Package 'topmodel'

February 1, 2018

Type Package

Title Implementation of the Hydrological Model TOPMODEL in R

Version 0.7.3

Date 2018-01-31

Author Wouter Buytaert

Maintainer Wouter Buytaert <w.buytaert@imperial.ac.uk>

Description Set of hydrological functions including an R implementation of the hydrological model TOPMODEL, which is based on the 1995 FORTRAN version by Keith Beven. From version 0.7.0, the package is put into maintenance mode.

License GPL (>= 2)

Imports methods

URL http://paramo.cc.ic.ac.uk/topmodel_tutorial

Repository CRAN

NeedsCompilation yes

Date/Publication 2018-02-01 22:21:11 UTC

R topics documented:

flowlength 2
huagrahuma 2
huagrahuma.dem 3
infiltration 3
make.classes 4
NSeff 5
outlet 6
river 7
sinkfill 8
subcatch 9
topidx 10
topmodel 11

Index **14**

`flowlength` *Calculates the flow distance towards the catchment outlet*

Description

Given a digital elevation model and the coordinates of the catchment outlet, this function calculates the flowlength of each gridcell to the outlet, based on a single flow direction algorithm (D8).

Usage

```
flowlength(DEM,outlet)
```

Arguments

`DEM`	A matrix representing a digital elevation model [m] with equally sized pixels and equal NS and EW resolution
`outlet`	A vector containing the row and column indices of the pixel representing the catchment outlet.

Details

The function returns the flowlength in cell size units. So you have to multiply by the map resolution to get the flow in meters.

Value

A matrix of the same size as `DEM`

Author(s)

Wouter Buytaert, Imperial College London, based on an implementation from the Hydrology Group of Lancaster University

References

See `http://paramo.cc.ic.ac.uk/topmodel_tutorial` for examples.

See Also

sinkfill, river

`huagrahuma` *Hydrological dataset to run TOPMODEL*

Description

This is an example dataset from a small mountainous catchment in the South American Andes and contains the following data:

delay	Cumulative delay function of the catchment
ETp	FAO Penman Monteith potential evapotranspiration [m / 15 min]
parameters	Values for the TOPMODEL parameters
Qobs	Observed discharge [m / 15 min]
rain	Observed precipitation [m / 15 min]

Usage

```
huagrahuma
```

References

Buytaert, W., De Bievre, B., Wyseure, G., Deckers, J., 2005. The effect of land use changes on the hydrological behaviour of Histic Andosols in south Ecuador. Hydrological Processes 19: 3985 - 3997.

`huagrahuma.dem` *DEM of the Huagrahuma catchment, Ecuador*

Description

Matrix containing a Digital Elevation Model of the Huagrahuma microcatchment, Ecuador, at 25 m resolution.

Usage

```
huagrahuma.dem
```

References

Buytaert, W., De Bievre, B., Wyseure, G., Deckers, J., 2005. The effect of land use changes on the hydrological behaviour of Histic Andosols in south Ecuador. Hydrological Processes 19: 3985 - 3997.

`infiltration` *Infiltration routine of TOPMODEL*

Description

Infiltration routine used in TOPMODEL, based on the Green-Ampt model.

Usage

```
infiltration(rain, parameters)
```

Arguments

`rain`	A vector of rain data (m per timestep)
`parameters`	A vector containing 3 parameters (see below for the exact structure)

Details

This function gives direct access to the infiltration routine implemented in `topmodel()`. The function needs three parameters: `c(dt, CD, K0, m)`, where:

dt	The timestep (hours)
CD	capillary drive, see Morel-Seytoux and Khanji (1974)
K0	Surface hydraulic conductivity (m/h)
m	Model parameter controlling the rate of decline of transmissivity in the soil profile, see Beven, 1984

Value

The function returns a vector with the same length as the input vector rain representing infiltration.

Author(s)

Wouter Buytaert, Imperial College London, based on original FORTRAN code from Lancaster University

References

Morel-Seytoux, H.J., Khanji, J., 1974. Derivation of an Equation of Infiltration. Water Resources Research, 10, 795-800. Beven, K., 1984. Infiltration into a Class of Vertically Non-Uniform Soils. Hydrological Sciences Journal 29, 425-434.

See also `http://paramo.cc.ic.ac.uk/topmodel_tutorial` for examples.

See Also

`topmodel`

`make.classes` *make topographic index classes from a topographic index map*

Description

This function splits a dataset in n evenly distributed classes and calculates the number of elements of each class. It is very similar to `hist()`, but `hist()` does not always keep the number of breaks requested

Usage

```
make.classes(array,n)
```

Arguments

`array`	A numeric vector or matrix
`n`	Number of breaks

Value

The function returns a data matrix with 2 columns. Column `breaks` contains the upper limits of each class, while `counts` contains the number of elements

Author(s)

Wouter Buytaert, Imperial College London

See Also

`topidx`, `topmodel`

`NSeff` *Calculation of the Nash-Sutcliffe efficiency*

Description

Calculation of the Nash-Sutcliffe efficiency

Usage

```
NSeff(Qobs,Qsim)
```

Arguments

`Qobs`	A vector with observed discharges
`Qsim`	A vector with simulated discharges of the same length as Qobs

Details

Qobs and Qsim should have the same dimensions. Both can contain NA values.

Author(s)

Wouter Buytaert, Imperial College London

See Also

`topmodel`

outlet *Conveniently query a subset of a matrix*

Description

Convenience function to extract a pixel and its neighbourhood from matrices, e.g. to query raster maps.

Usage

```
outlet(map,out,radius = 2)
```

Arguments

map	matrix
out	Center of the matrix to be extracted
radius	Radius of the matrix to be extracted

Details

The function can be used to find a catchment outlet using a drainage area map generated by `atb`, when the coordinates are approximately known (for instance from the location of a discharge station). These coordinates can then be used by `subcatch`. However, note that `atb` uses a multiple direction flow algorithm, while `subcatch` uses single flow. Therefore, the drainage area for a pixel calculated by `atb` is likely to differ from the catchment area identified by `subcatch`.

Value

Square matrix of size (radius * 2 + 1)

Author(s)

Wouter Buytaert, Imperial College London

References

See `http://paramo.cc.ic.ac.uk/topmodel_tutorial` for examples.

See Also

`subcatch`

river — *Identification of rivers in a digital elevation model*

Description

This function identifies river headwater cells based on threshold values in a topographic index and accumulated area river (generated using `topidx`. Rivers are then traced downslope using a single flow algorithm.

Usage

```
river(DEM,atb,area,res,thatb,tharea)
```

Arguments

`DEM`	A matrix representing a digital elevation model [m] with equally sized pixels and equal NS and EW resolution
`atb`	Matrix with topographic index values generated by `topidx`
`area`	Matrix with drainage area values generated by `topidx`
`res`	Resolution of the digital elevation model (m)
`thatb`	A topographic index threshold for headwater cells
`tharea`	A drainage threshold for headwater cells

Details

Cells that exceed thatb or tharea in the respective maps are identified as headwater cells. The routine then traces down rivers from these cells based on a D8 algorithm and calculates the distance towards the outlet. Outlets are recognized by sinks, map borders or excluded areas (NA). The `subcatch` function can be used to set areas outside the target catchment to NA.

Value

A matrix of the same size as DEM.

Author(s)

Wouter Buytaert, Imperial College London, based on an implementation from the Hydrology Group of Lancaster University

References

See http://paramo.cc.ic.ac.uk/topmodel_tutorial for examples.

See Also

`subcatch`, `topidx`

sinkfill *Fill sinks in a digital elevation model*

Description

Removes sinks in a digital elevation model by filling depressions

Usage

```
sinkfill(DEM,res,degree)
```

Arguments

DEM	A matrix representing a digital elevation model [m] with equally sized pixels and equal x and y resolution
res	Resolution of the digital elevation model [m]
degree	Minimum slope to be kept between cells when filling [degrees]

Details

For deep sinks or large maps, it may be possible that not all sinks are filled in one run. Then the function should be applied repeatedly over the same object.

Value

A matrix of the same size as the DEM.

Author(s)

Wouter Buytaert, Imperial College London, based on an implementation from the Hydrology Group of Lancaster University

References

See http://paramo.cc.ic.ac.uk/topmodel_tutorial for examples.

See Also

topmodel, topidx

Examples

```
data(huagrahuma.dem)
filled.dem <- sinkfill(huagrahuma.dem, 25, 0.1)
```

`subcatch` *Identify a hydrological catchment based on a single direction flow algorithm*

Description

Identify a hydrological catchment based on a single direction flow algorithm

Usage

```
subcatch(DEM,outlet)
```

Arguments

`DEM`	A matrix representing a digital elevation model [m] with equally sized pixels and equal x and y resolution
`outlet`	A vector containing the row and column indices of the pixel representing the catchment outlet.

Value

A binary matrix of the same size as `DEM`.

Author(s)

Wouter Buytaert, Imperial College London, based on an implementation from the Hydrology Group of Lancaster University

References

See `http://paramo.cc.ic.ac.uk/topmodel_tutorial` for examples.

See Also

`topmodel`

topidx *Calculation of the topographic index from a DEM raster*

Description

Calculation of the topographic index from a DEM raster

Usage

```
topidx(DEM, resolution, river = NULL)
```

Arguments

DEM	A matrix representing a digital elevation model with equally sized pixels and equal NS and EW resolution
resolution	Resolution of the digital elevation model (m)
river	A matrix representing a river map (optional). All cells with values higher than 0 will be treated as river cells and are not included in the calculations.

Details

The river map should not contain negative values

Value

The function returns a list, containing 2 rasters. Atb are the topographic index values of each grid cell. Area contains the contributing area

Author(s)

Wouter Buytaert, Imperial College London, based on routines developed by the hydrology group of Lancaster University

References

See http://paramo.cc.ic.ac.uk/topmodel_tutorial for examples.

See Also

topmodel

Examples

```
data(huagrahuma.dem)
topidx <- topidx(huagrahuma.dem, resolution= 25)$atb
image(topidx)
```

`topmodel` *implementation of the hydrological model TOPMODEL*

Description

Implementation of the 1995 Fortran version of TOPMODEL by Keith Beven.

Usage

```
topmodel(parameters, topidx, delay, rain, ETp, verbose = F, Qobs=NA)
```

Arguments

`parameters`	A vector or a matrix containing the input parameters (see below for the exact structure)
`topidx`	A 2 column matrix with respectively the topographic index classes and values (see below for the exact structure)
`delay`	Delay function for overland flow (see below)
`rain`	A vector of rain data (m per timestep)
`ETp`	A vector of potential evapotranspiration data (m per timestep)
`verbose`	If set to TRUE, returns besides predicted discharge also overland flow, base flow and saturated zone storage
`Qobs`	If Qobs is given, normal output is suppressed and only the Nash and Sutcliffe efficiency is returned (m per timestep)

Details

topmodel() automatically implements a Monte Carlo simulation. If the parameter argument is a single vector, the model is run once. If the parameter argument is a matrix, each row should represent a parameter set. In that case, the model is run with each parameter set (see the examples below).

A single parameter set consists of: `c(qs0,lnTe,m,Sr0,SrMax,td,vch,vr,k0,CD,dt)`, with:

qs0	Initial subsurface flow per unit area [m]
lnTe	log of the areal average of T0 [m2/h]
m	Model parameter controlling the rate of decline of transmissivity in the soil profile, see Beven, 1984
Sr0	Initial root zone storage deficit [m]
Srmax	Maximum root zone storage deficit [m]
td	Unsaturated zone time delay per unit storage deficit [h/m]
vch	channel flow outside the catchment [m/h] (currently not used)
vr	channel flow inside catchment [m/h]
k0	Surface hydraulic conductivity [m/h]
CD	capillary drive, see Morel-Seytoux and Khanji (1974)
dt	The timestep [h]

The topidx dataframe can be derived conveniently with `make.classes()`. It should contain 2 columns. The first column should give the lower boundary of each topographic index class, and the second column should give the respective area fraction. The second column must sum to 1.

k0 and CD are used only for the infiltration excess routine. Set k0 to a very high value to avoid infiltration excess overland flow

Flow is routed through a delay function which represents the time spent in the channel system. The parameter `delay` is used for this. Delay is a matrix with 2 columns. The first column gives the cumulative relative area. The second column gives the average distance towards the outlet (m).

Value

The function returns an array of observed discharges. If more than one parameter set is given, a matrix is returned, with each column representing a discharge set coinciding with the parameter sets. If Qobs is given, the function returns an array of Nash-Sutcliffe efficiencies, 1 for each parameter sets.

If verbose output is requested, a list is returned, with the modelled discharge (Q), overland flow (qo), subsurface flow (qs), storage deficit (S), infiltration excess overland flow (fex), and actual evapotranspiration (Ea) for each time step.

Be aware that invoking `topmodel()` without Q for a large number of runs, may require a large amount of memory.

Author(s)

Wouter Buytaert, Imperial College London

References

Beven, K. J., Kirkby, M. J., 1979. A physically based variable contributing area model of basin hydrology. Hydrol. Sci. Bull. 24, 43-69.

Beven K, Lamb R, Quinn P, Romanowicz R, Freer J, 1995. TOPMODEL. In: Sing VP (Ed), Computer Models of Watershed Hydrology. Water Resources Publications, Colorado. pp. 627-668.

Morel-Seytoux, H.J., Khanji, J., 1974. Derivation of an Equation of Infiltration. Water Resources Research, 10, 795-800. Beven, K., 1984. Infiltration into a Class of Vertically Non-Uniform Soils. Hydrological Sciences Journal 29, 425-434.

See also `http://paramo.cc.ic.ac.uk/topmodel_tutorial` for a more examples on how to run topmodel in R.

See Also

`topidx`

Examples

```
data(huagrahuma)
attach(huagrahuma)

## returns the simulated runoff (Qobs not given)
Qsim <- topmodel(parameters, topidx, delay, rain, ETp)
```

```
## returns a list of simulated runoff (Q), overland flow (qo), subsurface flow (qs) and storage (S):
Qsim <- topmodel(parameters, topidx, delay, rain,ETp, verbose = TRUE)

## plot observed and simulated discharge:
plot(Qobs)
points(Qsim$Q, col="red", type="l")

## For a Monte carlo sampling from a uniform distribution, we construct a parameter matrix:

runs<-10

qs0   <- runif(runs, min=0, max=4e-5)
lnTe  <- runif(runs, min=-2, max=1)
m     <- runif(runs, min=0, max=0.2)
Sr0   <- runif(runs, min=0, max=0.02)
Srmax <- runif(runs, min=0, max=2)
td    <- runif(runs, min=0, max=3)
vch   <- 1000
vr    <- runif(runs, min=100, max=2500)
k0    <- runif(runs, min=0, max=0.01)
CD    <- runif(runs, min=0, max=5)
dt    <- 0.25

parameters<-cbind(qs0,lnTe,m,Sr0,Srmax,td,vch,vr,k0,CD,dt)

## returns an array of 10 Nash Sutcliffe efficiencies; one for each parameter set:
result<-topmodel(parameters, topidx, delay, rain, ETp, Qobs = Qobs)
```

Index

∗Topic **datasets**
huagrahuma, 2
huagrahuma.dem, 3
∗Topic **models**
infiltration, 3
make.classes, 4
NSeff, 5
topidx, 10
topmodel, 11

delay *(*huagrahuma*)*, 2

ETp *(*huagrahuma*)*, 2

flowlength, 2

huagrahuma, 2
huagrahuma.dem, 3

infiltration, 3

make.classes, 4

NSeff, 5

outlet, 6

parameters *(*huagrahuma*)*, 2

Qobs *(*huagrahuma*)*, 2

rain *(*huagrahuma*)*, 2
river, 7

sinkfill, 8
subcatch, *6*, *7*, 9

topidx, *5*, *7*, 10, *12*
topmodel, *4*, *5*, *9*, *10*, 11