

Package ‘RcamelsCL’

May 18, 2026

Type Package

Title Easy Handling of the CAMELS-CL Dataset

Version 0.2-0

Maintainer Hector Garces-Figueroa <hegarcesf@gmail.com>

Description Download and handle spatial and temporal data from the CAMELS-CL dataset (Catchment Attributes and Meteorology for Large Sample Studies, Chile) <<https://camels.cr2.cl/>>, developed by Alvarez-Garreton et al. (2018) <[doi:10.5194/hess-22-5817-2018](https://doi.org/10.5194/hess-22-5817-2018)>. The package does not generate new data, it only facilitates direct access to the original dataset for hydrological analyses.

License GPL (>= 2)

Depends R (>= 4.0.0)

Imports htr2, zoo (>= 1.7-2), hydroTSM (>= 0.5-0), terra (>= 1.7.78)

Suggests rmarkdown

URL <https://gitlab.com/hgarcesf/RcamelsCL>

MailingList <https://stat.ethz.ch/mailman/listinfo/r-sig-ecology>

BugReports <https://gitlab.com/hgarcesf/RcamelsCL/-/issues>

LazyLoad yes

NeedsCompilation no

Repository CRAN

Author Hector Garces-Figueroa [aut, cph, cre] (ORCID: <<https://orcid.org/0009-0001-2496-4854>>),
Mauricio Zambrano-Bigiarini [aut, cph] (ORCID: <<https://orcid.org/0000-0002-9536-643X>>)

Date/Publication 2026-05-18 18:30:29 UTC

Contents

getCatchments	2
getData	4

Index	9
--------------	----------

getCatchments	<i>getCatchments</i>
---------------	----------------------

Description

Select catchments from the CAMELS-CL dataset based on user-defined attribute filters.

Usage

```
getCatchments(attr, op, attr.val, include.values = FALSE, attr.val2 = NULL,
              include.lower = TRUE, include.upper = TRUE,
              get.data = FALSE, ...)
```

Arguments

<code>attr</code>	Character. Name of the catchment attribute used for filtering. See <code>RcamelsCL:::print_attr_names()</code> for the list of valid attribute names.
<code>op</code>	Character. Comparison operator applied to <code>attr</code> . Valid values are "=", "<", "<=", ">", ">=" or "between".
<code>attr.val</code>	Numeric or character. Threshold used together with <code>op</code> to filter the selected attribute. When <code>op = "between"</code> , <code>attr.val</code> defines the lower limit of the range.
<code>include.values</code>	Logical. If TRUE, the output includes both the catchment IDs and their attribute values. If FALSE, only the IDs are returned. Default is FALSE.
<code>attr.val2</code>	Numeric. Upper limit of the range when <code>op = "between"</code> .
<code>include.lower</code>	Logical. Should the lower limit defined in <code>attr.val</code> be included in the "between" operation? Default is TRUE.
<code>include.upper</code>	Logical. Should the upper limit defined in <code>attr.val2</code> be included in the "between" operation? Default is TRUE.
<code>get.data</code>	Logical. If TRUE, the function returns the data for each identified catchment using <code>RcamelsCL:::getData()</code> . If FALSE, only the matching catchment IDs are returned. Default is FALSE.
<code>...</code>	Additional arguments passed to <code>RcamelsCL:::getData()</code> when <code>get.data = TRUE</code> . See <code>?RcamelsCL:::getData</code> for details on the supported options.

Details

`getCatchments()` filters the CAMELS-CL catchments by applying a user-defined condition to a single attribute. The condition is specified through a comparison operator (`op`) and one or two threshold values (`attr.val` and `attr.val2`, when `op = "between"`).

The function first identifies all catchments whose attribute values satisfy the specified condition. By default, only the corresponding catchment IDs are returned. When `get.data = TRUE`, `getCatchments()` internally calls `getData()` for each identified catchment, returning their data in a list.

Value

If `get.data = FALSE`, a numeric vector containing the IDs of all catchments that satisfy the specified attribute filter.

If `get.data = TRUE`, a list where each element corresponds to the output of `getData()` for one of the selected catchments.

Author(s)

Hector Garces-Figueroa <hegarcesf@gmail.com>

References

<https://camels.cr2.cl>

Alvarez-Garretón, C.; Mendoza, P. A.; Boisier, J. P.; Addor, N.; Galleguillos, M.; Zambrano-Bigiarini, M.; Lara, A.; Puelma, C.; Cortes, G.; Garreaud, R.; McPhee, J.; and Ayala, A. (2018). *The CAMELS-CL dataset: catchment attributes and meteorology for large sample studies - Chile dataset*, *Hydrology and Earth System Sciences* 22, 5817-5846. doi:10.5194/hess-22-5817-2018.

Examples

```
#####
```

```
# Example 1: Getting the IDs of catchments whose most common geologic class is
# "Pyroclastics"
x1 <- getCatchments(attr = "geol_class_1st", attr.val = "Pyroclastics")
head(x1)
```

```
# Example 2: Getting the IDs of catchments whose most common geologic class is
# "Pyroclastics" and the daily time series of hydrometeorological variables
x2 <- getCatchments(attr = "geol_class_1st", attr.val = "Pyroclastics",
                    get.data = TRUE)
str(x2, 1)
```

```
# Example 3: Getting the IDs of catchments whose most common geologic class is
# "Pyroclastics" and the monthly time series of hydrometeorological variables
x3 <- getCatchments(attr = "geol_class_1st", attr.val = "Pyroclastics",
                    get.data = TRUE, tscale = "monthly")
str(x3, 1)
```

```
# Example 4: Getting the IDs of catchments whose most common geologic class is
# "Pyroclastics", the monthly time series of hydrometeorological variables,
# and the shapefile of the selected catchments
x4 <- getCatchments(attr = "geol_class_1st", attr.val = "Pyroclastics",
                    get.data = TRUE, tscale = "monthly", include.shp = TRUE)
str(x4, 2)
```

```
# Example 5: Getting the IDs of catchments with area greater than or equal to
# 20,000 km2
x5 <- getCatchments(attr = "area_km2", op = ">=", attr.val = 20000)
head(x5)
```

```

# Example 6: Getting the IDs of catchments with area less than or equal to
# 20,000 km2 and the corresponding area values for each selected catchment
x6 <- getCatchments(attr = "area_km2", op = "<=", attr.val = 20000,
                    include.values = TRUE)

head(x6)

# Example 7: Getting the IDs of catchments with area greater than or equal to
# 20,000 km2 but less than 25,000 km2
x7 <- getCatchments(attr = "area_km2", op = "between", attr.val = 20000,
                    attr.val2 = 25000, include.lower = TRUE,
                    include.upper = FALSE)

head(x7)

# Example 8: Getting the IDs of catchments with area greater than 20,000 km2
# but less than or equal to 25,000 km2, and the monthly time series of
# hydrometeorological variables
x8 <- getCatchments(attr = "area_km2", op = "between", attr.val = 20000,
                    attr.val2 = 25000, include.lower = FALSE,
                    include.upper = TRUE, get.data = TRUE)

str(x8, 2)

# Example 9: Getting the IDs of catchments with area greater than 20,000 km2
# but less than or equal to 25,000 km2, the corresponding area values for
# each selected catchment, and the daily time series of hydrometeorological
# variables
x9 <- getCatchments(attr = "area_km2", op = "between", attr.val = 20000,
                    attr.val2 = 25000, include.lower = FALSE,
                    include.upper = TRUE, get.data = TRUE,
                    include.values = TRUE)

str(x9, 2)

```

getData

getData

Description

Download data for a specific catchment from the CAMELS-CL dataset.

Usage

```

getData(x, tscale, from, to, na.rm, na.rm.max, include.shp, include.attr,
        include.extra.P, verbose, req.verbosity)

## Default S3 method:
getData(x, tscale=c("daily", "monthly", "annual"),
        from="auto", to="auto", na.rm=TRUE, na.rm.max=0, include.shp=FALSE,
        include.attr=FALSE, include.extra.P=FALSE, verbose=TRUE,
        req.verbosity=0)

```

Arguments

x	Numeric, indicating the ID of the catchment for which all the data will be downloaded.
tscale	Character, indicating the temporal scale to be used for the output time series. Valid values of tscale are: <ul style="list-style-type: none"> • "daily": All the output time series have a daily temporal frequency, directly read from the CAMELS-CL dataset. • "monthly": All the output time series have a monthly temporal frequency, aggregated from the original daily time series in the CAMELS-CL dataset. • "annual": All the output time series have an annual temporal frequency, aggregated from the original daily time series in the CAMELS-CL dataset.
from	Date, indicating the starting date of the time series to be returned. Its default value is "auto", which automatically selects the first date with at least one non-missing value among all variables.
to	Date, indicating the ending date of the time series to be returned. Its default value is "auto", which automatically selects the last date with at least one non-missing value among all variables.
na.rm	Logical. Should missing values be removed? <ul style="list-style-type: none"> • TRUE : the monthly/annual values are computed only for months/years with a percentage of missing values less than na.rm.max • FALSE: if there is AT LEAST one NA within a month/year, the corresponding monthly/annual values in the output object will be NA.
na.rm.max	Numeric in [0, 1]. It is used to define the maximum percentage of missing values allowed in each month/year to keep the temporally aggregated value (i.e., monthly/annual) in the output object of this function. In other words, if the percentage of missing values in a given month/year is larger or equal than na.rm.max the corresponding monthly or annual value will be NA.
include.shp	Optional. Logical, indicating whether to include a shapefile of the catchment in the function's output. Its default value is FALSE.
include.attr	Optional. Logical, indicating whether to include all the attributes of the catchment in the function's output. Its default value is FALSE.
include.extra.P	Optional. Logical, indicating whether to include all the additional precipitation datasets available in the CAMELS-CL database (i.e., MSWEP, CHIRPSv2, and TMPA). Its default value is FALSE.
verbose	Logical, if TRUE, progress messages are printed.
req.verbosity	Integer controlling how much HTTP request information is displayed. Possible values are: <ul style="list-style-type: none"> • 0: no output • 1: headers • 2: headers and bodies • 3: headers, bodies, and curl status information

Details

Currently, this function works with CAMELS-CL version "2022 enero".

Value

The daily time series output contains a single zoo object with the following columns:

- 1) P_mm: precipitation obtained from the CR2MET dataset, [mm/day].
- 2) Tmin_degC: minimum air temperature obtained from CR2METv2 dataset, [degC].
- 3) Tavg_degC: average air temperature obtained from CR2METv2 dataset, [degC].
- 4) Tmax_degC: maximum air temperature obtained from CR2METv2 dataset, [degC].
- 5) PET_mm: potential evapotranspiration computed with the Hargreaves-Samani equation, [mm/day].
- 6) Qobs_mm: streamflow obtained from 'Direccion General de Aguas (DGA)', [mm/day].
- 7) Qobs_m3s: streamflow obtained from 'Direccion General de Aguas (DGA)', [m3/s].

If include.extra.P = TRUE, the following columns will also be included:

- 8) P_CHIRPSv2_mm: precipitation obtained from Climate Hazards Group InfraRed Precipitation with Station (CHIRPS) version 2 dataset, [mm/day].
- 9) P_MSWEP_mm: precipitation obtained from Multi-Source Weighted-Ensemble Precipitation (MSWEP) v1.1 dataset, [mm/day].
- 10) P_TMPA_mm: precipitation obtained from TMPA 3B42v7 dataset, [mm/day].

Author(s)

Hector Garces-Figueroa <hegarcesf@gmail.com> and Mauricio Zambrano-Bigiarini <mzb.devel@gmail.com>

References

<https://camels.cr2.cl>

Alvarez-Garreton, C.; Mendoza, P. A.; Boisier, J. P.; Addor, N.; Galleguillos, M.; Zambrano-Bigiarini, M.; Lara, A.; Puelma, C.; Cortes, G.; Garreaud, R.; McPhee, J.; and Ayala, A. (2018). *The CAMELS-CL dataset: catchment attributes and meteorology for large sample studies - Chile dataset*, *Hydrology and Earth System Sciences* 22, 5817-5846. doi:10.5194/hess-22-5817-2018.

Examples

```
#####

# Example 1: Getting only daily time series data for the 'Rio Trancura Antes Rio Llafenco'
# catchment
x1 <- getData(9414001)
head(x1)

# Example 2: Getting only daily time series of P [mm/day], Tmin [degC], Tavg [degC],
# Tmax [degC], PET [mm/day] and Q [mm/day] data for the 'Rio Trancura Antes Rio Llafenco'
# catchment
var.names <- c("P_mm_day", "Tmin_degC_day", "Tavg_degC_day",
               "Tmax_degC_day", "PET_mm_day", "Qobs_mm_day")
```

```
x2 <- getData(9414001)
x2 <- x2[, var.names]
names(x2) <- c("P", "Tmin", "Tavg", "Tmax", "PET", "Q")

# Example 3: Getting only daily time series and catchment borders data for the
# 'Rio Trancura Antes Rio Llafenco' catchment
x3 <- getData(9414001, include.shp=TRUE)
terra::plot(x3[["CatchmentBorders"]])

# Example 4: Getting only daily time series and catchment attributes data for the
# 'Rio Trancura Antes Rio Llafenco' catchment
x4 <- getData(9414001, include.attr=TRUE)
head(x4[["CatchmentAttributes"]])

# Example 5: Getting daily time series, catchment borders and catchment attributes data for the
# 'Rio Trancura Antes Rio Llafenco' catchment
x5 <- getData(9414001, include.shp=TRUE, include.attr=TRUE)
head(x5[["CatchmentTS"]])
terra::plot(x5[["CatchmentBorders"]])
head(x5[["CatchmentAttributes"]])

# Example 6: Getting daily time series, catchment borders, catchment attributes and additional
# precipitation datasets for the 'Rio Trancura Antes Rio Llafenco' catchment
x6 <- getData(9414001, include.shp=TRUE, include.attr=TRUE,
             include.extra.P=TRUE)
head(x6[["CatchmentTS"]])
terra::plot(x6[["CatchmentBorders"]])
head(x6[["CatchmentAttributes"]])

# Example 7: Getting only monthly time series data for the 'Rio Trancura Antes Rio Llafenco'
# catchment
x7 <- getData(9414001, tscale="monthly")
head(x7)

# Example 8: Getting only annual time series data for the 'Rio Trancura Antes Rio Llafenco'
# catchment
x8 <- getData(9414001, tscale="annual")
head(x8)

# Example 9: Getting only annual time series data between 2008 and 2015 for the
# 'Rio Trancura Antes Rio Llafenco' catchment
x9 <- getData(9414001, tscale="annual", from = as.Date("2008-01-01"),
             to = as.Date("2015-12-31"))
head(x9)

# Example 10: Getting only monthly time series data for the 'Rio Mapocho en Los Almendros'
# and 'Rio Trancura Antes Rio Llafenco' catchments

target.ids <- c(5722002, 9414001)
nelements <- length(target.ids)
x10 <- vector(mode = "list", length = nelements)

for (i in seq_len(nelements)) {
```

```
x10[[i]] <- getData(target.ids[[i]], tscale="monthly")
}
str(x10)
```

Index

* **data**

getCatchments, [2](#)

getData, [4](#)

getCatchments, [2](#)

getData, [4](#)