

# Package: FlowScreen (via r-universe)

October 27, 2024

**Title** Daily Streamflow Trend and Change Point Screening

**Version** 1.2.6

**Description** Screens daily streamflow time series for temporal trends and change-points. This package has been primarily developed for assessing the quality of daily streamflow time series. It also contains tools for plotting and calculating many different streamflow metrics. The package can be used to produce summary screening plots showing change-points and significant temporal trends for high flow, low flow, and/or baseflow statistics, or it can be used to perform more detailed hydrological time series analyses. The package was designed for screening daily streamflow time series from Water Survey Canada and the United States Geological Survey but will also work with streamflow time series from many other agencies.

**Depends** R (>= 3.0)

**Imports** zyp, changepoint, evir, graphics, grDevices, stats, utils

**License** GPL (>= 2)

**LazyData** true

**RoxygenNote** 5.0.1

**Encoding** UTF-8

**Maintainer** Jennifer Dierauer <jen.r.brand@gmail.com>

**NeedsCompilation** no

**Author** Jennifer Dierauer [aut, cre], Paul Whitfield [aut]

**Date/Publication** 2019-04-05 16:12:52 UTC

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

**RemoteUrl** <https://github.com/cran/FlowScreen>

**RemoteRef** HEAD

**RemoteSha** 126383e098670dd8b1b30856cfe3df29a0b5ed6e

## Contents

axis_doy.internal . . . . .	3
bf.seas . . . . .	3
bf.stats . . . . .	4
bf_boughton . . . . .	5
bf_eckhardt . . . . .	6
bf_oneparam . . . . .	7
cania.sub.ts . . . . .	7
caniapiscau . . . . .	9
caniapiscau.res . . . . .	10
create.ts . . . . .	10
dr.events . . . . .	11
dr.pds . . . . .	13
dr.seas . . . . .	14
FDC . . . . .	15
FlowScreen . . . . .	16
get.station.internal . . . . .	18
get.titles.internal . . . . .	18
hyear.internal . . . . .	19
MAMn . . . . .	19
metrics.all . . . . .	20
mqt . . . . .	23
NA.runs . . . . .	24
NA.sum . . . . .	25
pk.bf.stats . . . . .	26
pk.cov . . . . .	27
pk.max . . . . .	28
pk.max.doy . . . . .	29
pks . . . . .	29
pks.dur . . . . .	30
Qn . . . . .	31
read.flows . . . . .	32
regime . . . . .	32
screen.cpts . . . . .	33
screen.frames . . . . .	34
screen.frames.internal . . . . .	37
screen.metric . . . . .	37
screen.series . . . . .	39
screen.summary . . . . .	40
screen.summary.internal . . . . .	41
station.info . . . . .	42
YMD.internal . . . . .	43

## Index

44

---

axis_doy.internal	<i>Create custom axis starting on hydrologic year start month</i>
-------------------	---

---

**Description**

Create custom axis starting on hydrologic year start month

**Usage**

```
axis_doy.internal(hyrstart = 10)
```

**Arguments**

hyrstart            numeric indicating month for start of the hydrologic year (water year).

**Author(s)**

Paul Whitfield

---

bf.seas	<i>Seasonal baseflow percentage</i>
---------	-------------------------------------

---

**Description**

This function estimates the percentage of baseflow in a given period relative to the total annual baseflow.

**Usage**

```
bf.seas(TS, seas = c(6:8))
```

**Arguments**

TS                    output from [create.ts](#) containing a data.frame of flow time series  
seas                  Integers representing months of the year. Default is c(6:8), i.e. June-August.

**Details**

This function calls [bf\\_eckhardt](#) to complete the baseflow separation.

**Value**

Returns a vector containing the calculated percentage for each year in the input time series. The "times" attribute provides the corresponding year for each calculated value.

**Author(s)**

Jennifer Dierauer

**See Also**

See [bf.stats](#) to calculate additional baseflow metrics.

**Examples**

```
data(cania.sub.ts)
res <- bf.seas(cania.sub.ts)
res2 <- screen.metric(res, "Percent Annual Baseflow in Jun-Aug")
```

---

bf.stats

*Baseflow statistics*

---

**Description**

This function estimates the baseflow and calculates the mean, max, and min baseflow and baseflow index for a user defined time period.

**Usage**

```
bf.stats(TS, by = "hyear")
```

**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
by	summary period. Options are "year", "hyear", "month", or "doy". Default is "hyear".

**Details**

This function calls [bf\\_eckhardt](#) to complete the baseflow separation.

**Value**

Returns a data.frame with the following columns:

- By - Unique values representing the summary periods, e.g. a list of unique years, months, or days of year
- MeanQ - Mean daily streamflow for the summary period, in m3/s
- MeanBF - Mean daily baseflow for the summary period, in m3/s
- MaxBF - Maximum daily baseflow for the summary period, in m3/s
- MinBF - Minimum daily baseflow for the summary period, in m3/s
- BFVol - Baseflow volume for the summary period, in km3

- MeanBFI - Mean daily baseflow index for the summary period, dimensionless
- MaxBFI - Maximum daily baseflow index for the summary period, dimensionless
- MinBFI - Minimum daily baseflow index for the summary period, dimensionless

**Author(s)**

Jennifer Dierauer

**Examples**

```
data(cania.sub.ts)

res <- bf.stats(cania.sub.ts)
res2 <- screen.metric(res[,2], "m3/s")
```

---

bf\_boughton                      *Boughton recursive digital filter*

---

**Description**

This function estimates baseflow

**Usage**

```
bf_boughton(discharge, k, C)
```

**Arguments**

discharge	Nnumeric vector of daily flow data
k	Numeric value of the recession constant (dimensionless).
C	Numeric value of the partitioning factor (dimensionless).

**Value**

Returns a numeric vector of the estimated baseflow.

**Author(s)**

Paul H. Whitfield

**References**

Boughton, WC. 1993. A hydrograph-based model for estimating the water yield of ungauged catchments. In Hydrology and Water Resources Symposium, Institution of Engineers Australia, Newcastle, NSW; 317-324.

## Examples

```
data(cania.sub.ts)
res <- bf_boughton(cania.sub.ts$Flow, k=0.9, C=0.1)
plot(cania.sub.ts$Date, cania.sub.ts$Flow, xlab="", ylab="Q (m3/s)", type="l")
points(cania.sub.ts$Date, res, type="l", col="blue")
```

---

bf\_eckhardt

*Eckhardt two parameter recursive digital filter*

---

## Description

This function takes vector of discharge data and estimates the baseflow

## Usage

```
bf_eckhardt(discharge, a, BFI)
```

## Arguments

discharge	vector of daily discharge observations
a	Numeric value.
BFI	Numeric value.

## Value

Returns

## Author(s)

Paul Whitfield

## References

Eckhardt, K. 2012. Technical note: Analytical sensitivity analysis of two parameter recursive digital baseflow separation filter. *Hydrology and Earth System Sciences* 16: 451-455.

## Examples

```
data(cania.sub.ts)
bf <- bf_eckhardt(cania.sub.ts$Flow, 0.97, 0.8)
plot(cania.sub.ts$Date, cania.sub.ts$Flow, type="l")
points(cania.sub.ts$Date, bf, type="l", col="blue")
```

---

bf_oneparam	<i>One parameter recursive digital filter</i>
-------------	---

---

**Description**

This function estimates baseflow.

**Usage**

```
bf_oneparam(discharge, k)
```

**Arguments**

discharge	Numeric vector of daily flow data
k	Numeric value for the recession constant (dimensionless).

**Value**

Returns a numeric vector of the estimated baseflow.

**Author(s)**

Paul H. Whitfield

**References**

Eckhardt, K. 2005. How to construct recursive digital filters for baseflow separation methods. *Journal of Hydrology* 352: 168-173.

**Examples**

```
data(cania.sub.ts)
res <- bf_oneparam(cania.sub.ts$Flow, k=0.9)
plot(cania.sub.ts$date, cania.sub.ts$Flow, xlab="", ylab="Q (m3/s)", type="l")
points(cania.sub.ts$date, res, type="l", col="blue")
```

---

cania.sub.ts	<i>Subset of the Caniapiscau River Daily Flows</i>
--------------	--

---

**Description**

This data set includes a subset of the mean daily streamflow for the Caniapiscau Rivers. It includes observations from 1970-1995 (hydrologic years). The code used to subset and modify the original data is shown below.

**Usage**

```
data(caniapiscou)
```

**Format**

Formatted as a data.frame with the following columns:

- ID - Water Survey Canada Station ID
- Date - Date of observation, formatted as YYYY-mm-dd
- Flow - Mean daily streamflow, measured in m3/s
- Code - Data Quality Code
- Agency - Source Agency (Water Survey Canada)
- Year - Calendar year
- month - Calendar month
- doy - Calendar day of year
- hyear - Hydrologic year
- hmonth - Hydrologic month
- hdoy - Hydrologic day of year

**Source**

Environment Canada. 2010. EC Data Explorer V1.2.30.  
Water Survey of Canada V1.2.30 <https://www.ec.gc.ca/rhc-wsc/>

**Examples**

```
# Code used to subset and modify original Caniapiscou series:
## Not run:
data(caniapiscou)
cania.ts <- create.ts(caniapiscou, hrstart=3)
cania.sub.ts <- subset(cania.ts, cania.ts$year %in% c(1970:1995))

## End(Not run)
# example use of example subset flow series
data(cania.sub.ts)
head(cania.sub.ts)
str(cania.sub.ts)
```



---

`caniapiscou`*Caniapiscou River Daily Flows*

---

## Description

This data set includes the mean daily streamflow for the Caniapiscou River. The file has been read from the original .csv format using `read.flows`. The Caniapiscou River is located in Nunavik, Quebec, Canada, and flows northward. The headwaters (representing 45 percent of the total flow) were dammed to create the Caniapiscou Reservoir, which started filling in 1981. In 1985, the reservoir was diverted to the west into the La Grande hydroelectric complex. This flow time series is used as an example of a river with a known change point to demonstrate the package's screening capabilities.

## Usage

```
data(caniapiscou)
```

## Format

Formatted as a `data.frame` with the following columns:

- ID - Water Survey Canada Station ID
- PARAM - Parameter ID (1 indicates flow)
- Date - Date of observation, formatted as YYYY-mm-dd
- Flow - Mean daily streamflow, measured in m<sup>3</sup>/s
- Agency - Source Agency (Water Survey Canada)

## Source

Environment Canada. 2010. EC Data Explorer V1.2.30.  
Water Survey of Canada V1.2.30 <https://www.ec.gc.ca/rhc-wsc/>

## Examples

```
data(caniapiscou)
head(caniapiscou)
str(caniapiscou)
```

---

`caniapiscou.res`*Screening results for the Caniapiscou River*

---

**Description**

Contains the results from `metrics.all` for the full Caniapiscou River daily flow series. Data set created as indicated below. This data set is used in the example documentation for the `screen.frames`, `screen.summary`, and `screen.cpts` functions in order to reduce example run times.

**Usage**

```
data(caniapiscou)
```

**Format**

Formatted as indicated in the documentation for `metrics.all`

**Source**

Original flow series from Environment Canada. 2010. EC Data Explorer V1.2.30. Water Survey of Canada V1.2.30 <https://www.ec.gc.ca/rhc-wsc/>

**Examples**

```
# Code used produce this data set:
## Not run:
data(caniapiscou)
caniapiscou.ts <- create.ts(caniapiscou, hrstart=3)
caniapiscou.ts <- subset(caniapiscou.ts, caniapiscou.ts$year > 1962)
caniapiscou.res <- metrics.all(caniapiscou.ts)

## End(Not run)
# example use of example subset flow series
data(caniapiscou.res)
```

---

`create.ts`*Create a Time Series of daily streamflow observations*

---

**Description**

This function creates a daily time series formatted for use with the functions in this package.

**Usage**

```
create.ts(Flows, hrstart = 10)
```

**Arguments**

Flows	Data.frame containing daily streamflow time series loaded with the <code>read.flows</code> function.
hystart	define start month of hydrologic year. Defaults to 10 (October).

**Value**

Returns a data.frame with year, month, doy, and hyear columns appended to the original input data.frame.

**Author(s)**

Jennifer Dierauer

**Examples**

```
data(caniapiscau)
# subset flow series for shorter example run time
caniapiscau.sub <- caniapiscau[300:1800,]
caniapiscau.sub.ts <- create.ts(caniapiscau.sub)
```

---

dr.events

*Partial Duration Series and Event Statistics for streamflow droughts*

---

**Description**

This function extracts the partial duration series for all streamflow droughts based on a moving window quantile threshold. Also returns summary information (start date, end date, duration, deficit volume) for each drought event.

**Usage**

```
dr.events(TS, Qdr = 0.2, WinSize = 30, IntEventDur = 10, EventDur = 15)
```

**Arguments**

TS	output from <code>create.ts</code> containing a data.frame of flow time series
Qdr	Numeric value of the drought threshold quantile. Default is 0.2.
WinSize	Numeric value specifying the size of the moving window in days. Default is 30.
IntEventDur	Numeric value for the minimum inter-event duration in days. Drought events with less than the specified number of days between will be pooled and considered as one event.
EventDur	Numeric value for the minimum drought duration in days. Default is 15.

**Value**

Returns a list with the following elements:

DroughtEvents: A data.frame with the following columns:

- Event - Integer indicating the original event number assigned before minor drought events were removed.
- Start - Date of the start of the drought event.
- End - Date of the end of the drought event
- maxDef - Numeric value of the maximum streamflow deficit.
- Severity - Numeric value indicating the drought severity, calculated as the cumulative daily streamflow deficit in m<sup>3</sup>/s.
- Duration - Numeric value of the drought duration in days.
- Magnitude - Numeric value indicating the drought magnitude, which is calculated as the mean daily streamflow deficit in m<sup>3</sup>/s.
- stdtotDef - Numeric value indicating the standardized cumulative streamflow deficit, calculated as the drought severity divided by the mean annual daily streamflow.

DroughtPDS: A data.frame of the original input TS that has been subset to include only the days on which the streamflow was below the drought threshold. The data.frame also has the following columns appended:

- Thresh - Numeric value indicating the streamflow drought threshold, as calculated by [mqt](#)
- BelowThresh - Logical indicating whether the observed streamflow was below the streamflow drought threshold.
- Def - Numeric value of the streamflow deficit, calculated as the streamflow drought threshold (m<sup>3</sup>/s) minus the observed streamflow (m<sup>3</sup>/s).

**Author(s)**

Jennifer Dierauer

**See Also**

See [dr.seas](#) to calculate metrics for droughts occurring in a user-defined season.

This function calls [dr.pds](#) which calls [mqt](#).

**Examples**

```
data(cania.sub.ts)
res1 <- dr.events(cania.sub.ts)
events <- res1$DroughtEvents
plot(events$Start, events$Duration, pch=19, ylab="Drought Duration (days)", xlab="")
```

---

`dr.pds`*Get the partial duration series for streamflow droughts*

---

## Description

This function returns the partial duration series for streamflow droughts based on a moving window quantile threshold.

## Usage

```
dr.pds(TS, Qdr = 0.2, WinSize = 30)
```

## Arguments

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
Qdr	Numeric value of the drought threshold quantile. Default is 0.2.
WinSize	Numeric value specifying the size of the moving window in days. Default is 30.

## Details

This function defines a daily streamflow threshold and finds the partial duration series of streamflow droughts. Drought events are identified in the daily streamflow time series with the threshold level approach. In this function, the threshold is defined by a moving quantile, where daily threshold values are based on the 80th percentile of the flow duration curve from a 30-day moving window (Beyene et al. 2014). With this method, every day of the year has a different threshold based on the streamflow measured on the day, the 15 days before the day, and the 15 days after the day. The size of the moving window can be modified with the WinSize argument, and the percentile can be modified with the Qdr argument.

## Value

Returns the input TS data.frame with "Thresh" and "BelowThresh" columns appended. The Thresh column contains the daily flow threshold, and the BelowThresh column is a binary indicating whether the flow on each day was below the drought threshold.

## Author(s)

Jennifer Dierauer

## References

Beyene, B.S., Van Loon, A.F., Van Lanen, H.A.J., Torfs, P.J.J.F., 2014. Investigation of variable threshold level approaches for hydrological drought identification. Hydrol. Earth Syst. Sci. Discuss. 11, 12765-12797. <http://dx.doi.org/10.5194/hessd-11-12765-2014>.

**See Also**

See [create.ts](#) to format the input flow series.

See [mqt](#) to return only the daily moving quantile threshold.

See [dr.events](#) to pool drought events, remove minor events, and calculate metrics.

See [dr.seas](#) to calculate metrics for streamflow droughts that start in a specific month or months.

**Examples**

```
data(cania.sub.ts)
pds <- dr.pds(cania.sub.ts)
pds <- subset(pds, pds$BelowThresh==TRUE)

# plot the flow time series with black and the drought events in red
plot(cania.sub.ts$Date, cania.sub.ts$Flow, ylab="m3/s", xlab="", type="l")
points(pds$Date, pds$Flow, pch=19, cex=0.7, col="red")
```

---

dr.seas

*Find the start, middle, end, and duration of seasonal droughts*


---

**Description**

This function returns the day of year for the start, middle, and end of seasonal droughts. It also returns the duration and severity of each drought event. The function allows for seasonal analysis by defining a season argument which lists months during which droughts of interest may start.

**Usage**

```
dr.seas(TS, Qdr = 0.2, WinSize = 30, IntEventDur = 10, EventDur = 15,
        Season = c(4:9))
```

**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
Qdr	Numeric value for drought quantile. Default is 0.2.
WinSize	Numeric value for moving window size in days. Default is 30.
IntEventDur	Numeric value for the minimum inter-event duration in days. Drought events with less than the specified number of days between will be pooled and considered as one event. Default is 10.
EventDur	Numeric value for the minimum drought duration in days. Default is 15.
Season	Numeric vector of months during which droughts start. Default is c(4:9) for non-frost season droughts.

**Details**

This function calls [dr.events](#) which calls [dr.pds](#) and [mqt](#)

**Value**

Returns a data.frame of drought event metrics; the columns are:

- StartDay - day of year that the drought event started on
- MidDay - day of year for the middle of the drought event, which is defined as the day when the cumulative drought deficit reached 50 total cumulative daily streamflow deficit. Total cumulative streamflow deficit is also referred to as drought severity in this package.
- EndDay - day of year that the drought ended on
- Duration - length of the drought event, in days
- Severity - severity of the drought event, calculated as the total cumulative daily streamflow deficit

The "times" attribute provides the start date to preserve year information and aid in plotting the time series.

**Author(s)**

Jennifer Dierauer

**See Also**

See [create.ts](#) to format the input flow series.  
See [dr.events](#) and [mqt](#) for details on how drought events are defined.

**Examples**

```
data(cania.sub.ts)
res <- dr.seas(cania.sub.ts)
res2 <- screen.metric(res[,1], "Day of Year")
```

---

FDC

*Flow Duration Curve*

---

**Description**

Produces a flow duration curve plot with optional Gustard type-curves that can be used to estimate catchment permeability.

**Usage**

```
FDC(flow, title = NULL, normal = FALSE, gust = TRUE)
```

**Arguments**

flow	daily streamflow time series
title	character string for plot title
normal	boolean indicating whether to plot on normal probability axis (normal=TRUE) or linear probability axis (default, normal=FALSE)
gust	boolean indicating whether to plot Gustard type curves.

**Author(s)**

Paul Whitfield

**References**

Gustard, A., Bullock, A., and Dixon, J.M. (1992). Report No. 108: Low flow estimation in the United Kingdom. Oxfordshire, United Kingdom: Institute of Hydrology.

**Examples**

```
data(caniapiscou)
caniapiscou <- subset(caniapiscou, !is.na(caniapiscou$Flow))
FDC(caniapiscou$Flow, title="Caniapiscou River")
```

FlowScreen

*Screen Daily Discharge Time Series for Temporal Trends and Change Points*

**Description**

This package can be used to calculate more than 30 different streamflow metrics and identify temporal trends and changepoints. It is intended for use as a data quality screening tool aimed at identifying streamflow records that may have anthropogenic impacts or data inhomogeneity.

**Details**

Package: FlowScreen  
 Type: Package  
 Version: 1.2.6  
 Date: 2019-04-05  
 License: GPL (>= 2)

Daily streamflow time series downloaded with the Environment Canada Data Explorer can be loaded with `read.flows`. The `read.flows` function can also be used to load daily streamflow time series from the USGS. The streamflow regime can be visualized with `regime`. A list of 30 streamflow metrics that describe high flows, low flows, and baseflows can be calculated using `metrics.all`. The temporal occurrence of changepoints for all metrics or for only the high flow, baseflow, or low flow metrics can be analyzed using `screen.cpts`. If the streamflow time series has multiple metrics exhibiting changepoints within the same year (or few years), the time series can be further analyzed using `screen.summary` which creates a summary plot showing the significant temporal trends and changepoints for the high flow, low flow, or baseflow metrics. The `screen.metric` can be used to create a time series plot for one metric at a time. The `screen.metric` function works with individual metrics output from the following functions: `pk.max`, `pk.max.doy`, `Qn`, `pk.bf.stats`, `dr.seas`, `MAMn`, `bf.stats`, `pk.cov`, and `bf.seas`. The `screen.frames` function creates individual plots from the `screen.summary` function. The `screen.frames` function can also be used to create custom summary plots, see the example code in the function documentation.



**Author(s)**

Jennifer Dierauer, Paul H. Whitfield

Maintainer: Jennifer Dierauer <jen.r.brand@gmail.com>

**References**

Bard, A., Renard, B., Lang, M. 2011. The AdaptAlp Dataset: Description, guidance, and analyses. In AdaptAlp WP 4 Report, 15. Lyon, France: Cemagraf.

Bard, A., Renard, B., Lang, M., Giuntoli, I., Korck, J., Koboltschnig, G., Janza, M., d'Amico, M., Volken, D. 2015. Trends in the hydrologic regime of Alpine rivers. Journal of Hydrology online.

Svensson, C., Kundzewicz, Z.W., Maurer, T. 2005. Trend detection in river flow series: 2. Flood and low-flow index series. Hydrological Sciences Journal 50:811-824.

Whitfield, P.H. 2012. Why the provenance of data matters: Assessing "Fitness for Purpose" for environmental data. Canadian Water Resources Journal 37:23-36.

Whitfield, P.H. 2013. Is 'Center of Volume' a robust indicator of changes in snowmelt timing? Hydrological Processes 27:2691-2698.

**See Also**

[pot](#), [decluster](#), [cpt.meanvar](#), [zyp.trend.vector](#), [Kendall](#)

**Examples**

```
## Not run:
# load daily streamflow time series for the Caniapiscou River
data(caniapiscou)

# summary plot of the annual flow regime
caniapiscou.ts <- create.ts(caniapiscou)
regime(caniapiscou.ts)

# calculate high flow, low flow, and baseflow metrics
res <- metrics.all(caniapiscou.ts)

# plot histogram of changepoints for high flow, low flow, and baseflow metrics
screen.cpts(res, type="h")
screen.cpts(res, type="l")
screen.cpts(res, type="b")

# or plot all changepoints together
cpts <- screen.cpts(res)

# create screening plots for high, low, and baseflow metrics
screen.summary(res, type="h")
```

```

screen.summary(res, type="l")
screen.summary(res, type="b")

## End(Not run)

```

---

`get.station.internal` *Get station information for USGS or WSC hydrometric stations*

---

**Description**

Get station information for USGS or WSC hydrometric stations

**Usage**

```
get.station.internal(stnID)
```

**Arguments**

`stnID`                    Character string of station ID.

**Value**

Returns a list of station information

**Author(s)**

Jennifer Dierauer

---

`get.titles.internal` *Returns plot titles and labels based on plot type and language preference*

---

**Description**

Returns plot titles and labels based on plot type and language preference

**Usage**

```
get.titles.internal(type, language = "English", Qmax)
```

**Arguments**

`type`                    character indicating the type of summary plot  
`language`                "English" or "French"  
`Qmax`                    the flow quantile used to define peaks of threshold, e.g. 0.95

**Author(s)**

Jennifer Dierauer

---

hyear.internal	<i>Add hydrologic Year, month, and doy columns to a daily time series</i>
----------------	---

---

**Description**

Add hydrologic Year, month, and doy columns to a daily time series

**Usage**

```
hyear.internal(TS, hystart = 10)
```

**Arguments**

TS	Output from <a href="#">create.ts</a> function.
hystart	define start month of hydrologic year. Defaults to 10 (October).

**Value**

Returns a data.frame with hyear, hmonth, and hdoy columns appended to the original input data.frame.

**Author(s)**

Jennifer Dierauer

---

MAMn	<i>Calculate mean annual minimum n-day flows</i>
------	--

---

**Description**

This function calculates the mean annual minimum n-day flow by calendar year or by hydrologic year. This function can also be used to find the annual minimum series by setting n=1.

**Usage**

```
MAMn(TS, n = 7, by = "hyear")
```

**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
n	Numeric value for the number of days in the n-day flow period. Default is 7.
by	Character string indicating whether to use hydrologic years or calendar years. Default is "hyear". Other option is "year".

**Value**

Returns a numeric vector containing the calculated MAM n-day flow for each year in the input time series. The "times" attribute provides the corresponding year for each calculated value.

**Author(s)**

Jennifer Dierauer

**See Also**

[screen.metric](#)

**Examples**

```
data(cania.sub.ts)

# find the annual minimum series and plot
res <- MAMn(cania.sub.ts, n=1)
res2 <- screen.metric(res, "Q (m3/s)")

# do the same with MAM 7-day flow instead of annual minimum
res <- MAMn(cania.sub.ts, n=7)
res2 <- screen.metric(res, "Q (m3/s)")
```

---

metrics.all

*Streamflow metrics*

---

**Description**

Calculates 30 different flow metrics, 10 each for high flows, low flows, and baseflow.

**Usage**

```
metrics.all(TS, Qmax = 0.95, Dur = 5, Qdr = 0.2, WinSize = 30,
  Season = c(4:9), NATHresh = 0.5, language = "English")
```

**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
Qmax	Numeric value for peaks over threshold quantile. Default is 0.95.
Dur	Numeric value for minimum number of days between flood peaks. Default is 5.
Qdr	Numeric value for drought quantile. Default is 0.2, i.e. the 80th percentile of the flow duration curve.
WinSize	Numeric value for moving window size (in days) for the moving window quantile drought threshold. See <a href="#">mqt</a> . Default is 30.
Season	Numeric vector of months during which droughts start. Default is c(4:9) for non-frost season droughts.

NAtresh	Numeric value indicating the threshold for missing data points in any one year. Default is 0.5, indicating that years with more than 50 percent missing data will be omitted from the metric calculations. This value should always be set to greater than 0.1, as years with fewer observations than approximately 1 month will cause errors.
language	Character string indicating the language to be used for naming the different plot metrics. These names are used in <a href="#">screen.summary</a> to label individual plots. Options are "English" or "French". Default is "English".

## Details

This function calculates streamflow metrics and calculates the prewhitened trend using [zyp.trend.vector](#) and looks for changpoints in mean and variance using [cpt.meanvar](#). This function is intended for use as a data quality screening tool aimed at identifying streamflow records with anthropogenic impacts and should not be used to complete a temporal trend analysis, as the calculated metrics may not be appropriate for all catchments. See the functions linked in the following section for details on how each metric is calculated.

## Value

Returns a list with the following elements:

metricTS: a list containing a vector of each metric calculated. Each vector has a times attribute providing either the year for metrics with one observation per year or a date for metrics that may have more than one observation per year (e.g., Peaks Over Threshold). This list has the following elements:

- Annual Maximum Series - calculated with [pk.max](#)
- Day of Annual Maximum - calculated with [pk.max.doy](#)
- Peaks Over Threshold (Qmax) - calculated with [pks](#)
- Inter-Event Duration - calculated with [pks.dur](#)
- Q80 - calculated with [Qn](#)
- Q90 - calculated with [Qn](#)
- Day of Year 25 percent Annual Volume - calculated with [pk.cov](#)
- Center of Volume - calculated with [pk.cov](#)
- Day of Year 75 percent Annual Volume - calculated with [pk.cov](#)
- Duration between 25 percent and 75 percent Annual Volume - calculated with [cov](#)
- Q10 - calculated with [Qn](#)
- Q25 - calculated with [Qn](#)
- Drought Start - calculated with [dr.seas](#)
- Drought Center - calculated with [dr.seas](#)
- Drought End - calculated with [dr.seas](#)
- Drought Duration - calculated with [dr.seas](#)
- Drought Severity - calculated with [dr.seas](#)

- Annual Minimum Flow - calculated with [MAMn](#)
- Mean Annual Minimum 7-day Flow - calculated with [MAMn](#)
- Mean Annual Minimum 10-day Flow - calculated with [MAMn](#)
- Mean Daily Discharge - calculated with [bf.stats](#)
- Annual Baseflow Volume - calculated with [bf.stats](#)
- Annual Mean Baseflow - calculated with [bf.stats](#)
- Annual Maximum Baseflow - calculated with [bf.stats](#)
- Annual Minimum Baseflow - calculated with [bf.stats](#)
- Mean Annual Baseflow Index - calculated with [bf.stats](#)
- Day of Year 25 percent Baseflow Volume - calculated with [pk.bf.stats](#)
- Center of Volume Baseflow - calculated with [pk.bf.stats](#)
- Day of Year 75 percent Baseflow Volume - calculated with [pk.bf.stats](#)
- Duration between 25 percent and 75 percent Baseflow Volume - calculated with [pk.bf.stats](#)

tcpRes: this list contains the results of the trend and changepoint analysis for each of the metrics in the metricTS list described above. Each list element is a list containing the following elements:

- MetricID - integer used to identify the metric
- MetricName - Name of the metric.
- Slope - numeric vector containing the intercept and slope of the prewhitened linear trend calculated using the Yue Pilon method. See [zyp.trend.vector](#)
- ci1 - upper bound of the trend's 95 percent confidence interval
- ci2 - lower bound of the trend's 95 percent confidence interval
- pval - Kendall's P-value computed for the detrended time series
- cpts - Most probable location of a changepoint, if one is detected.
- means - Mean before and after the changepoint
- NumObs - The number of data points for the metric

inData: A data.frame of the original input daily streamflow time series.

OmitYrs: A data.frame containing the years and the number of observations for any years omitted from the analysis due to insufficient data. If no years were omitted, NA is returned.

### Author(s)

Jennifer Dierauer

### See Also

See the documentation for individual functions linked in the output description for a details on methods.

See [screen.metric](#) to create individual plots for each metric.

## Examples

```
# load subset of daily streamflow time series for the Caniapiscou River
data(cania.sub.ts)

## Not run:
# calculate low flow, high flow, and baseflow metrics
res <- metrics.all(cania.sub.ts)

## End(Not run)
```

---

mqt	<i>Moving quantile threshold</i>
-----	----------------------------------

---

## Description

This function calculates the daily moving window quantile threshold for use in identifying the partial duration series of streamflow droughts.

## Usage

```
mqt(TS, Qdr = 0.2, WinSize = 30)
```

## Arguments

TS	output from <code>create.ts</code> containing a data.frame of flow time series
Qdr	Numeric value of the drought threshold quantile. Default is 0.2.
WinSize	Numeric value specifying the size of the moving window in days. Default is 30.

## Details

The threshold is defined by a moving quantile, where daily threshold values are based on the 80th percentile of the flow duration curve (i.e. 0.2 quantile) from a 30-day moving window (Beyene et al. 2014). With this method, every day of the year has a different threshold based on the streamflow measured on the day, the 15 days before the day, and the 15 days after the day. The size of the moving window can be modified with the `WinSize` argument, and the percentile can be modified with the `Qdr` argument.

## Value

Returns a numeric vector containing the streamflow drought threshold in m<sup>3</sup>/s for each day of the year.

## Author(s)

Jennifer Dierauer

## References

Beyene, B.S., Van Loon, A.F., Van Lanen, H.A.J., Torfs, P.J.J.F., 2014. Investigation of variable threshold level approaches for hydrological drought identification. *Hydrol. Earth Syst. Sci. Discuss.* 11, 12765-12797. <http://dx.doi.org/10.5194/hessd-11-12765-2014>.

## See Also

See [create.ts](#) to format the input flow series.

The following functions use this function: [dr.pds](#), [dr.events](#), [dr.seas](#)

## Examples

```
data(cania.sub.ts)
res <- mqt(cania.sub.ts)

# subset one year of the flow series
flow.sub <- cania.sub.ts[cania.sub.ts$year == 1990,]

# plot the 1990 observed flows in dark blue and the daily drought threshold in red
plot(flow.sub$doy, flow.sub$Flow, ylab="Q (m3/s)", xlab="Day of Year",
      pch=19, col="darkblue", type="b")
points(res, pch=19, cex=0.7, col="red")
```

---

NA.runs

*Missing data runs for daily time series.*

---

## Description

This function takes a data.frame from [create.ts](#) and returns a data.frame of missing data runs.

## Usage

```
NA.runs(TS)
```

## Arguments

TS                    output from [create.ts](#) containing a data.frame of flow time series

## Value

Returns a data.frame with the following columns:

- Start - Date of the start of the missing data period
- End - Date of the end of the missing data period
- Duration - number of days in the missing data period



**Author(s)**

Jennifer Dierauer

**See Also**[create.ts](#) to create input, [NA.sum](#) to sum the the missing data occurrences by year or month.**Examples**

```
data(caniapiscau)
cania.sub <- caniapiscau[300:1200,]
cania.ts <- create.ts(cania.sub)
res <- NA.runs(cania.ts)
```

---

**NA.sum***Sum missing data points from a daily time series*

---

**Description**

Counts the number of missing data points by calendar year, hydrologic year, or month

**Usage**

```
NA.sum(input, by = "hyear", hystart = 1)
```

**Arguments**

input	output from <a href="#">NA.runs</a>
by	character string identifying the time period to summarize by. Defaults is hydrologic year ("hyear"), other choices are "year" and "month". The "month" option will return the number of missing data points for each month in the time series.
hystart	optional argument, define start month of hydrologic year

**Value**

Returns a numeric vector of the number of missing observations per summary period. The "times" attribute of the returned vector provides the corresponding year, hyear, or month.

**Author(s)**

Jennifer Dierauer

**See Also**[NA.runs](#)

## Examples

```
data(caniapiscou)
cania.sub <- caniapiscou[300:1200,]
cania.ts <- create.ts(cania.sub)
res <- NA.runs(cania.ts)
res2 <- NA.sum(res)
```

---

pk.bf.stats

*Calculate baseflow peak statistics*

---

## Description

This function finds the start, middle, end, and duration of the baseflow peak based on percent of the total annual baseflow volume. A value of 0 is returned for years with no flow. Hydrologic years with fewer than normal observations (outliers) are excluded from the analysis, and for stations with seasonal flow records, additional seasonal subsetting is done to include only days with observations in all years.

## Usage

```
pk.bf.stats(TS, bfpct = c(25, 50, 75))
```

## Arguments

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
bfpct	numeric vector of percentages used to define the start, middle, and end of the baseflow peak. Default is <code>c(25, 50, 75)</code>

## Details

This function calculates metrics intended to focus on snowmelt-related streamflow occurring in spring and summer. For catchments in cold climates, the baseflow peak can be interpreted as snowmelt-induced. Baseflow is estimated with [bf\\_eckhardt](#). If total annual flow is equal to 0, returns NA for that year.

## Value

Returns a data.frame with the following columns:

- Start - day of year defining the start of the baseflow peak
- Mid - day of year defining the middle of the baseflow peak
- End - day of year defining the end of the baseflow peak
- Dur - duration of the baseflow peak, in days

## Author(s)

Jennifer Dierauer

## Examples

```
data(cania.sub.ts)
res1 <- pk.bf.stats(cania.sub.ts)

# trend and changepoint plot for baseflow peak start doy
res2 <- screen.metric(res1[,1], "Day of Year")
```

---

pk.cov

*Center of Volume*

---

## Description

This function calculates center of volume metrics, including the day of the hydrologic year that 25 percent, 50 percent, and 75 percent of the total annual streamflow is reached. A value of 0 is returned for years with no flow. Hydrologic years with fewer than normal observations (outliers) are excluded from the analysis, and for stations with seasonal flow records, additional seasonal subsetting is done to include only days with observations in all years.

## Usage

```
pk.cov(TS)
```

## Arguments

TS                      output from [create.ts](#) containing a data.frame of flow time series

## Value

Returns a data.frame with the following columns:

- hYear - Hydrologic Years
- Q25 - day of hydrologic year for 25 percent of the total annual streamflow
- Q50 - day of hydrologic year for 50 percent of the total annual streamflow, i.e. Center of Volume
- Q75 - day of hydrologic year for 75 percent of the total annual streamflow
- Dur - duration of between the 25 percent and 75 percent day of year, in days

## Author(s)

Jennifer Dierauer

## Examples

```
data(cania.sub.ts)
res1 <- pk.cov(cania.sub.ts)

# trend and changepoint plot for baseflow peak start doy
res2 <- screen.metric(res1[,2], "Day of Year")
```

---

pk.max	<i>Annual maximum series</i>
--------	------------------------------

---

### Description

This function returns the annual maximum series from a daily streamflow time series.

### Usage

```
pk.max(TS)
```

### Arguments

TS                    output from [create.ts](#) containing a data.frame of the daily streamflow time series

### Value

Returns a numeric vector containing the annual maximum flow (m<sup>3</sup>/s) series, by hydrologic year. The "times" attribute contains the hydrologic year for each element in the vector.

### Author(s)

Jennifer Dierauer

### See Also

See [create.ts](#) to format the input flow series.

See [pk.max.doy](#) to find the day of year for each annual maximum flow event.

### Examples

```
data(cania.sub.ts)
res <- pk.max(cania.sub.ts)
res2 <- screen.metric(res, "Q (m3/s)")
```

---

pk.max.doy	<i>Day of year for annual maximum series</i>
------------	--

---

**Description**

This function returns the day of the hydrologic year for each annual maximum flow.

**Usage**

```
pk.max.doy(TS)
```

**Arguments**

TS                    output from [create.ts](#) containing a data.frame of flow time series

**Value**

Returns a numeric vector containing the day of the (hydrologic) year for each annual maximum flow. The "times" attribute contains the hydrologic year for each element in the vector.

**Author(s)**

Jennifer Dierauer

**See Also**

See [create.ts](#) to format the input flow series.  
See [pk.max](#) for the annual maximum flow series.

**Examples**

```
data(cania.sub.ts)  
res <- pk.max.doy(cania.sub.ts)  
res2 <- screen.metric(res, "Day of Year")
```

---

pks	<i>Get the flow peaks over a threshold</i>
-----	--

---

**Description**

This function finds the flow peaks over a user defined threshold and declusters to remove dependent peaks.

**Usage**

```
pks(TS, Dur = 5, Qmax = 0.95)
```

### Arguments

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
Dur	numeric value of the minimum number of days between peaks
Qmax	numeric value for peaks over threshold quantile. Default is 0.95.

### Details

Peaks Over Threshold values are calculated as mean daily streamflow (m3/s) minus the threshold streamflow value (m3/s) defined by the input quantile (Qmax). Peaks are identified with [pot](#) and the minimum inter-event duration (Dur) is applied by [decluster](#).

### Value

Returns a numeric vector of peaks of threshold values in m3/s. The "times" attribute contains the date by calendar year, and the "names" attribute contains the hydrologic year and hydrologic day of year, e.g., 2012 55.

### Author(s)

Jennifer Dierauer

### Examples

```
data(cania.sub.ts)
res <- pks(cania.sub.ts)
res2 <- screen.metric(res, "Peak Over Threshold (m3/s)")
```

---

pks.dur	<i>Calculate the inter-event duration</i>
---------	---

---

### Description

This function calculates duration (in days) between flow peaks.

### Usage

```
pks.dur(Peaks)
```

### Arguments

Peaks	Output from <a href="#">pks</a> .
-------	-----------------------------------

### Value

Returns a numeric vector containing the duration (in days) between peaks over threshold from [pks](#). The "times" attribute contains the calendar year date of the earlier peak. The "names" attribute contains the hydrologic year and the day (1-366) of the hydrologic year.

**Author(s)**

Jennifer Dierauer

**Examples**

```
data(cania.sub.ts)
res1 <- pks(cania.sub.ts)
res2 <- pks.dur(res1)
res3 <- screen.metric(res2, "Inter-Event Duration (days)")
```

---

Qn *Calculate flow quantiles*

---

**Description**

This function calculates flow quantiles by hydrologic year, calendar year, month, or doy.

**Usage**

```
Qn(TS, n = 0.1, by = "hyear")
```

**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
n	Numeric value of the quantile. Default is 0.1.
by	Character string indicating time unit to summarize by. Default is "hyear" for hydrologic year, see <a href="#">create.ts</a> . Other options are "year" for calendar year, "month", or "doy" for day of year.

**Value**

Returns a numeric vector of the calculated flow quantile for the time periods indicated with the "by" argument. The "times" attribute contains the hydrologic year, calendar year, month, or day of year for each data point.

**Author(s)**

Jennifer Dierauer

**Examples**

```
data(cania.sub.ts)

# 50% quantile, i.e. mean, by calendar year
res <- Qn(cania.sub.ts, n=0.5, by="year")
res2 <- screen.metric(res, "Q (m3/s)")

# Default 10% quantile, by hydrologic year
```

```
res <- Qn(cania.sub.ts)
res2 <- screen.metric(res, "Q (m3/s)")
```

---

read.flows	<i>Read .csv or .Rdata file of streamflows</i>
------------	--

---

### Description

Reads .csv or .Rdata files of daily streamflow time series. Recognizes several formats, including those used by Water Survey Canada and the United States Geological Survey. Uses read.csv() or load() functions from base package, returns data frame with ID, Date, and Flow, and, if available, associated quality codes and source agency. Replaces negative values that are sometimes used to denote missing data with NAs.

### Usage

```
read.flows(filename, convert = FALSE)
```

### Arguments

filename	name of .csv file to be read.
convert	Boolean indicating whether or not to convert USGS flow values from cubic feet per second to cubic meters per second. (Axes are automatically labeled in cubic meters per second)

### Author(s)

Jennifer Dierauer

---

regime	<i>Plot flow regime</i>
--------	-------------------------

---

### Description

This function plots the min, max, mean, and two user-defined quantiles of daily streamflow to provide visual summary of the flow regime. Area between the upper and lower quantile is shaded grey, the dark blue line represents the mean daily discharge, gray line represents the median daily discharge, and the period of record daily maximum and minimum are shown with the blue points.

### Usage

```
regime(TS, q = c(0.9, 0.1), text = "d", by = "hdoy", y.lims = NA)
```



**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
q	Numeric vector of the upper and lower quantile values. Default is c(0.9, 0.1).
text	optional character string for margin text, e.g. for station name, location, or other notes. Set to NULL if no margin text is wanted, or set to "d" to use default text containing the station ID, station name, and province/state returned from <a href="#">station.info</a> .
by	Character string indicating whether to plot the regime by day of the hydrologic year (defined using <a href="#">create.ts</a> ) or by day of the calendar year. Options are "doy" (calendar year) or "hdoy" (default, hydrologic year).
y.lims	optional user-defined y-axis minimum and maximum. e.g. c(0, 500)

**Author(s)**

Jennifer Dierauer

**Examples**

```
# plot the flow regime of the Caniapiscou River
data(cania.sub.ts)
regime(cania.sub.ts)
```

---

screen.cpts

*Change point time series plot*

---

**Description**

Compiles change point information for all metrics and outputs a daily flow time series plot overlain with a bar plot of changepoint counts by year.

**Usage**

```
screen.cpts(metrics, type = "a", text = NULL)
```

**Arguments**

metrics	output from <a href="#">metrics.all</a>
type	character indicating which type of metric to compile change points for. Options are "h" for high flow metrics, "l" for low flow metrics, "b" for baseflow metrics, or "a" for all 30 metrics (10 high, 10 low, 10 baseflow).
text	optional character string for margin text, e.g. for station name, location, or other notes. Set to NULL if no margin text is wanted, or set to "d" to use default text containing the station ID, station name, and province/state returned from <a href="#">station.info</a> .

**Value**

When type="a", returns a data.frame of changepoint counts by metric type and year.

**Author(s)**

Jennifer Dierauer

**See Also**

[metrics.all](#)

**Examples**

```
# load results from metrics.all function for the Caniapiscou River
data(caniapiscou.res)

# plot changepoints for all metrics
screen.cpts(caniapiscou.res, type="l")
```

---

screen.frames

*Plot one or more frames from the summary screening plot*

---

**Description**

This function plots one or more frames (i.e. time series plot) from any of the three plot.screening summary plots at a time. It can be used to create custom summary plots - see the example code.

**Usage**

```
screen.frames(metrics, type = "h", element = NULL, language = "English",
  StnInfo = NULL, mmar = c(3, 4, 0.5, 0.5), text = "d", multi = F,
  xaxis = T)
```

**Arguments**

metrics	output from <a href="#">metrics.all</a>
type	Character string indicating the set of metrics to plot. Options are "h" for high flow metrics, "l" for low flow metrics, or "b" for baseflow metrics.
element	Numeric index(es) (1-10) of the frame(s) to plot, see details of this function for the list of metrics for each category (high, low, baseflow). Each category has ten different metrics that can be plotted individually. Default is NULL, which creates individual plots for all ten metrics. A list of elements c(1, 5, 10) can be specified or a range c(1:5).
language	Language for plot labels. Choice of either "English" or "French". Default is "English".

StnInfo	Optional data.frame containing user-supplied station info for plot. data.frame must have 7 columns containing station info in the following order: Station ID, Station Name, Prov/State, Country, Latitude, Longitude, Catchment Area. If any of the information is unavailable, fill with NA. The Station ID column must match the Station ID in column 1 of the data.frame input from <a href="#">create.ts</a> .
mmar	Numeric vector specifying plot margins. Default is <code>c(3,4,0.5,0.5)</code>
text	Character string containing text for margin. This can be set to NULL if no margin text is wanted, or set to "d" to use default text containing the station ID, station name, and the prov/state output from <a href="#">station.info</a> . Set to NULL to use this function in a multi-plot layout.
multi	Boolean indicating whether the function is being used to create one plot in a multi-plot layout. Default is F. If T, suppresses the reset of plot parameter settings. This plot function will only work for a multi-plot layout if <code>text=F</code>
xaxis	Boolean indicating whether to plot an x-axis. Default = T.

### Details

High flow metrics include:

1. Annual Maximum Series
2. Annual Maximum Day of Year
3. Peaks Over Threshold (Qmax)
4. Inter-Event Duration
5. Q80
6. Q90
7. Day of Year 25 percent Annual Flow
8. Center of Volume
9. Day of Year 75 percent Annual Flow
10. Duration between 25 percent and 75 percent Annual Flow

Low flow metrics include:

1. Q10
2. Q25
3. Drought Start
4. Drought Center
5. Drought End
6. Drought Duration
7. Drought Severity
8. Annual Minimum Flow
9. Mean Annual Minimum 7-day Flow
10. Mean Annual Minimum 10-day Flow

Baseflow metrics include:

1. Mean Daily Discharge
2. Annual Baseflow Volume
3. Annual Mean Baseflow
4. Annual Maximum Baseflow
5. Annual Minimum Baseflow
6. Mean Annual Baseflow Index
7. Day of Year 25 percent Baseflow Volume
8. Center of Volume Baseflow
9. Day of Year 75 percent Baseflow Volume
10. Duration between 25 percent and 75 percent Baseflow Volume

### Author(s)

Jennifer Dierauer and Paul Whitfield

### Examples

```
# load results from metrics.all function for the Caniapiscau River
data(caniapiscau.res)
caniapiscau.ts <- caniapiscau.res$indata

# plot one frame from the baseflow screening plot
screen.frames(caniapiscau.res, type="b", element=1)

# plot three frames from the low flow screening plot
screen.frames(caniapiscau.res, type="l", element=c(1:3))

# create a custom summary plot
opar <- par(no.readonly = TRUE)
layout(matrix(c(1,2,3,4), 2, 2, byrow=TRUE))
par(oma=c(0,0,3,0))
stninfo <- station.info(caniapiscau.ts, Plot=TRUE)
screen.frames(caniapiscau.res, type="h", element=1, text=NULL, multi=TRUE)
screen.frames(caniapiscau.res, type="l", element=1, text=NULL, multi=TRUE)
screen.frames(caniapiscau.res, type="b", element=1, text=NULL, multi=TRUE)
mtext(paste("Station ID: ", caniapiscau.ts[1,1], ", Agency: WSC, Country: CA", sep=""),
side=3, line=1, outer=TRUE, cex=0.9)
par <- opar
layout(1,1,1)

# or plot everything!
opar <- par(no.readonly = TRUE)
layout(matrix(c(1:30), 5, 6, byrow=TRUE))
screen.frames(caniapiscau.res, type="h", text=NULL, multi=TRUE)
screen.frames(caniapiscau.res, type="l", text=NULL, multi=TRUE)
screen.frames(caniapiscau.res, type="b", text=NULL, multi=TRUE)
par <- opar
layout(1,1,1)
```

---

`screen.frames.internal`*Internal wrapper for creating trend and change-point plots*

---

**Description**

Internal wrapper for creating trend and change-point plots

**Usage**

```
screen.frames.internal(input, mparam, mylab, DataType, maf, mmar, text, xaxis,
    Year1, YearEnd, hydrstart)
```

**Arguments**

<code>input</code>	metric time series
<code>mparam</code>	trend and change point info
<code>mylab</code>	y axis label
<code>DataType</code>	numeric indicating data type
<code>maf</code>	mean annual flow series
<code>mmar</code>	plot margins
<code>text</code>	boolean indicating whether to add text
<code>xaxis</code>	boolean indicating whether to plot the x axis
<code>Year1</code>	start year of original time series
<code>YearEnd</code>	end year of original time series
<code>hydrstart</code>	numeric indicating month for start of the hydrologic year

**Author(s)**

Jennifer Dierauer

---

`screen.metric`*Plot a metric with trend and change points*

---

**Description**

This function plots a time series of a streamflow metric with the prewhitened linear trend and any detected change-points in mean and variance.

**Usage**

```
screen.metric(y, ylabel = "", text = NULL)
```

## Arguments

y	Numeric vector with "times" attribute
ylabel	Character string for the y-axis label
text	optional character string for margin text, e.g. for station name, location, or other notes.

## Details

This function plots detected changepoints as a vertical dashed line. The means on either side of a changepoint are plotted as solid black lines. If the temporal trend is significant (p-value < 0.1), the trend is plotted as a blue or red line for an increasing or decreasing trend, respectively. The upper and lower 95 dotted red or blue lines. If a trend is not significant, it is not plotted.

## Value

Returns a list containing results from the trend and changepoint analysis. This list has the following elements:

- slope - Numeric vector containing the intercept and slope of the prewhitened linear trend computed with [zyp.trend.vector](#) using Yue Pilon's method
- ci1 - numeric vector containing the intercept and slope of the upper confidence bound. See [confint.zyp](#)
- ci2 - numeric vector of length 2 containing the intercept and slope of the lower confidence bound. See [confint.zyp](#)
- pval - numeric value indicating the significance value of the detected trend, Kendall test computed within [zyp.trend.vector](#)
- cpts - numeric vector of changepoints if any are found, computed with [cpt.meanvar](#)
- means - numeric vector of means computed with [cpt.meanvar](#)

## Author(s)

Jennifer Dierauer

## See Also

See [screen.summary](#) to create a summary screening plot of high flow, low flow, or baseflow metrics.

See [metrics.all](#) to calculate 30 different streamflow metrics at once. The [screen.metric](#) function could then be used to loop through the metrics and create an individual plot for each.

## Examples

```
data(cania.sub.ts)

# calculate and plot the annual maximum series
res <- pk.max(cania.sub.ts)
res1 <- screen.metric(res, ylabel="Q (m3/s)",
```

```
text="Caniapiscou River, Annual Maximum Series")

# calculate and plot the annual minimum series
res <- MAMn(cania.sub.ts, n=1)
res1 <- screen.metric(res, ylabel="Discharge (m3/s)",
text="Caniapiscou River, Annual Minimum Series")
```

---

`screen.series`*Create a plot of the daily streamflow time series*

---

## Description

Plots the daily streamflow time series and color codes points by data quality codes if the data are from Water Survey Canada. Also highlights date ranges with missing observations.

## Usage

```
screen.series(TS, StnInfo = NULL, text = "d")
```

## Arguments

TS	output from <a href="#">create.ts</a> containing a data.frame of flow time series
StnInfo	Optional data.frame containing user-supplied station info for plot. data.frame must have 7 columns containing station info in the following order: Station ID, Station Name, Prov/State, Country, Latitude, Longitude, Catchment Area If any of the information is unavailable, fill with NA. The Station ID column must match the Station ID in column 1 of the data.frame input from <a href="#">create.ts</a> .
text	optional character string for margin text, e.g. for station name, location, or other notes. Set to NULL if not margin text is wanted, or set to "d" to use default text containing the station ID, station name, and province/state returned from <a href="#">station.info</a> .

## Author(s)

Jennifer Dierauer and Paul Whitfield

## Examples

```
# load flow time series for the Caniapiscou River
data(cania.sub.ts)

# plot daily time series with default margin text
screen.series(cania.sub.ts)
```

---

screen.summary      *Create a screening plot*

---

### Description

Produces summary screening plots of high flow, low flow, or baseflow metrics. Each plot shows significant temporal trends and step changes. Intended for use as a data quality screening tool aimed at identifying streamflow records with anthropogenic impacts or data inhomogeneities.

### Usage

```
screen.summary(metrics, type = "h", language = "English", StnInfo = NULL)
```

### Arguments

metrics	output from <a href="#">metrics.all</a>
type	Character indicating the set of metrics to plot. Options are "h" for high flow metrics, "l" for low flow metrics, or "b" for baseflow metrics.
language	Language for plot labels. Choice of either "English" or "French". Default is "English".
StnInfo	Optional data.frame containing user-supplied station info for plot. data.frame must have 7 columns containing station info in the following order: Station ID, Station Name, Prov/State, Country, Latitude, Longitude, Catchment Area. If any of the information is unavailable, fill with NA. The Station ID column must match the Station ID in column 1 of the data.frame input from <a href="#">create.ts</a> .

### Details

For the center of volume (COV) plots on the high flow and baseflow screening plots, the correlation coefficients for COV and years and for mean annual flow (MAF) and years are added to the plot. The ratio of the correlation coefficients ( $r_{COV-years} / r_{COV-MAF}$ ) is included as a rudimentary indication of whether or not the temporal trend in COV is meaningful. See Whitfield (2013) for a discussion of COV.

Drought metrics for the low flow plot may not be applicable to intermittent streams, and plots will be empty in this case.

Important note: If "French" is the language wanted for the plot labels, the language option must also be specified in [metrics.all](#), as this plotting function pulls the metric names from the output metrics.all output.

### Author(s)

Jennifer Dierauer

### References

Whitfield, P.H. 2013. Is 'Center of Volume' a robust indicator of changes in snowmelt timing? *Hydrological Processes* 27:2691-8.



**Examples**

```
# load results from metrics.all function for the Caniapiscau River
data(caniapiscau.res)

# create a summary flow screening plot of the high flow metrics
screen.summary(caniapiscau.res, type="1")
```

---

```
screen.summary.internal
```

*Internal wrapper for creating trend and change-point summary plots*

---

**Description**

Internal wrapper for creating trend and change-point summary plots

**Usage**

```
screen.summary.internal(input, mparam, type, ylabs, i, DataType, maf, Year1,
  YearEnd, hrstart)
```

**Arguments**

input	metric time series
mparam	trend and change point info
type	character indicating type of summary plot
ylabs	y axis labels
i	plot position
DataType	numeric indicating data type
maf	mean annual flow series
Year1	start year of original time series
YearEnd	end year of original time series
hrstart	numeric indicating month for start of the hydrologic year

**Author(s)**

Jennifer Dierauer

---

station.info	<i>Retrieve Station Info</i>
--------------	------------------------------

---

**Description**

Returns station information for Water Survey Canada or United States Geological Survey stream gauges.

**Usage**

```
station.info(TS, StnInfo = NULL, Plot = F, language = "English")
```

**Arguments**

TS	output from <a href="#">create.ts</a> containing a data.frame of flow
StnInfo	Optional data.frame containing user-supplied station info for plot. data.frame must have 7 columns containing station info in the following order: Station ID, Station Name, Prov/State, Country, Latitude, Longitude, Catchment Area. If any of the information is unavailable, fill with NA. The Station ID column must match the Station ID in column 1 of the data.frame input from <a href="#">create.ts</a> .
Plot	Boolean indicating whether a plot of station information should be created. Default is F. Plot is intended for use as the upper-left panel of the plot produced by <a href="#">screen.summary</a> .
language	Language for plotting when Plot = T. Choice of either "English" or "French". Default is "English".

**Value**

Returns a list of the following station information:

- \$StationID
- \$StnName
- \$Prov/State - Abbreviation for the province or state in which the station is located
- \$Country - Abbreviation for the country in which the station is located
- \$Lat - Latitude of the station
- \$Long - Longitude of the station
- \$Area - Catchment area, in square kilometers
- \$RHN - Boolean indicating whether the station is part of a reference hydrologic network

**Author(s)**

Jennifer Dierauer

**Examples**

```
data(cania.sub.ts)
StnInfo <- station.info(cania.sub.ts)
```

---

YMD.internal	<i>Add calendar year, month, and day of year columns</i>
--------------	--

---

**Description**

Add calendar year, month, and day of year columns

**Usage**

```
YMD.internal(TS)
```

**Arguments**

TS                    Output from [create.ts](#) function.

**Value**

Returns a data.frame with year, month, and doy columns appended.

**Author(s)**

Jennifer Dierauer

# Index

## \* datasets

cania.sub.ts, 7  
caniapiscau, 9  
caniapiscau.res, 10

## \* package

FlowScreen, 16

axis\_doy.internal, 3

bf.seas, 3, 16  
bf.stats, 4, 4, 16, 22  
bf\_boughton, 5  
bf\_eckhardt, 3, 4, 6, 26  
bf\_oneparam, 7

cania.sub.ts, 7  
caniapiscau, 9  
caniapiscau.res, 10  
confint.zyp, 38  
cov, 21  
cpt.meanvar, 17, 21, 38  
create.ts, 3, 4, 10, 11, 13–15, 19, 20, 23–31,  
33, 35, 39, 40, 42, 43

decluster, 17, 30  
dr.events, 11, 14, 15, 24  
dr.pds, 12, 13, 14, 24  
dr.seas, 12, 14, 14, 16, 21, 24

FDC, 15  
FlowScreen, 16

get.station.internal, 18  
get.titles.internal, 18

hyear.internal, 19

Kendall, 17

MAMn, 16, 19, 22  
metrics.all, 10, 16, 20, 33, 34, 38, 40

mqt, 12, 14, 15, 20, 23

NA.runs, 24, 25  
NA.sum, 25, 25

pk.bf.stats, 16, 22, 26  
pk.cov, 16, 21, 27  
pk.max, 16, 21, 28, 29  
pk.max.doy, 16, 21, 28, 29  
pks, 21, 29, 30  
pks.dur, 21, 30  
pot, 17, 30

Qn, 16, 21, 31

read.flows, 9, 11, 16, 32  
regime, 16, 32

screen.cpts, 10, 16, 33  
screen.frames, 10, 16, 34  
screen.frames.internal, 37  
screen.metric, 16, 20, 22, 37, 38  
screen.series, 39  
screen.summary, 10, 16, 21, 38, 40, 42  
screen.summary.internal, 41  
station.info, 33, 35, 39, 42

YMD.internal, 43

zyp.trend.vector, 17, 21, 22, 38