consideration about its performance. The first splits will probably be very clear (as in figure 9), while the final ones could be arguable, especially if the test threshold, `tVt`, was set too low. In this case, re-running the process with a higher threshold would be advisable.

After all the split anomaly graphs of the first stage, summarizing graphics are presented, showing the maximum shift test values of the resulting split series (figure 10, with colored bars turning from green to red when the values increase), and a histogram of all these values (figure 11). Both figures show the distribution of the maximum shift test values, allowing to judge if the higher values are showing series with prominent inhomogeneities or rather they are only the right tail of the shift tests distribution.

This block of anomaly series shift tests and splits is repeated for stage 2, where the SNHT test is applied to the whole series, with the bar and histogram summaries of the maximum values of the test in the resulting series at the end of the stage. Two other summarizing graphic are then appended: a histogram of the number of splits per station (figure 12), and a bar graph of number of splits per year (figure 13). An accumulation of many splits in the same year could point at changes in observational practices in a significant part of the network[9].

[9]Changes should never be simultaneously applied to all the network, since no reference data would be left to assess the effect of the changes.

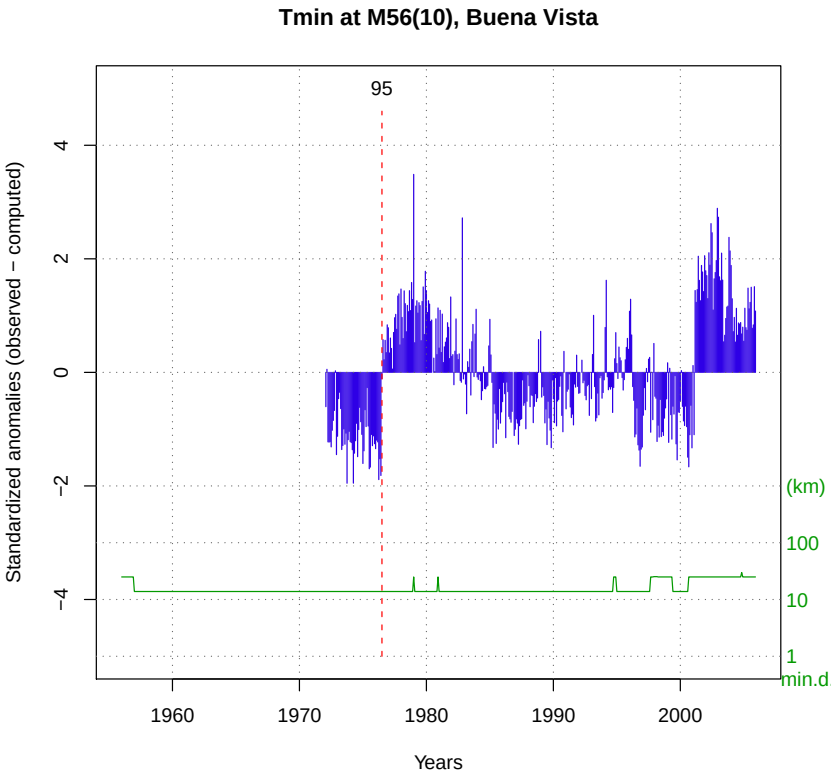

Figure 9: Analysis of the anomalies, marking the most significant break point.

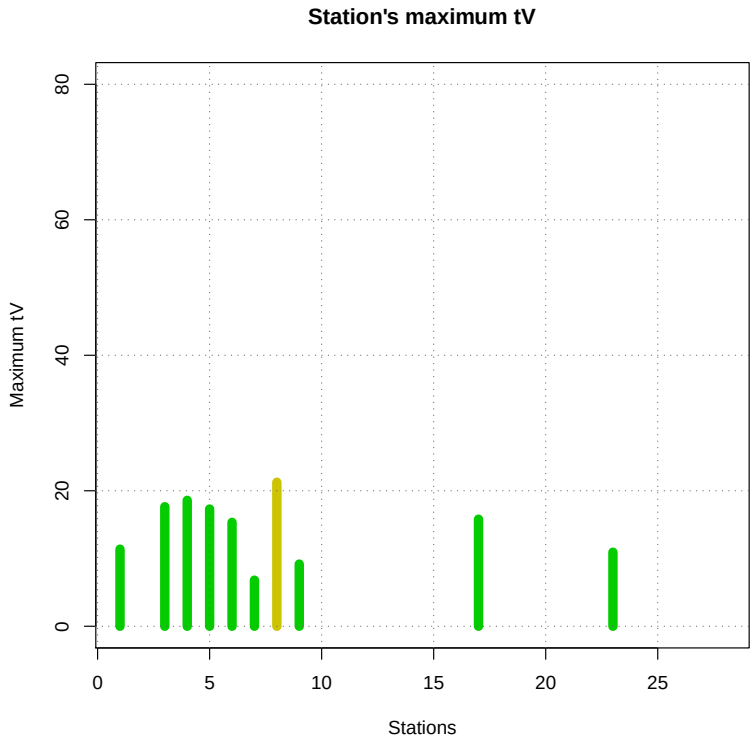

Figure 10: Remaining maximum shift test values of the resulting series after the splitting process. (Some stations display no bar because they have a too short period of observation for the stepped window SNHT test to be applied).

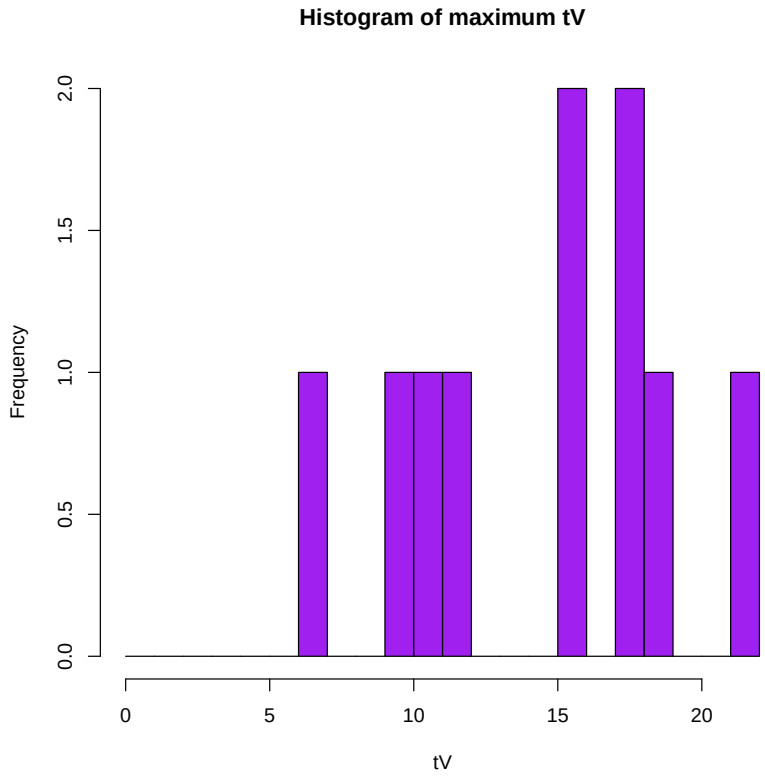

Figure 11: Histogram of the remaining maximum shift test values.

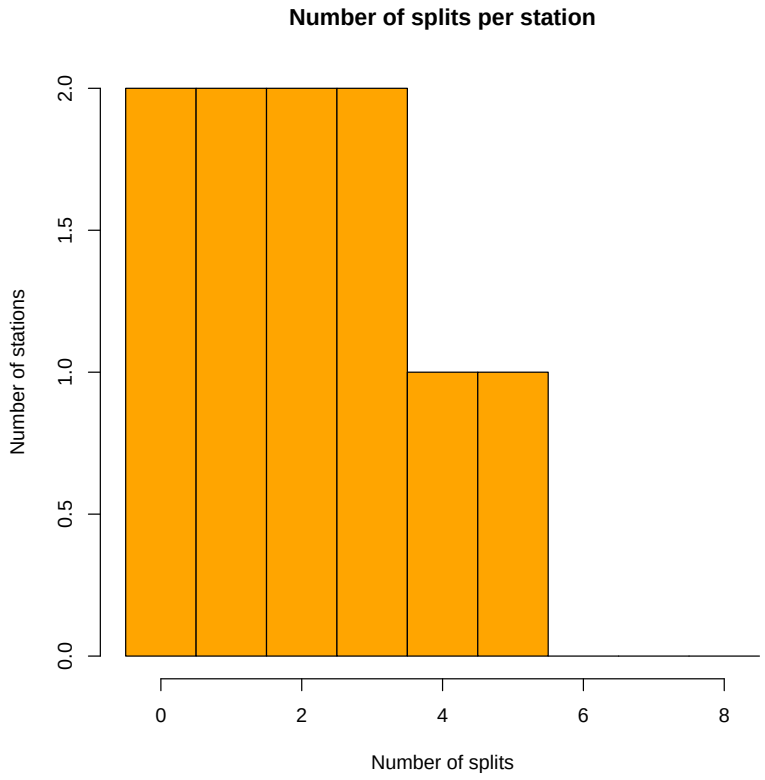

Figure 12: Histogram of number of splits per station.

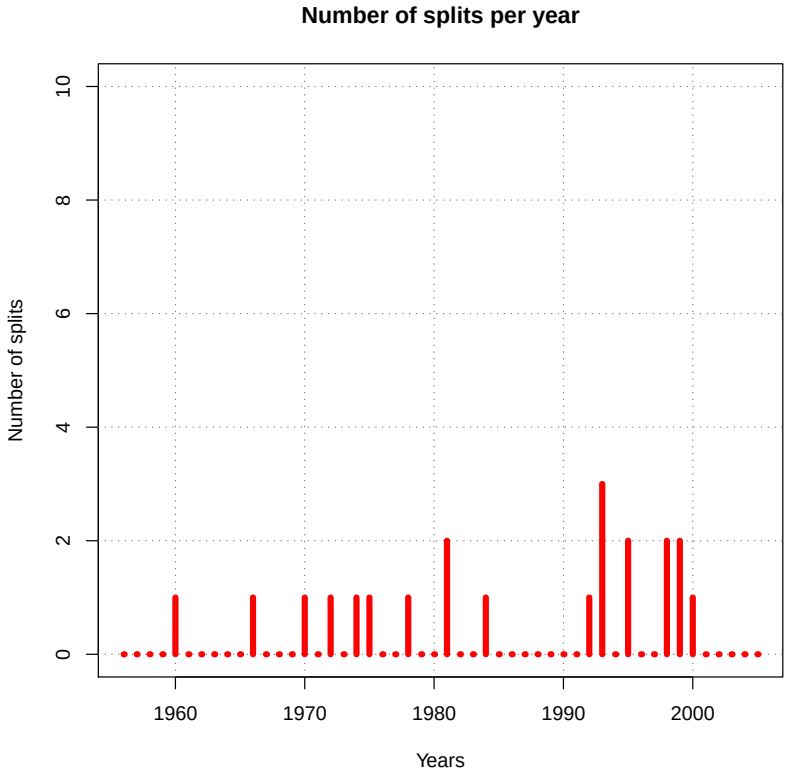

Figure 13: Number of splits per year applied through the homogenization.

As mentioned before, the third stage of the homogenization process is devoted to the final missing data estimation, including not only the original missing data, but also the rejected outliers and the data moved to a new series after a sharp shift detection and series splitting. This final stage generates two new blocks of figures: anomaly graphics, similar to those originated in stages 1 and 2, and final series and corrections applied.

Figure 14 shows one example of the final anomaly graphics, in which vertical dashed lines mark the locations of the maximum SNHT test values (the stepped one, in green, only if the series have enough data, `2*swa` at least, for its application). A regression line is drawn in blue if the trend of the anomalies is significant at the 0.05 level.

After the anomaly graphs of every final series (original or split), new graphs are produced for every original series showing, in the upper part, the running annual means (or totals, if `gp=4` is set), and in the lower part, the corrections applied for every reconstruction (see example in figure 15).

The last graphics include histograms of normalized anomalies (with frequency bars outside the prescribed outlier threshold filled in red), and of the maximum values of the shift tests (tVx and SNHT). Note that these may yield values higher than their corresponding thresholds if, as in the default values, the weight distance `wd` is lower in the third missing value recalculation stage than in the previous shift detection and correction phases.

The very last graphic of the PDF output file is a plot of RMSE-SNHT points (figure 16) where the quality (or singularity) of every reconstructed series can be inspected.

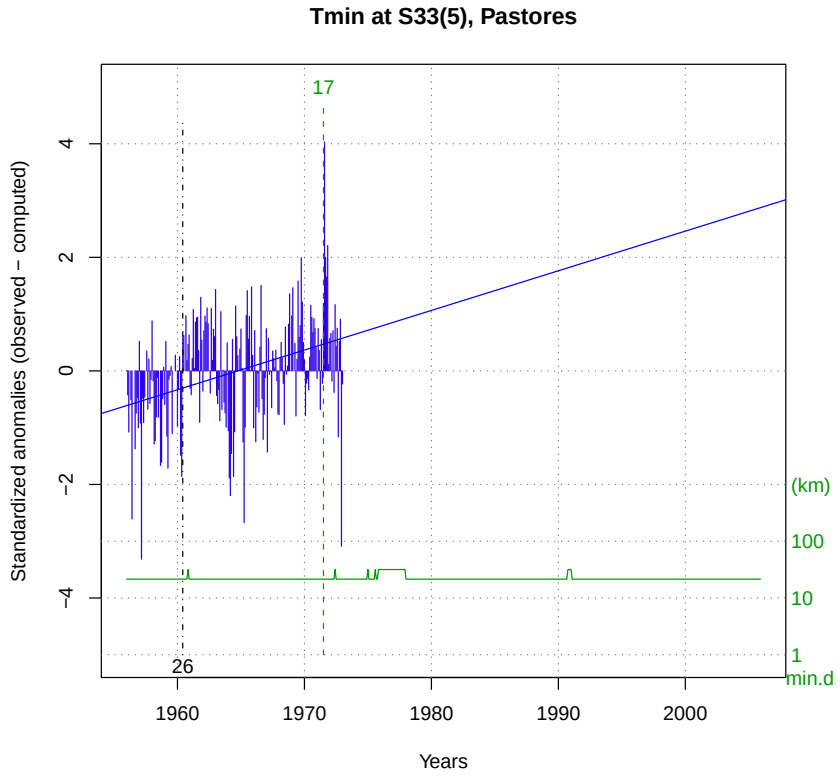

Figure 14: Anomalies of the final series, with maximum SNHT locations and general trend (if significant).

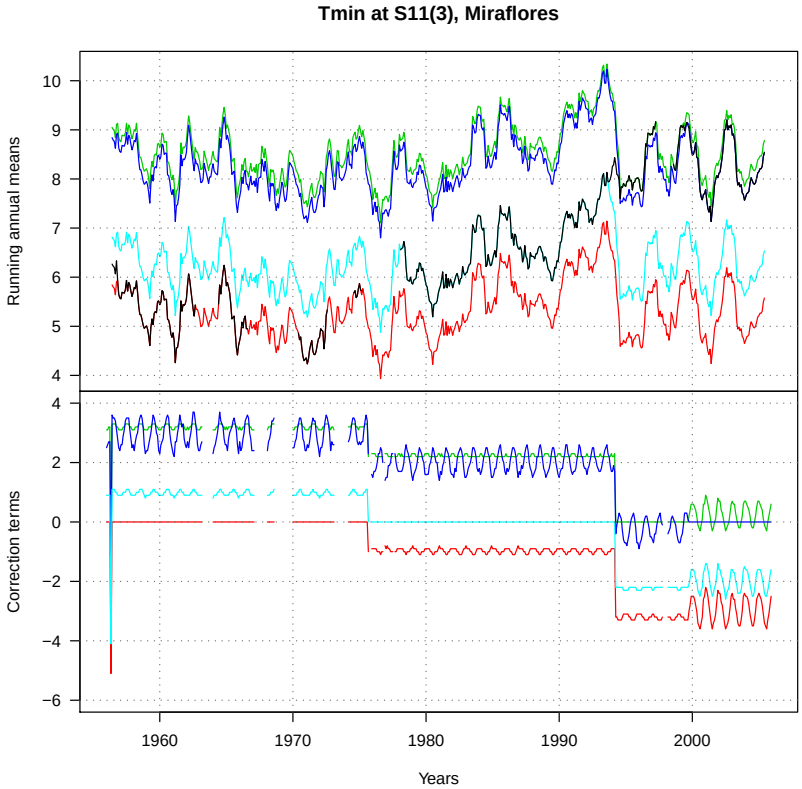

Figure 15: Original (in black) and reconstructed running annual series (top), and corrections applied to each fragment (bottom).

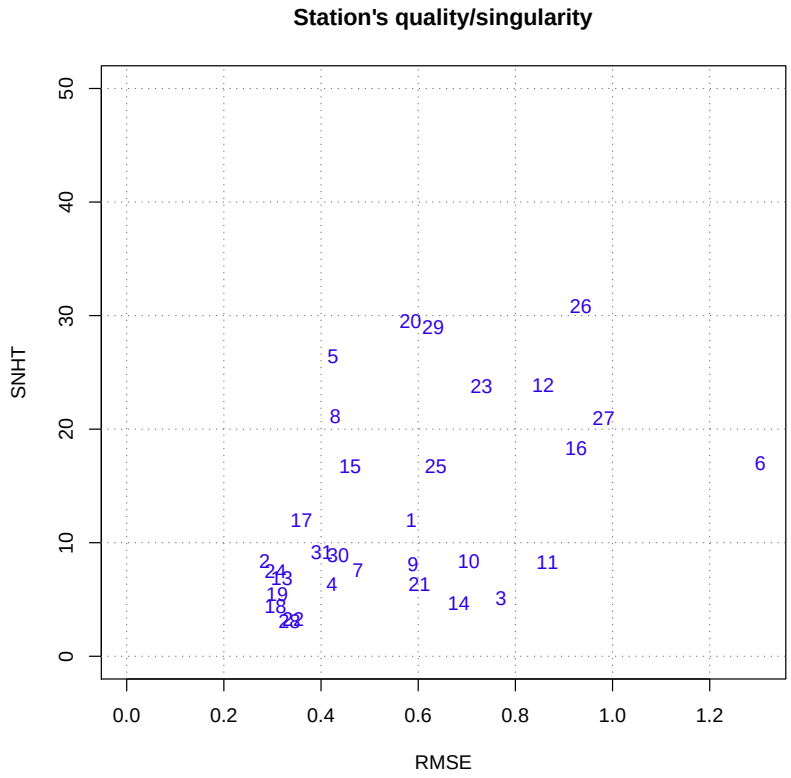

Figure 16: Plot showing the SNHT and RMSE of every final series (original or fragmented).

4.3. `*.esh` and `*.dah` files

The `*.esh` and `*.dah` files are the equivalents of the input files `*.est` and `*.dat`, but holding the results of the homogenization. However, the stations file `*.esh` will have additional information, as we can see in the first lines of the file `Tmin_1956-2005.esh` output in our example exercise:

```
27 53.9 456 "S03" "La Perla" 79 1 0 12
31.8 26.5 123 "S08" "El Palmeral" 11 2 0 8.4
49.2 30 154 "S11" "Miraflores" 31 3 0 5.1
```

In each line, the following items are related (the first five are the same than in the `*.est` input file):

1 Longitude, `X`.

2 Latitude, `Y`.

3 Altitude, `Z`.

4 Code of the station, `Cod`.

5 Name of the station, `Name`.

6 Percentage of original data, `PD`.

7 Index of the original station in the input data, `io`.

8 Binary flag marking whether the station was operating at the end of the study period (1) or not (0), `op`.

9 Maximum SNHT value, `SNHT`.

`X` and `Y` will be expressed in the same units (km or degrees) as in the input file. As to the index of original station (`io`), its purpose is to identify which fragments belong to the same original series. E.g., the eighth station in our exercise, Esmeraldas, has been split twice. Therefore, three fragments appear in the `Tmin_1956-2005.esh` file (for which completely reconstructed series are available in `Tmin_1956-2005.dah`):

```
31.6 56.2 498 "S40" "Esmeraldas" 48 8 0 21.1
31.6 56.2 498 "S40-2" "Esmeraldas-2" 7 8 0 5.5
31.6 56.2 498 "S40-3" "Esmeraldas-3" 8 8 0 3.1
```

From these lines (not consecutive in the file) we can see that they all belong to the same original series because: a) They share the same coordinates; b) Their codes and names are the same, except for a numerical suffix that has been appended to provide a way to differentiate them; and c) their `io` value is the same (8). But note that the numerical suffixes are in no way informing about the chronological order of the fragments in the original series, since they are created by order of shift test importance. In our example, if we search for the words `S40` and `breaks` in the `Tmin_1956-2005.txt` log file, we find the following two lines, that indicate that the first split (originating the `S40-2` series) happened in March 2000, while the second split took place earlier (in March 1996), hence giving birth to the `S40-3` series:

```
S40(8) breaks at 2000 3 (47.2)
S40(8) breaks at 1996 3 (28.5)
```

5. Discussion and suggestions

If you need quickly homogenized values for your project, you will be tempted to use the homogenization function as a black box, but it is advisable to revise the output files to see if the parameters used, whether set by the user or with their default values, are fit to your particular climatic network. Take into account that the optimal values of the parameters will vary according to the climatic element under study, its spatial variability, and the temporal and spatial density of the observations, and hence no universal default values can be provided.

Moreover, the chosen parameters can be optimal or not depending on the final purpose of the series analysis. E.g.: If you want to obtain climatic normals, the variance adjustments will have no importance, while they can be crucial if deriving extreme value return periods from the

series. In the latter case, you can limit the variance diminution of the weighted estimates by setting a short weighting distance in the third stage (e.g.: `wd=c(0,200,30)`), or totally avoid it by using only one reference data in this last data re-computation stage (`nref=c(10, 10, 1)`).

Therefore, you should look at the diagnostic graphics and see if there are remaining inhomogeneities that should be corrected, in which case the shift correction thresholds `tVt` and/or `snhtt` should be lowered, or if too low values of these thresholds have produced an excessive fragmentation of the series. Critical values of SNHT can be found in the literature (e.g., Khaliq and Ouarda, 2007), and reference values are also discussed in the annex at the end of this document.

Similarly, depending on the kurtosis of the studied variable, too many (or too few) outliers may have been deleted. The default value, 5 standard deviations, is rather conservative. You may adjust it to your needs, and even set different values for each of the three stages of the process. E.g., `dz.max=c(6, 3.5, 9)` would remove only the more outstanding outliers in the first stage, and would be more drastic in the second, while avoiding any outlier removal in the last stage (unless very big outliers appear, which could only happen if the number of references has been very much reduced in this stage).

Do not forget to set `deg=TRUE` if your coordinates are in degrees, as well as to choose the appropriate normalization type, preferring `std=2` for zero limited climatic variables (as precipitation or wind speed) and applying a root transformation if their histograms show a clear L-shaped distribution. Note that `std=1` will apply constant corrections to the data, and therefore no seasonal differences in the inhomogeneities will be accounted for, nor any variance adjustment will take place.

If you are homogenizing a reduced number of series, it is advisable to set `tVf=0`, to avoid too many splits at a time. In these cases, you may face a situation in which, at some time in the period of study (more likely at the beginning, normally with less observing stations), you have data only in one or two series. One data item is the absolute minimum at any time step for the homogenization process to be able to proceed, but in this points, due to the lack of references, no outlier nor shift can be detected, and the corresponding missing data in all other series, whether near or far away, will be filled with the only available value. On the other hand, at the time steps where only two stations have observations, the outlier and shift tests can be performed, but if the values are greater than the prescribed thresholds, no decision can be made about which of the two series is the one to be pinned with the inhomogeneous label. Therefore, no outlier deletion nor split is made in these cases, which are merely reported in the log output file with the annotations:

For outliers: `... Only 1 reference! (Unchanged)`
For shifts: `... could break at ..., but it has only one reference`

(Dots would be replaced by the relevant information about the involved station and date of the suspect data).

In these cases, the only way to decide which of the two suspect shifts is the real one relays on metadata. The history of the stations may give light about which of the stations was relocated or suffered any change that could account for that shift in the mean of the observations. If this information is available, we could then manually split the inhomogeneous series and rerun the homogenization process.

The importance of keeping records of all kind of changes affecting a station or its surroundings can never be sufficiently stressed, and several homogenization methods relay heavily on them (Aguilar *et al.*, 2003). Unfortunately, metadata are often incomplete or entirely lacking, and therefore this package is not using them, although they can be useful to check the results of the process.

Another possibility when trying to homogenize a few series and having only one or two available in some sub-period is to complement the observational series with others derived from reanalysis products in nearby grid points, but if your series begin before mid-XXth-Century these reanalysis may be difficult to obtain.

In summary, it can be a good practice to try new homogenizations with different parameters and check which one is more satisfactory. To avoid overwriting your output files at every trial, you can rename them with the function `outrename`, that will append a suffix to their base name. E.g., if we want to keep the previous results as `Tmin_1956-2005-old.*`, we would use the command:

```
outrename("Tmin", 1956, 2005, "old")
```

Once you have tuned the application of `homogen` to your particular dataset with suitable parameters, you can keep them for future homogenizations of the same dataset as, e.g., if you are managing a data base in which new data are been included regularly. In these cases, re-homogenizing the dataset once a year is advisable, since the new data may confirm or reject suspect inhomogeneities near the end of the series.

6. Post-processing the output

After having obtained satisfactorily homogenized series, you will want to apply your own analysis to them to obtain statistical values and graphics showing temporal or spatial variations of the studied climatic element. To facilitate some of the more common statistical computations, the package includes the function `dahstat`, that can be called with the following parameters (default values within parenthesis), of which only the first three do not have default values (and hence must always be provided):

`varcli` Acronym of the name of the climatic variable under study.

`anyi` First year of the study period.

`anyf` Last year of the study period.

`anyip` First year of the period to compute (`anyi`).

`anyfp` Last year of the period to compute (`anyf`).

`nm` Number of data per year in each station (`12`).

`ndec` Number of desired decimal digits of the output (`1`).

`vala` Annual value to compute (`2`). Can be set to `0` (no annual value), `1` (sum of the monthly or other sub-annual data), `2` (mean of the data; the default), `3` (maximum) or `4` (minimum).

mnpd Minimum percentage of original data (0).

mnsh Minimum SNHT (0).

out Type of output (the file name will have the corresponding extension):
"med" for means of the data (the default).
"mdn" for medians.
"max" for maximum values.
"min" for minimum values.
"std" for standard deviations.
"q" for quantiles (see the prob parameter).
"tnd" for trends.
Any unrecognized option will just read the homogenized data, allowing you to apply your own analysis on them.

prob Probability for the computation of the quantiles (if option out="q" is used. 0.5 by default, which yields the same output than out="mdn").

func Only compute statistics on series of stations functioning at the end of the period of study (FALSE).

pernum Number of years on which to express computed trends (100).

eshcol Columns of the homogenized station file to be included in the output file (4).

sep String to be used for separating the output data (" ").

eol Line termination style (defaults to the *new line* code "\n").

Parameters mnpd, mnsh and func act as filters to produce results only for series that have those minimum percentages of original data and SNHT values, and to select only those stations working at the end of the period studied. No selection is performed by default, listing the desired statistic for all the reconstructed series.

By default, values are separated by spaces in the listings, but you can change this with the sep parameter. E.g., you can get semicolon separated values by setting sep=';'. (Remember that in R you can delimitate character strings either with single or double quotes).

The output files will have the base name with an extension equal to the chosen out option, with the exception of the quantiles, which will have an extension qPP, where PP will be replaced by the probability set with the prob option (in %).

Therefore, to obtain 1971-2000 monthly normals from the previously homogenized minimum temperatures, we would do:

```
dahstat("Tmin", 1956, 2005, 1971, 2000)
```

But if we want to compute the trends for the whole period of study 1956-2005, expressed in units per 10 years (instead of per century) with two decimals, and including the coordinates of the stations (columns 1 and 2 in the Tmin_1956-2005.esh output file) after the station codes, we should do:

```
dahstat("Tmin",1956,2005,out="tnd",pernum=10,vala=1,ndec=2,eshcol=c(4,1,2))
```
[10]

In this way we would obtain the list of the trends in a text file called `Tmin_1956-2005.tnd`, that could be imported by a GIS to produce a trend map. Alternatively, we could generate that map withing R, with the help of other packages (Bivand *et al.*, 2008). Example of the beginning of that output file:

```
"Code" "X" "Y" "Jan" "Feb" "Mar" "Apr" "May" "Jun" "Jul" "Aug" "Sep" "Oct" "Nov" "Dec" "Annual"
"S03" 27 53.9 0.04 0.01 0.05 -0.07 0.13 0.19 0.08 0.17 -0.21 0.1 -0.1 0.12 0.53
"S08" 31.8 26.5 -0.02 -0.01 0.15 0.06 0.17 0.21 0.05 0.13 -0.15 0.12 -0.04 0.08 0.75
"S11" 49.2 30 -0.01 0.05 0.08 0.05 0.24 0.23 0.11 0.12 -0.19 0.1 -0.09 0.07 0.76
...
```

7. What about daily (or sub-daily) data?

Increased concern has aroused during the last years about the need to homogenize daily data. This is a very difficult task, since the detection of shifts in the mean of a series is basically a signal/noise problem, and daily values are generally too noisy for a successful detection. Therefore, while work is being done in the quest of techniques to approach this challenging homogenization, this package should not be applied to adjust daily series for changes in the mean, unless that signal to noise ratio is high enough, either because the break is very big or the noise is very low.

One real example of very low noise was found when investigating a shift in the mean of a thermometric station located at an airport: the runway sensors of air temperature provided a very close reference to the problem climatological records, allowing to detect the date of a badly performed maintenance operation that resulted in a 0.9°C shift in the temperature. Due to the existence of this close reference, the detection of the break was made on the differences of the 10-minutes series, while it would have been very problematic otherwise even with the daily series.

The main problem when trying to homogenize such sub-daily series resides in the lack of synchronization of the measures of most (or all) climatic elements, since the passage of fronts, moving thunderstorms or convective cells will have a different timing throughout a region, transmitting this lack of synchronization to the meteorological observations of the different observatories. This may often happen even in daily precipitation data, when a rain shower around the observing time is differently assigned in two consecutive days depending on when the precipitation reached the gauge stations.

But, although the correction of shifts in the mean of daily series is generally not advised, it is best to perform the quality control of the data on these kind of series rather than on the aggregated, monthly data. Note that a 10°C error when reading or transcribing a daily maximum temperature becomes an error of only 0.17°C in the monthly mean temperature. Therefore, whenever possible, outlier detection and correction must be done on the original measurements, weather daily (the usual case in the cooperative stations) or at shorter intervals (normally at the Automatic Weather Stations), although in this latter case the aforementioned lack of synchronization may preclude the availability of reference stations.

[10] The `c` concatenation function is the standard R way to provide a vector of numbers.

To avoid conflicting file names between daily and monthly data, the former are distinguished in this package by appending the suffix `-d` to the acronym of the variable. E.g., while we have been working with monthly values located at the file `Tmin_1956-2005.dat`, the file containing the corresponding daily values would be called `Tmin-d_1956-2005.dat` (and the needed stations file `Tmin-d_1956-2005.est` would be a copy of `Tmin_1956-2005.est`). The call to the `homogen` function would be in this case:

```
homogen("Tmin", 1956, 2005, nm=0, tVt=0, ini="2007-01-01")
```

The output graphics in `Tmin-d_1956-2005.pdf` may reveal some big shifts in the mean that could be worth removing. In this case, a new application of `homogen` can be performed, setting an appropriate value of `tVt` (and possibly `snhtt`) instead of the zero value set to prevent any break analysis. Also important will be to set a big window `swa` for the stepped SNHT, since the persistence of some circulation types will induce highly auto-correlated daily anomalies, an effect more rarely seen when dealing with monthly values.

After having obtained daily values with missing data filled by `homogen`, the user may want to derive their corresponding monthly values. The function `dd2m` is provided for this *daily data to monthly* conversion, which has many parameters in common with the `dahstat` function: `varcli`, `anyi`, `anyf`, `anyip`, `anyfp` and `ndec`. The rest of parameters governing `dd2m` are already familiar to us because they are equal or similar to others already seen:

`ini` Initial date, with no default value here. Must be provided, with format `'YYYY-MM-DD'`, to allow proper attribution of the daily data to their months, since daily data need not begin by January the first.

`valm` Monthly value to compute: `1` (sum), `2` (mean, the default), `3` (maximum), or `4` (minimum).

`nmin` Minimum number of available data in a month to compute the monthly value (`15` by default).

`na.strings` Missing data code in the original daily data (defaults to the R standard `"NA"`).

Therefore, we would apply this function to the "homogenized" (probably only with the missing data filled) daily data of our example in this way:

```
dd2m("Tmin", 1956, 2005, ini='2007-01-01')
```
[11]

The output file would be `Tmin-m_1956-2005.dah`. Note that now the variable suffix `-d` has changed to `-m`, to indicate that this file contains monthly data computed from the daily data file, and to avoid overwriting any existing monthly file named `Tmin_1956-2005.dah`. This will be the only output of `dd2m`, and the user may want to manually include this monthly series into a bigger dataset. If needed, the corresponding station coordinates and names may be taken directly from the file `Tmin-d_1956-2005.est`, since the monthly data will be allocated in the same station ordering.

[11] Double or single quotation are equivalent in R.

8. Other climatol functions

This package includes two additional functions that have no relation with homogenization, but provide tools for producing wind-rose and Walter&Lieth climatic graphs. Examples for both functions are provided in the following subsections.

8.1. Wind-rose graphs

This function is called `rosavent`[12], and accepts the following parameters:

`frec` Data frame containing the wind frequencies.

`fnum` Number of reference circles to plot (4 by default).

`fint` Frequency steps (in %) of the reference circles (5 by default).

`flab` Parameter indicating which circles must be labelled: 1 (outer circle only), 2 (all circles, the default), other (do not label any circle).

`ang` Angle along which circles will be labelled (`3*pi/16`).

`col` Colors to fill the frequency polygons (`rainbow(10,.5,.92,start=.33,end=.2)`).

`margen` Margins vector for the plot (to be passed to `par`, see the R help of graphic parameters. (Defaults to `c(0, 0, 4, 0)`).

`key` Set to `FALSE` if you do not want a legend of the wind-rose, that will otherwise be plotted if more than one row of frequencies (speed intervals) are supplied.

`uni` Speed units for the legend header ('m/s').

`...` Other graphic parameters that you may want to set (e.g., the main title of the figure, etc).

Example: Suppose we have the following frequencies in a data frame called `windfr` (that we may have read from a file or computed by other means):

```
     N NNE NE ENE  E ESE SE SSE  S SSW SW WSW  W WNW NW NNW
0-3 59  48 75  90 71  15 10  11 14  20 22  22 24  15 19  33
3-6  3   6 29  42 11   3  4   3  9  50 67  28 14  13 15   5
6-9  1   3 16  17  2   0  0   0  2  16 33  17  6   5  9   2
```

Then, with the following command we would obtain the graphic of figure 17:

```
rosavent(windfr, 4, 4, ang=-3*pi/16, main="Annual windrose")
```

There are no restrictions respect to the number of columns of the data frame, as long as they begin by the North frequencies. (Column headers are disregarded by the function).

[12] A contraction of the Catalan for *wind-rose*.

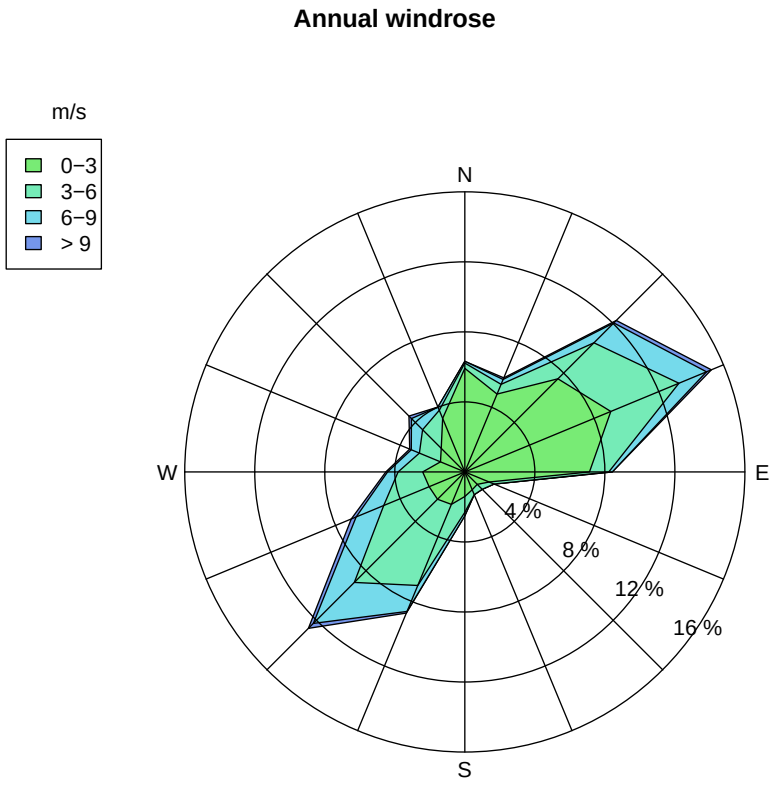

Figure 17: Example of a wind rose obtained with the `rosavent` function.

8.2. Walter&Lieth climograms

Climate diagrams have long been used as a means to synthesize the climate of a place, and the botanists Bagnouls and Gaussen had the idea of plotting the monthly values of temperature and precipitation with a scale 1:2 in order to allow a quick appreciation of the wet months (those for which the precipitation graph goes over that of the temperature). They called this *ombrothermic diagram* (Bagnouls and Gaussen, 1957). Soon after, Walter and Lieth improved it by adding supplementary climatic information, plotting the freeze periods and shrinking the precipitation scale when it surpassed 100 mm to allow a worldwide application of this diagram, no matter how rainy a site can be (Walter and Lieth, 1960).

The function `diagwl` has been provided to generate this kind of climatic diagram from a data frame containing the monthly averages of precipitation and daily maximum, daily minimum and monthly minimum temperatures. The parameters of this function are (default values within parenthesis):

`dat` Monthly climatic data for which the diagram will be plotted.

`est` Name of the climatological station (`""`).

alt Altitude of the climatological station (NA).

per Period on which the averages have been computed ("").

margen Margins vector for the plot (c(4,4,5,4)).

mlab Month labels for the X axis: "en" for English, "es" for Spanish, or otherwise numeric labels will be used ("").

pcol Color pen for precipitation ("#005ac8").

tcol Color pen for temperature ("#e81800").

pfcol Fill color for probable frosts ("#79e6e8").

sfcol Fill color for sure frosts ("#09a0d1").

shem Set to TRUE for southern hemisphere stations (FALSE).

p3line Set to TRUE to draw a supplementary precipitation line referenced to three times the temperature[13] (FALSE).

... Other graphic parameters that you may want to set.

As an example of application, let us suppose that our climatic monthly averages, located in a data frame called datcli, are the following:

```
         Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
Prec.   97.4  69.3  85.5  71.1  48.9  25.1   8.1  37.2  81.6 144.8 110.6 126.5
Max.t.  15.4  16.1  17.2  19.7  23.9  27.9  31.3  31.4  26.5  22.9  18.2  15.8
Min.t.  -0.1  -0.4   1.9   4.9   8.3  11.9  14.8  15.5  13.4   9.7   4.6   2.2
Ab.m.t. -5.1  -7.0  -3.5  -1.7   3.4   8.2  11.6  12.2   9.0   3.0  -1.7  -3.6
```

To generate the climogram of figure 18, we would call the function in this way:

```
diagwl(datcli,est="Example station",alt=100,per="1961-90",mlab="en")
```

You can see the plot of the precipitation and temperature monthly averages, plus annotations of annual averages of both elements (in the upper part) and monthly averages of the daily maximum and minimum temperatures of the warmest and coldest months respectively (at the left margin). Frost likelihood is shown with flat rectangles just under the 0°C axis. When the average daily minimum is zero or negative, we can be sure about the occurrence of frosts, and the rectangle of that month is filled with a darker blue (by default). Otherwise, if it is the absolute monthly miminum that is zero or negative, the rectangle is filled with a lighter blue to indicate the probability of having frosts in that month. The blue vertical pattern depicts the humid months, while the dotted red one shows when the aridity prevails and, in both cases, the corresponding areas give an idea of how intense is the water surplus or shortage.

[13] As suggested by Bogdan Rosca

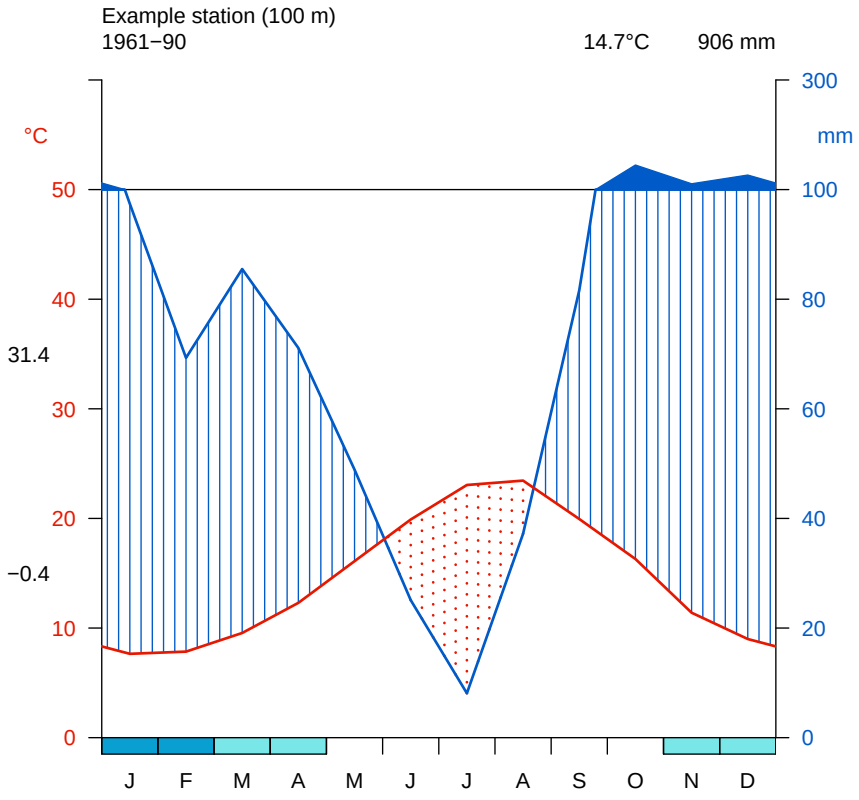

Figure 18: Example of a Walter&Lieth diagram obtained with the `diagwl` function.

9. References

Aguilar E, Auer I, Brunet M, Peterson TC, Wieringa J (2003): *Guidelines on climate metadata and homogenization*. WCDMP-No. 53, WMO-TD No. 1186. World Meteorological Organization, Geneve.

Alexandersson H (1986): A homogeneity test applied to precipitation data. *Jour. of Climatol.*, 6:661-675.

Bagnouls F, Gaussen H (1957): Les climats biologiques et leurs classifications. *Ann. de Geogr.*, 355:193-220.

Bivand RS, Pebesma EJ, Gómez-Rubio V (2008): *Applied Spatial Data Analysis with R*. Springer, 376 pp.

Daget J (1979): *Les modèles mathematiques en écologie*. Collection d'Écologie 8, 172 pp, Masson, Paris.

Khaliq MN, Ouarda TBMJ (2007): On the critical values of the standard normal homogeneity test (SNHT). *Int. J. Climatol.*, 27:681687.

Paulhus JLH, Kohler MA (1952): Interpolation of missing precipitation records. *Month. Weath. Rev.*, 80:129-133.

Peterson TC, Easterling DR, Karl TR, Groisman P, Nicholls N, Plummer N, Torok S, Auer I, Böhm R, Gullett D, Vincent L, Heino R, Tuomenvirta H, Mestre O, Szentimrey T, Salinger J, Førland E, Hanssen-Bauer I, Alexandersson H, Jones P, Parker D (1998): Homogeneity Adjustments of 'In Situ' Atmospheric Climate Data: A Review. *Int. J. Climatol.*, 18:1493-1518.

Sokal RR, Rohlf PJ (1969): *Introduction to Biostatistics*. 2^{nd} edition, 363 pp, W.H. Freeman, New York.

Walter H, Lieth H (1960): *Klimadiagramm Weltatlas*. G. Fischer, Jena.

10. Annex: Threshold values for the SNHT shift detection

Although there are some published critical values of Alexandersson's SNHT test, Monte Carlo simulations were made here especially designed in accordance with the way this test is applied in the climatol package:

White noise series of 600 terms were generated with the R function `rnorm` to simulate the anomaly series of a homogeneous station in a homogeneous network with 50 years of monthly data. For each of this series, shifts of 0.0 (no shift), 0.5, 1.0, 1.5 and 2.0 standard deviation were applied in the middle (at term 301), and the SNHT test was computed for windows of `2*swa` terms and 10 sizes of forward steps of `swa` = 6, 12, 24, 48, 60, 90, 120, 180, 240 and 300 terms. For every SNHT application, the maximum value of the test and the location in the series were recorded, and since 2000 white noise series were simulated, 100000 results were obtained in total.

In the first place, we will analyze the results of the homogeneous series (those to which no shift was imposed). By studying the right tail of the cumulative distribution of the maximum SNHT values, we can get threshold values to avoid false break detection with confidence levels of 90%, 95%, 99%, 99.5% and 99.9%. Figure 19 shows those thresholds, for these confidence levels and for the 10 different `swa` stepped forward windows of `2*swa` term sizes. Randomness is to be blamed for the irregularities of the graphs, but the overall figures are not expected to change significantly if a much bigger number of simulations were performed. It is curious how the plots present a maximum for middle sized windows, since SNHT critical levels reported in the literature show an increase that tend to the horizontal as the sample size grows. But here only for swa=300 is the test applied only once and, the more times the test is computed on the same series in the stepped forward windows scheme, the higher the maximum value reached by the test.

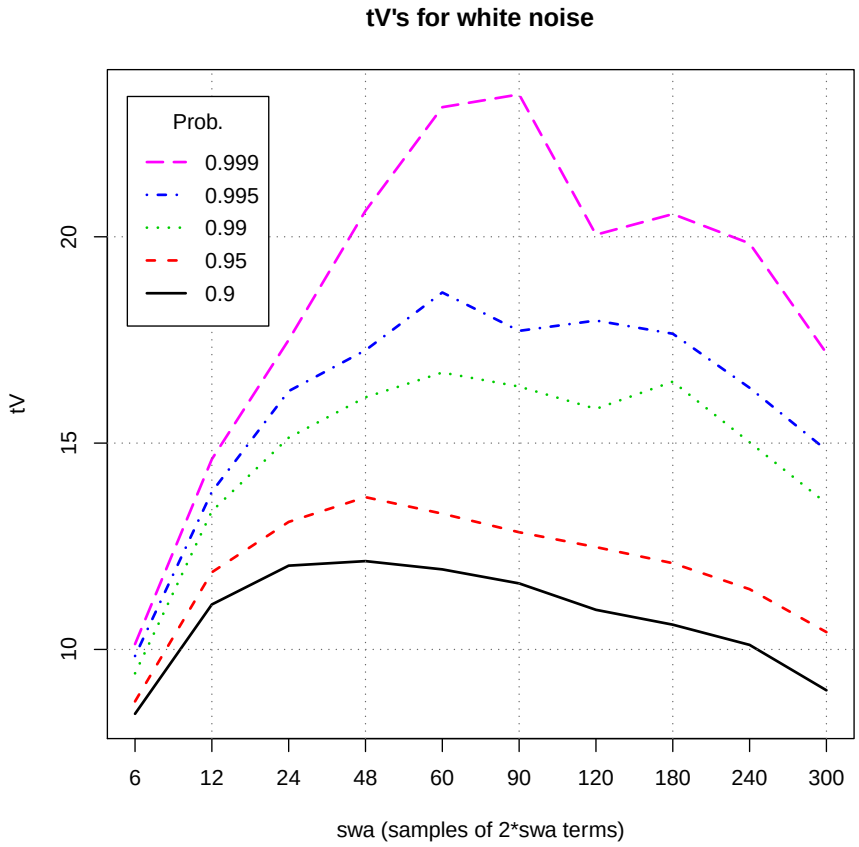

Figure 19: Threshold values (`tV`) of ten `swa` stepped SNHT tests on white noise series for five probabilities of avoiding false break detection.

The next thing we may want to know is how good are these `tV` threshold values in correctly detecting the shifts in the series. To answer this question, the number of `tV`'s higher than the previously obtained thresholds was computed, and they were qualified as correct hits when the location of the shift had an error lower than 12 terms[14], and as false hits otherwise. Both were separately accounted for every one of the 10 `swa` values, 4 shifts and 5 confidence levels.

Figure 20 summarizes the hit rates results, showing that 0.5 standard deviation (sd) shifts are quite difficult to detect, even for the largest sample size, for which the hit rate index lies around 63% for all five confidence levels of avoiding false break detections. The shifts of 1 sd are more reliably detected: 95% when the test is applied to the whole series (`swa=300`), and more than 90% even for samples of 120 terms (`swa=60`), as far as we allow a 10% probability of detecting false breaks (confidence level of 0.90). The higher shifts (of 1.5 and 2.0 sd) are almost totally detected for stepped windows of around 100 terms or more (from `swa=48` onwards).

As for the false breaks (figure 21), with the larger sample sizes the probability of detecting the 0.5 sd breaks at a wrong position is as high as around 35%, while with the smaller sample sizes this probability falls to less than 1% if the test threshold (`tVt`) is set high enough (confidence levels greater than 0.99). In the simulations of 1.0 sd shifts this wrong location probability lies around 5 to 6% in the majority of cases, and for greater shift magnitudes becomes almost negligible (except for the smaller sample sizes in combination with low confidence levels).

[14] A year, if we think of monthly series

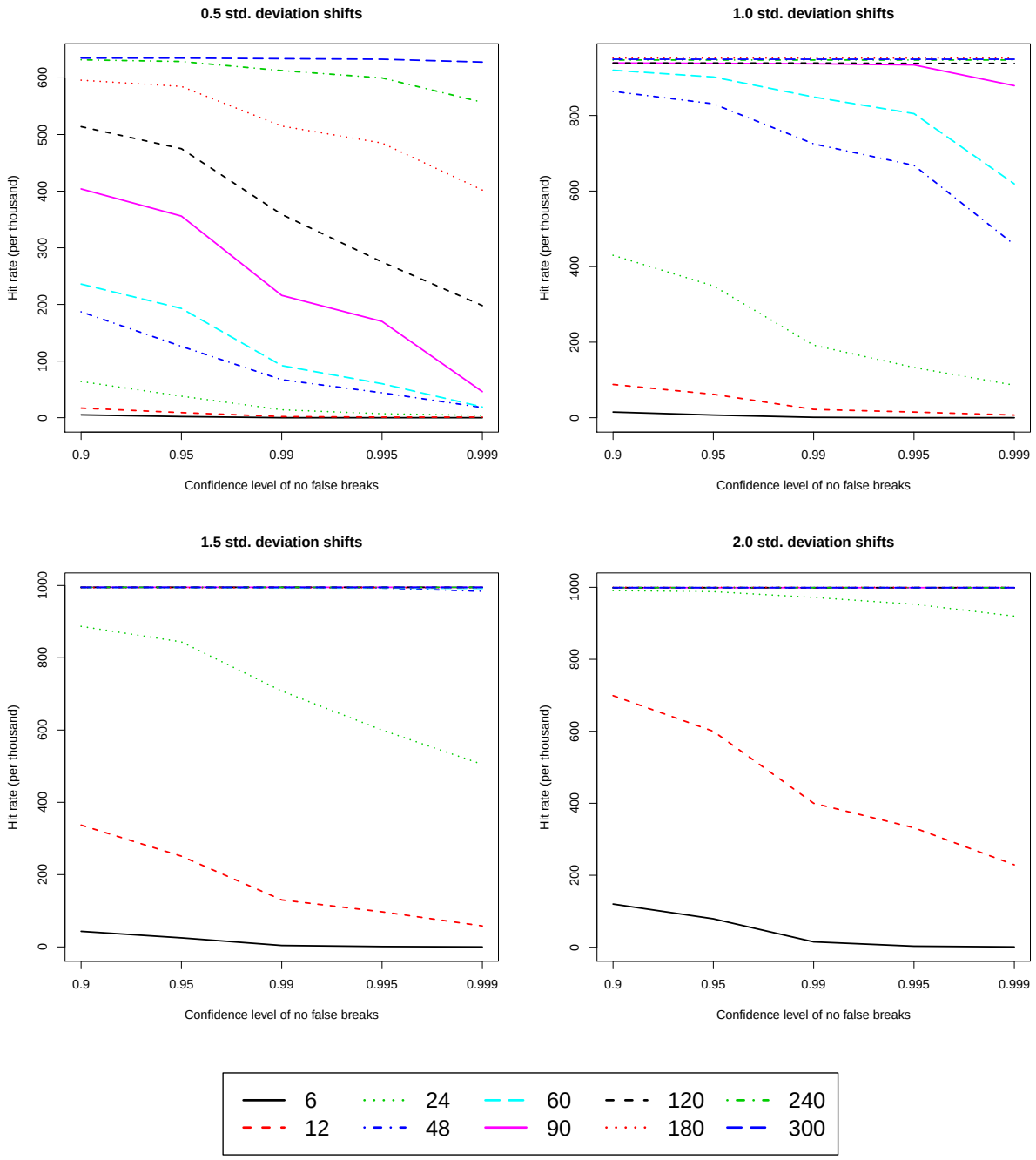

Figure 20: Hit rates for different shift magnitudes, confidence levels of avoiding false breaks, and stepped sample sizes of `2*swa` terms (in different line styles).

Therefore, the greater challenges are met when trying to detect shifts up to 1 standard deviation of the series, because in these situations the lower probability of detection is associated with higher risks of locating the breaks in wrong positions. Figure 22 shows a synthetic "goodness index" computed as the product of the hit rates and the complements of the false hit rates. For 0.5 sd shifts, there is a neat ceiling at 0.4, while for shifts of 1 sd this index reaches 0.9 for the bigger sample sizes.

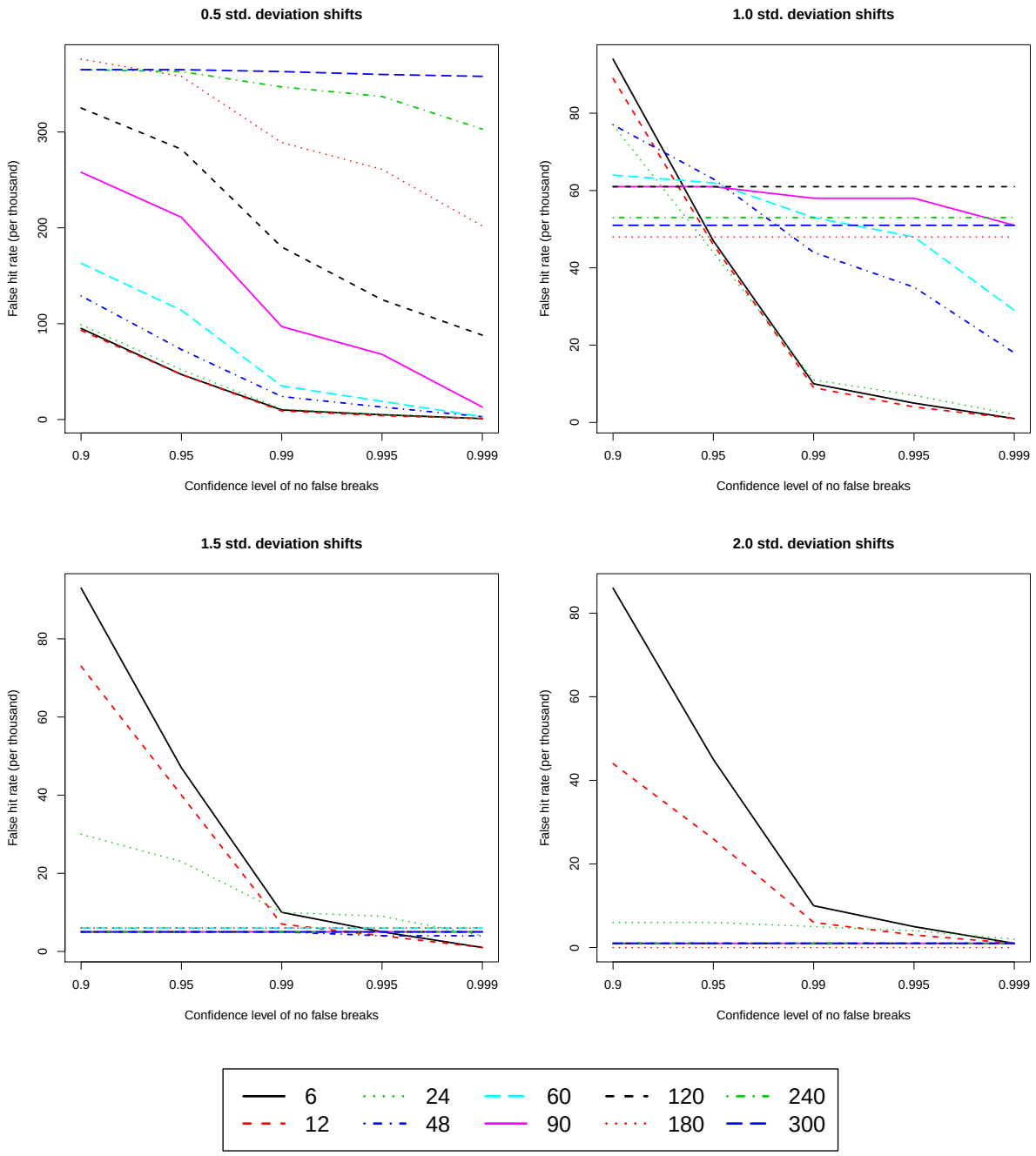

Figure 21: False hit rates for different shift magnitudes, confidence levels of avoiding false breaks, and stepped sample sizes of `2*swa` terms (in different line styles).

The default value of `swa` in the `homogen` function has been set to 60 (dashed cyan line in these figures), as a compromise between a good performance in break detection and power of discrimination when more than one break is present in the series (a common situation). And the SNTH application to the whole series in the second stage of the homogenization process will help in detecting those breaks missed by the stepped window procedure in the first stage.

Important last remark: These `tVt` threshold values have been derived from synthetic white noise series. But in the real world, the anomaly series will unavoidably show some degree of auto-correlation and local or general trends, depending on the type of climatic variable, its

spatial variability, the density of the observing network, and the kind of data (annual, seasonal, monthly, daily, ...). That is why the default values of `tVt` and `snhtt` in the `homogen` function have been set considerably higher than those obtained in these Monte Carlo simulations, and that is why it is advisable to adjust them empirically, with the help of the graphic diagnostic output of a first exploratory application, to tune the performance of the homogenization to your particular needs.

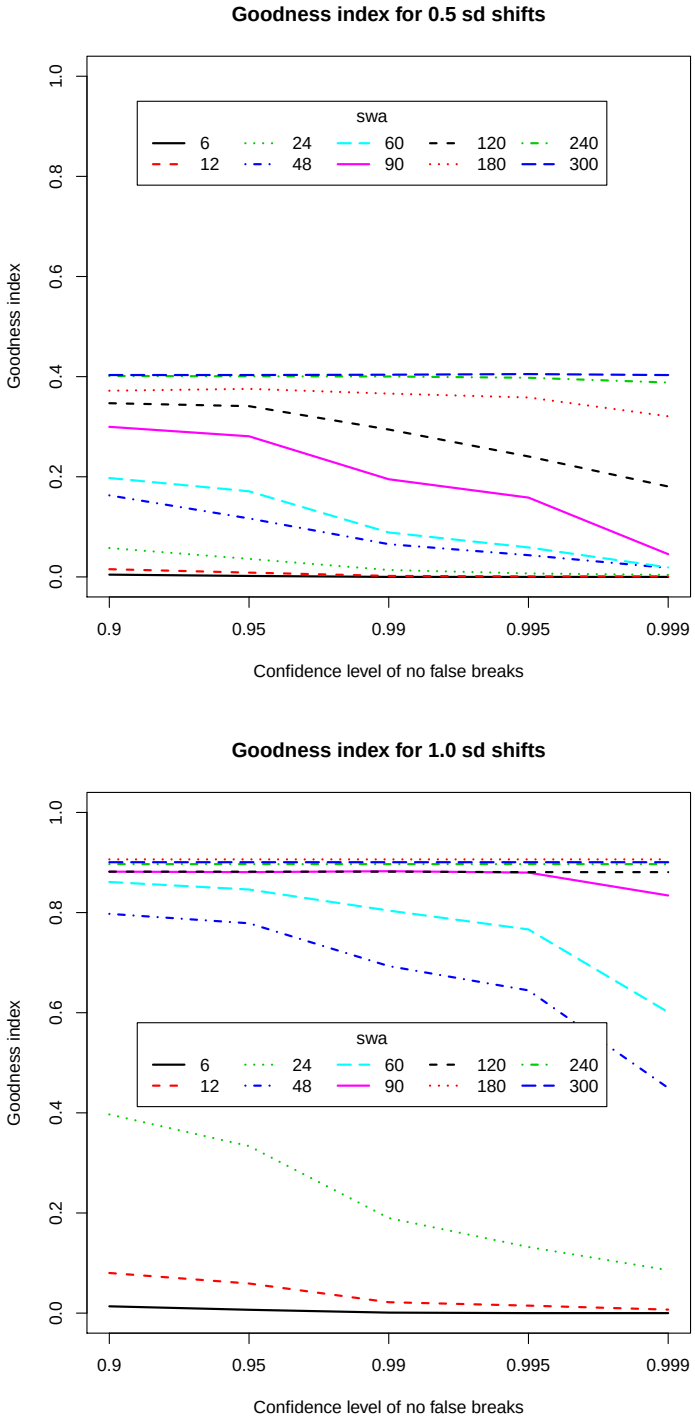

Figure 22: Goodness index for shifts of 0.5 (top) and 1.0 (bottom) standard deviations.