Supplement of

**Hydrograph separation: an impartial parametrisation for an imperfect method**

Antoine Pelletier and Vazken Andréassian

Correspondence to: Antoine Pelletier (antoine.pelletier@inrae.fr)

The copyright of individual parts of the supplement might differ from the CC BY 4.0 License.
Package ‘baseflow’

November 13, 2019

Type Package
Title Computes Hydrograph Separation
Version 0.11.2
Date 2019-11-05
Depends R (>= 3.0.3)
Imports methods, graphics, airGR
License GPL-2
SystemRequirements cargo (>= 1.36.0), rustc, GNU make
Contact antoine.pelletier@irstea.fr
Encoding UTF-8
NeedsCompilation yes
Author Antoine Pelletier [cre, aut] (<https://orcid.org/0000-0002-1286-9147>), Vazken Andréassian [ths, rev] (<https://orcid.org/0000-0001-7124-9303>), Olivier Delaigue [rev. ctb, dtc] (<https://orcid.org/0000-0002-7668-8468>)
Maintainer Antoine Pelletier <antoine.pelletier@irstea.fr>
Repository CRAN
Date/Publication 2019-11-13 16:20:02 UTC

R topics documented:

- BaseflowFilter ................................................................. 2
- BaseflowFilter-class ....................................................... 3
- BasinData ................................................................. 5
- BasinData-class ........................................................... 6
- bfi ................................................................. 7
Description

Standard construction function of BaseflowFilter objects, used by package baseflow to store filtering information for a given catchment. The object is created with NA values of baseflow and level of filtering reservoir, it is then necessary to run perform_filtering function to compute them.

Usage

BaseflowFilter(BasinData, alpha, updateFunction = "quadr")

Arguments

BasinData A BasinData object containing hydroclimatic data for the catchment under study.
alpha Numeric parameter of the filtering reservoir. Its dimension depends on the used update function.
updateFunction Character string giving reservoir function : must be one of "quadr", "lin" or "exp". Default is "quadr".

Value

A BaseflowFilter object containing provided data. Baseflow numeric field R and reservoir level V are filled with NA, it is thus necessary to call perform_filtering function to compute R and V.

Author(s)

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References

BaseflowFilter-class

See Also

- BasinData
- BaseflowFilter
- BasinData
- perform_filtering

Examples

```r
library(baseflow)

# Loading example data from airGR package
data(L0123001, package = 'airGR')

# Defining BasinData object
Name <- BasinInfo$BasinName
startDate <- BasinObs$DatesR[1]
endDate <- BasinObs$DatesR[length(BasinObs$DatesR)]
P <- BasinObs$P
PET <- BasinObs$PET
Qobs <- BasinObs$Qmm

BasinData_Example <- BasinData(Name, startDate, endDate, P, PET, Qobs, fill = "GR4J")

# Creating BaseflowFilter object
BaseflowFilter_Example <- BaseflowFilter(BasinData_Example, 1000, updateFunction = 'quadr')

# Computing baseflow
BaseflowFilter_Example <- perform_filtering(BaseflowFilter_Example)

# Plotting computed separation
plot(BaseflowFilter_Example)
```

Description

Class used by package baseflow to store results of a baseflow filtering for a given catchment.

Objects from the Class

Objects can be created by calls of the form `new("BaseflowFilter", ...),` but it is recommended to use BaseflowFilter function to create this object.
BaseflowFilter-class

Slots

- BasinData: Object of class "BasinData" giving hydroclimatic data of catchment (see BasinData-class).
- R: Vector of class "numeric" giving filtered baseflow.
- V: Vector of class "numeric" giving storage of the conceptual reservoir.
- update: Vector of class "logical", TRUE at timesteps in which reservoir's level is updated, FALSE otherwise.
- updateFunction: Object of class "character" equal to "quadr", "lin" or "exp", giving the nature of the conceptual reservoir.
- alpha: Object of class "numeric": value of "alpha" reservoir parameter.

Methods

- as.data.frame signature(x = "BaseflowFilter"): transforms object into dataframe.
- plot signature(x = "BaseflowFilter"): plots baseflow separation on the hydrograph.
- print signature(x = "BaseflowFilter"): prints object to console.
- show signature(object = "BaseflowFilter"): shows object in console.
- summary signature(object = "BaseflowFilter"): shows a summary of the object.

Author(s)

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References


See Also

- BaseflowFilter
- perform_filtering
- BasinData

Examples

showClass("BaseflowFilter")
**BasinData**

*Constructor function of class BasinData*

**Description**

Standard construction function of BasinData objects, used by package baseflow to store daily hydroclimatic data for a given catchment. As the separation algorithm proposed in baseflow does not handle missing streamflow values, a filling routine using airGR package is provided.

**Usage**

```r
BasinData(Name, startDate, endDate, P, PET, Qobs, fill = "none")
```

**Arguments**

- **Name** Character string giving the name of the catchment. Can be a code or a plain-text description.
- **startDate** Starting date of hydroclimatic data as a POSIXct object.
- **endDate** End date of hydroclimatic data as a POSIXct object.
- **P** Numeric vector giving daily total lumped precipitation over the catchment, in mm. Must be of the same length as PET and Qobs.
- **PET** Numeric vector giving daily total lumped potential evapotranspiration over the catchment, in mm. Must be of the same length as P and Qobs.
- **Qobs** Numeric vector giving daily streamflow of the catchment, as depth of runoff in mm. Must be of the same length as PET and Qobs.
- **fill** Character string describing filling methods. Must be one of the following: "none" (no filling, default), "GR4J", "GR5J" or "GR6J".

**Details**

The expected length of data vectors is equal to the number of days between the two provided dates. It is recommended to store data into a data frame before using this function. This function does not handle missing values: any NA will produce an error. If there are missing values in streamflow, the fill argument must be different from "none". Filling routine requires airGR package. One of the following hydrological lumped models can be used: GR4J, GR5J or GR6J. See airGR documentation for further details.

**Value**

A BasinData object containing provided data.

**Author(s)**

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)
References


See Also

BaselineData
BaseflowFilter
BaseflowFilter

Examples

library(baseflow)

# Loading example data from airGR package
data(L0123001, package = "airGR")

# Defining BasinData object
Name <- BasinInfo$BasinName
startDate <- BasinObs$DatesR[1]
endDate <- BasinObs$DatesR[length(BasinObs$DatesR)]
P <- BasinObs$P
PET <- BasinObs$PET
Qobs <- BasinObs$Qmm

BasinData_Example <- BasinData(Name, startDate, endDate, P, PET, Qobs, fill = "GR4J")

---

BasinData-class  Class BasinData

Description

Class used by package baseflow to store hydroclimatic data from a given catchment.

Objects from the Class

Objects can be created by calls of the form new("BasinData",...), but it is recommended to use BasinData function to create this object.

Slots

Name: Object of class "character" giving the name of the catchment (e.g. 'Seine in Paris'

Dates: Vector of "POSIXct" dates giving timestamps of measures

nbTimeStep: Object of class "integer" giving the number of measures

P: Vector of total precipitation of length 'nbTimeStep'

PET: Vector of potential evapotranspiration of length 'nbTimeStep'

Qobs: Vector of observed streamflow of length 'nbTimeStep'
Methods

- **as.data.frame** signature(x = "BasinData"): transforms object into a data frame.
- **print** signature(x = "BasinData"): prints object to console
- **show** signature(object = "BasinData"): shows object
- **summary** signature(object = "BasinData"): show a summary of object

Author(s)

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References


See Also

- BasinData
- BaseflowFilter

Examples

```r
showClass("BasinData")
```

---

**bfi**  
*Baseflow index computing function*

Description

Function provided by package baseflow to compute baseflow index from a BaseflowFilter object.

Usage

```r
bfi(filter)
```

Arguments

- **filter** A BaseflowFilter object created by BaseflowFilter function.

Details

Computes baseflow index from a BaseflowFilter object, created by BaseflowFilter function. If the baseflow values have not been computed yet in the object, a Rust routine is called to compute baseflow; else, baseflow is computed directly from baseflow and observed streamflow values. Providing a non-BaseflowFilter object creates an error.
Value

A numeric giving computed baseflow index.

Author(s)

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References


See Also

BaseflowFilter
BaseflowFilter
perform_filtering

Examples

library(baseflow)

# Loading example data from airGR package
data(L0123001, package = 'airGR')

# Defining BasinData object
Name <- BasinInfo$BasinName
startDate <- BasinObs$DatesR[1]
endDate <- BasinObs$DatesR[length(BasinObs$DatesR)]
P <- BasinObs$P
PET <- BasinObs$E
Qobs <- BasinObs$Qmm
BasinData_Example <- BasinData(Name, startDate, endDate, P, PET, Qobs, fill = "GR4J")

# Creating BaseflowFilter object
BaseflowFilter_Example <- BaseflowFilter(BasinData_Example, 1000, updateFunction = 'quadr')

# Computing baseflow
BaseflowFilter_Example <- perform_filtering(BaseflowFilter_Example)

# Plotting computed separation
plot(BaseflowFilter_Example)

# Computing baseflow index
bfi(BaseflowFilter_Example)
Description

Computation function of correlation criterion used by package baseflow to calibrate parameters of the baseflow separation algorithm. A vectorized version of this function correlation_criteria_vectorized also exists.

Usage

corr_crit(BasinData, alpha, tau, updateFunction = "quadr")

Arguments

BasinData    A BasinData object containing hydroclimatic data for the catchment under study.
alpha        Numeric parameter of the filtering reservoir. Its dimension depends on the used update function. Must be positive.
tau          Number of days used to compute cumulative effective rainfall. Must be a positive integer or any object that can be coerced to a positive integer.
updateFunction Character string giving reservoir function : must be one of "quadr", "lin" or "exp". Default is "quadr".

Details

NA values are not permitted for parameters alpha and tau. If a vector is supplied, the first element is used.

Usage of gradient-based optimization algorithms is not recommended to find the maximum value of this criterion. It is a better option to use vectorized function correlation_criteria_vectorized to perform a grid-search; a non-gradient-based optimization algorithm, like differential evolution, may be used to refine the optimal point afterwards.

Value

A numeric value of the computed criterion, between -1 and 1.

Author(s)

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References

See Also

perform_filtering
BaseflowFilter
corr_crit_vect
bfi

Examples

library(baseflow)

# Loading example data from airGR package
data(L0123001, package = 'airGR')

# Defining BasinData object

Name <- BasinInfo$BasinName
startDate <- BasinObs$DatesR[1]
endDate <- BasinObs$DatesR[length(BasinObs$DatesR)]
P <- BasinObs$P
PET <- BasinObs$PET
Qobs <- BasinObs$Qmm

BasinData_Example <- BasinData(Name, startDate, endDate, P, PET, Qobs, fill = "GR4J")

# Computing correlation criterion
corr_crit(BasinData_Example, 1500, 110, updateFunction = "quadr")

corr_crit_vect  Baseflow vectorized correlation criteria computation

Description

Vectorized computation function of correlation criteria used by package baseflow to calibrate parameters of the baseflow separation algorithm. A scalar version of this function correlation_criteria also exists.

Usage

corr_crit_vect(BasinData, alphas, taus, updateFunction = "quadr")

Arguments

Baseline  A BasinData object containing hydroclimatic data for the catchment under study.
alphas  Numeric vector of parameters of the filtering reservoir. Their dimension depends on the used update function. They must be positive.
taus  Vector of numbers of days used to compute cumulative effective rainfall. Must be an integer vector or any object that can be coerced to an integer vector. All values must be positive.

updateFunction Character string giving reservoir function: must be one of "quadr", "lin" or "exp". Default is "quadr".

Details
Providing empty vectors or vectors containing missing or non-positive values for parameters alphas and taus throws an error.

Value
A dataframe containing four columns and length(alphas) * length(taus) rows. Columns are the following:

- alpha  Values of alpha
- tau  Values of tau
- bfi  Values of baseflow index, which depends on alpha
- crit  Computed values of correlation criterion

Author(s)
Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References

See Also
perform_filtering
BaseflowFilter
corr_crit
bfi

Examples

library(baseflow)

# Loading example data from airGR package
data(L0123001, package = 'airGR')

# Defining BasinData object
Name <- BasinInfo$BasinName
startDate <- BasinObs$DatesR[1]
perform_filtering

Description

Function provided by package baseflow to compute baseflow variables from a raw BaseflowFilter object. It must be called after calling the BaseflowFilter function.

Usage

perform_filtering(filter)

Arguments

filter A BaseflowFilter object created by BaseflowFilter function.

Details

Providing a non-BaseflowFilter object creates an error. The function calls a Rust routine to compute vectors R (baseflow) and V (reservoir level) from parameters and hydroclimatic data stored in filter object.

Value

A BaseflowFilter object with the same parameters and BasinData slots as input, but R, V and update slots are replaced by computed values.

Author(s)

Pelletier and Andreassian (<antoine.pelletier@irstea.fr>)

References

See Also

BaseflowFilter
BaseflowFilter
bfi

Examples

library(baseflow)

# Loading example data from airGR package
data(L0123001, package = 'airGR')

# Defining BasinData object

Name <- BasinInfo$BasinName
startDate <- BasinObs$DatesR[1]
endDate <- BasinObs$DatesR[length(BasinObs$DatesR)]
P <- BasinObs$P
PET <- BasinObs$E
Qobs <- BasinObs$Qmm

BasinData_Example <- BasinData(Name, startDate, endDate, P, PET, Qobs, fill = "GR4J")

# Creating BaseflowFilter object
BaseflowFilter_Example <- BaseflowFilter(BasinData_Example, 1000, updateFunction = 'quadr')

# Computing baseflow
BaseflowFilter_Example <- perform_filtering(BaseflowFilter_Example)

# Plotting computed separation
plot(BaseflowFilter_Example)
Index

*Topic **classes**
  BaseflowFilter-class, 3
  BasinData-class, 6
*Topic **manip**
  BaseflowFilter, 2
  BasinData, 5
  bfi, 7
  corr_crit, 9
  corr_crit_vect, 10
  perform_filtering, 12

as.data.frame,BaseflowFilter-method
  (BaseflowFilter-class), 3
as.data.frame,BasinData-method
  (BasinData-class), 6

BaseflowFilter, 2, 3, 4, 6–8, 10, 11, 13
BaseflowFilter-class, 3
BasinData, 3, 4, 5, 6, 7
BasinData-class, 4, 6
bfi, 7, 10, 11, 13

corr_crit, 9, 11
corr_crit_vect, 10, 10

perform_filtering, 3, 4, 8, 10, 11, 12
plot,BaseflowFilter-method
  (BaseflowFilter-class), 3
print,BaseflowFilter-method
  (BaseflowFilter-class), 3
print,BasinData-method
  (BasinData-class), 6

show,BaseflowFilter-method
  (BaseflowFilter-class), 3
show,BasinData-method
  (BasinData-class), 6
summary,BaseflowFilter-method
  (BaseflowFilter-class), 3
summary,BasinData-method
  (BasinData-class), 6