Package 'RFmerge'

May 8, 2026

Type Package

Title Merging of Satellite Datasets with Ground Observations using Random Forests

Version 0.3-3

Maintainer Mauricio Zambrano-Bigiarini <mzb.devel@gmail.com>

Description S3 implementation of the Random Forest MErging Procedure (RF-MEP), which combines two or more satellite-based datasets (e.g., precipitation products, topography) with ground observations to produce a new dataset with improved spatio-temporal distribution of the target field. In particular, this package was developed to merge different Satellite-based Rainfall Estimates (SREs) with measurements from rain gauges, in order to obtain a new precipitation dataset where the time series in the rain gauges are used to correct different types of errors present in the SREs. However, this package might be used to merge other hydrological/environmental gridded datasets with point observations. For details, see Baez-Villanueva et al. (2020) <doi:10.1016/j.rse.2019.111606>. Bugs / comments / questions / collaboration of any kind are very welcomed.

License GPL (>= 3)

Depends R (>= 3.5.0)

Imports terra, randomForest, zoo, parallel, methods, stats, utils, pbapply

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL http://mzb.cl/RFmerge/, https://github.com/hzambran/RFmerge

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

BugReports https://github.com/hzambran/RFmerge/issues

LazyLoad yes

NeedsCompilation no

Repository CRAN

Config/Needs/website rmarkdown

Author Mauricio Zambrano-Bigiarini [aut, cre, cph] (ORCID:
<https://orcid.org/0000-0002-9536-643X>),
Oscar M. Baez-Villanueva [aut, cph] (ORCID:
<https://orcid.org/0000-0002-2262-1698>),
Juan Giraldo-Osorio [ctb] (ORCID:
<https://orcid.org/0000-0001-6205-3341>)

Date/Publication 2026-05-08 07:31:59 UTC

Contents

RFmerge-package 2
RFmerge 4
ValparaisoPPgis 8
ValparaisoPPts 9

Index **10**

`RFmerge-package` *Merging of Satellite Datasets with Ground Observations using Random Forests*

Description

S3 implementation of the Random Forest MErging Procedure (RF-MEP), which combines two or more satellite-based datasets (e.g., precipitation products, topography) with ground observations to produce a new dataset with improved spatio-temporal distribution of the target field. In particular, this package was developed to merge different Satellite-based Rainfall Estimates (SREs) with measurements from rain gauges, in order to obtain a new precipitation dataset where the time series in the rain gauges are used to correct different types of errors present in the SREs. However, this package might be used to merge other hydrological/environmental satellite fields with point observations. For details, see Baez-Villanueva et al. (2020) <doi:10.1016/j.rse.2019.111606>. Bugs / comments / questions / collaboration of any kind are very welcomed.

Details

Package:	RFmerge
Type:	Package
Version:	0.3-3
Date:	2026-05-07
License:	GPL >= 3
LazyLoad:	yes
Packaged:	Thu May 7 09:41:19 -04 2026 ; MZB
BuiltUnder:	R version 4.6.0 (2026-04-24) – "Because it was There" ; aarch64-apple-darwin23

Author(s)

Mauricio Zambrano-Bigiarini, Oscar M. Baez-Villanueva

Maintainer: Mauricio Zambrano-Bigiarini <mzb.devel@gmail>

References

Baez-Villanueva, O. M.; Zambrano-Bigiarini, M.; Beck, H.; McNamara, I.; Ribbe, L.; Nauditt, A.; Birkel, C.; Verbist, K.; Giraldo-Osorio, J.D.; Thinh, N.X. (2020). RF-MEP: a novel Random Forest method for merging gridded precipitation products and ground-based measurements, Remote Sensing of Environment, 239, 111610. doi:10.1016/j.rse.2019.111606. <https://doi.org/10.1016/j.rse.2019.111606>.

Hengl, T., Nussbaum, M., Wright, M. N., Heuvelink, G. B., & Gräler, B. (2018). Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. PeerJ, 6, e5518. doi:10.7717/peerj.5518.

See Also

https://cran.r-project.org/package=terra.
https://cran.r-project.org/package=hydroGOF.

Examples

```
library(terra)

data(ValparaisoPPts)
data(ValparaisoPPgis)

ValparaisoSHP.fname <- system.file("extdata/ValparaisoSHP.shp",package="RFmerge")
ValparaisoSHP       <- terra::vect(ValparaisoSHP.fname)

chirps.fname   <- system.file("extdata/CHIRPS5km.tif",package="RFmerge")
prsnncdr.fname <- system.file("extdata/PERSIANNcdr5km.tif",package="RFmerge")
dem.fname      <- system.file("extdata/ValparaisoDEM5km.tif",package="RFmerge")

CHIRPS5km        <- rast(chirps.fname)
PERSIANNcdr5km   <- rast(prsnncdr.fname)
ValparaisoDEM5km <- rast(dem.fname)

covariates <- list(chirps=CHIRPS5km, persianncdr=PERSIANNcdr5km,
                   dem=ValparaisoDEM5km)



# The following code assumes that the region is small enough for neglecting
# the impact of computing Euclidean distances in geographical coordinates.
# If this is not the case, please read the vignette 'Tutorial for merging
# satellite-based precipitation datasets with ground observations using RFmerge'

# without using parallelisation
```

```
rfmep <- RFmerge(x=ValparaisoPPts, metadata=ValparaisoPPgis, cov=covariates,
                 id="Code", lat="lat", lon="lon", mask=ValparaisoSHP, training=1)

# Detecting if your OS is Windows or GNU/Linux,
# and setting the 'parallel' argument accordingly:
onWin <- ( (R.version$os=="mingw32") | (R.version$os=="mingw64") )
ifelse(onWin, parallel <- "parallelWin", parallel <- "parallel")

#Using parallelisation, with a maximum number of nodes/cores to be used equal to 2:
par.nnodes <- min(parallel::detectCores()-1, 2)
rfmep <- RFmerge(x=ValparaisoPPts, metadata=ValparaisoPPgis, cov=covariates,
                  id="Code", lat="lat", lon="lon", mask=ValparaisoSHP,
                  training=0.8, parallel=parallel, par.nnodes=par.nnodes)
```

RFmerge *Merging of satellite datasets with ground observations using Random Forests (RF)*

Description

Merging of satellite datasets with ground observations using Random Forests (RF)

Usage

```
RFmerge(x, ...)

## Default S3 method:
RFmerge(x, metadata, cov, mask, training,
        id="id", lat = "lat", lon = "lon",
        ED = TRUE, rasterizedED=FALSE,
        seed = NULL, ntree = 2000, na.action = stats::na.omit,
        parallel=c("none", "parallel", "parallelWin"),
        par.nnodes=parallel::detectCores()-1,
        par.pkgs= c("terra", "randomForest", "zoo"), write2disk=FALSE,
        drty.out, use.pb=TRUE, verbose=TRUE,...)

## S3 method for class 'zoo'
RFmerge(x, metadata, cov, mask, training,
        id="id", lat = "lat", lon = "lon",
        ED = TRUE, rasterizedED=FALSE,
        seed = NULL, ntree = 2000, na.action = stats::na.omit,
        parallel=c("none", "parallel", "parallelWin"),
        par.nnodes=parallel::detectCores()-1,
        par.pkgs= c("terra", "randomForest", "zoo"), write2disk=FALSE,
        drty.out, use.pb=TRUE, verbose=TRUE, ...)
```

Arguments

x	data.frame with the ground-based values that will be used as the dependent variable to train the RF model. Every column must represent one ground-based station and the codes of the stations must be provided as colnames. class(data) must be zoo.
metadata	data.frame with the metadata of the ground-based stations. At least, it MUST have the following 3 columns: -) id: This column stores the unique identifier (ID) of each ground-based observation. Default value is "id". -) lat: This column stores the latitude of each ground observation. Default value is "lat". -) lon: This column stores the longitude of each ground observation. Default value is "lon".
cov	List with all the covariates used as independent variables in the Random Forest model. The individual covariates must be SpatRaster objects, either when they vary in time (e.g., individual gridded precipitation datasets) or does not vary in time (e.g., a digital elevation model). All time-varying covariates in cov MUST have the same number of layers (bands), which is internally checked using the , nlyr function. Covariates that do not change in time (e.g., a DEM) are internally transformed into SpatRaster objects with the same number of layers as the other time-varying elements in cov.
mask	OPTIONAL. If provided, the final merged product mask out all values in cov outside mask. Spatial object (vectorial) with the spatial borders of the study area (e.g., catchment, administrative borders). class(mask) must be a SpatVect object with "POLYGON" or "MULTIPOLYGON" geometry.
training	Numeric indicating the percentage of stations that will be used in the training set. The valid range is from zero to one. If training = 1, all the stations will be used for training purposes.
id	Character, with the name of the column in metadata where the identification code (ID) of each station is stored.
lat	Character, with the name of the column in metadata where the latitude of the stations is stored.
lon	Character, with the name of the column in metadata where the longitude of the stations is stored.
ED	logical, should the Euclidean distances be computed an used as covariates in the random forest model?. The default value is TRUE.
rasterizedED	logical, should the Euclidean distances between stations and grid cells be computed to the actual point coordinate (rasterizedED=FALSE) or to the rasterized station cell center (rasterizedED=TRUE). When rasterizedED=FALSE, the the self-distance between the station and its own grid cell is usually not zero. By default, rasterizedED=FALSE.
seed	Numeric, indicating a single value, interpreted as an integer, or null.

`parallel` character, indicates how to parallelise 'RFmerge' (to be precise, only the evaluation of the objective function `fn` is parallelised). Valid values are:

-)`none`: no parallelisation is made (this is the default value)

-)`parallel`: parallel computations for network clusters or machines with multiple cores or CPUs. A 'FORK' cluster is created with the `makeForkCluster` function.

-)`parallelWin`: parallel computations for network clusters or machines with multiple cores or CPUs (this is the only parallel implementation that works on Windows machines). A 'PSOCK' cluster is created with the `makeCluster` function.

`par.nnodes` OPTIONAL. Used only when `parallel!='none'`

numeric, indicates the number of cores/CPUs to be used in the local multi-core machine, or the number of nodes to be used in the network cluster.

By default `par.nnodes` is set to the amount of cores detected by the function `detectCores()` (**parallel** package)

`par.pkgs` OPTIONAL. Used only when `parallel='parallelWin'`

list of package names (as characters) that need to be loaded on each node for allowing the objective function `fn` to be evaluated. By default `c("terra", "randomForest", "zoo")`.

`ntree` Numeric indicating the maximum number trees to grow in the Random Forest algorithm. The default value is set to 2000. This should not be set to too small a number, to ensure that every input row gets predicted at least a few times. If this value is too low, the prediction may be biased.

`na.action` A function to specify the action to be taken if NAs are found. (NOTE: If given, this argument must be named.)

`write2disk` logical, indicates if the output merged SpatRaster layers and the training and evaluation datasets (two files each, one with time series and other with metadata) will be written to the disk. By default `write2disk=FALSE`

`drty.out` Character with the full path to the directory where the final merged product will be exported as well as the training and evaluation datasets. Only used when `write2disk=TRUE`

`use.pb` logical, indicates if a progress bar should be used to show the progress of the random forest computations (it might reduce a bit the performance of the computations, but it is useful to track if everything is working well). By default `use.pb=TRUE`

`verbose` logical, indicates if progress messages are to be printed. By default `verbose=TRUE`

`...` further arguments to be passed to the low level function `randomForest.default`.

Value

It returns a SpatRaster object with as many layers as time steps are present in x. Each one of the layers in the output object has the same spatial resolution and spatial extent as the cov argument.

Author(s)

Oscar M. Baez-Villanueva, <oscar.baezvillanueva@ugent.be>
Mauricio Zambrano-Bigiarini, <mzb.devel@gmail>
Juan D. Giraldo-Osorio, <j.giraldoo@javeriana.edu.co>

References

Baez-Villanueva, O. M.; Zambrano-Bigiarini, M.; Beck, H.; McNamara, I.; Ribbe, L.; Nauditt, A.; Birkel, C.; Verbist, K.; Giraldo-Osorio, J.D.; Thinh, N.X. (2020). RF-MEP: a novel Random Forest method for merging gridded precipitation products and ground-based measurements. *Remote Sensing of Environment*, 239, 111610. doi:10.1016/j.rse.2019.111606.

Hengl, T.; Nussbaum, M.; Wright, M. N.; Heuvelink, G. B.; Gräler, B. (2018). Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. *PeerJ*, 6, e5518. doi:10.7717/peerj.5518.

See Also

terra, rast, nlyr, resample, rotate, crop.

Examples

```
library(terra)

data(ValparaisoPPts)
data(ValparaisoPPgis)

ValparaisoSHP.fname <- system.file("extdata/ValparaisoSHP.shp",package="RFmerge")
ValparaisoSHP       <- terra::vect(ValparaisoSHP.fname)

chirps.fname   <- system.file("extdata/CHIRPS5km.tif",package="RFmerge")
prsnncdr.fname <- system.file("extdata/PERSIANNcdr5km.tif",package="RFmerge")
dem.fname      <- system.file("extdata/ValparaisoDEM5km.tif",package="RFmerge")

CHIRPS5km        <- rast(chirps.fname)
PERSIANNcdr5km   <- rast(prsnncdr.fname)
ValparaisoDEM5km <- rast(dem.fname)

covariates <- list(chirps=CHIRPS5km, persianncdr=PERSIANNcdr5km,
                   dem=ValparaisoDEM5km)



# The following code assumes that the region is small enough for neglecting
# the impact of computing Euclidean distances in geographical coordinates.
```

```
# If this is not the case, please read the vignette 'Tutorial for merging
# satellite-based precipitation datasets with ground observations using RFmerge'


# without using parallelisation
rfmep <- RFmerge(x=ValparaisoPPts, metadata=ValparaisoPPgis, cov=covariates,
                 id="Code", lat="lat", lon="lon", mask=ValparaisoSHP, training=1)


# Detecting if your OS is Windows or GNU/Linux,
# and setting the 'parallel' argument accordingly:
onWin <- ( (R.version$os=="mingw32") | (R.version$os=="mingw64") )
ifelse(onWin, parallel <- "parallelWin", parallel <- "parallel")

#Using parallelisation, with a maximum number of nodes/cores to be used equal to 2:
par.nnodes <- min(parallel::detectCores()-1, 2)
rfmep <- RFmerge(x=ValparaisoPPts, metadata=ValparaisoPPgis, cov=covariates,
                  id="Code", lat="lat", lon="lon", mask=ValparaisoSHP,
                  training=0.8, parallel=parallel, par.nnodes=par.nnodes)
```

`ValparaisoPPgis` *Spatial location of rain gauges in the Valparaiso region (Chile)*

Description

Spatial location of the 34 rain gauges with daily precipitation for the Valparaiso region (dataset *'ValparaisoPPts'*), Chile, with more than 70% of days with information (without missing values)

Usage

```
data(ValparaisoPPgis)
```

Format

A data.frame with seven fields:
*) 'ID : identifier of each gauging station.
*) 'STATION_NAME' : name of the gauging station.
*) 'lon' : easting coordinate of the gauging station, EPSG:4326.
*) 'lat' : northing coordinate of the gauging station, EPSG:4326.
*) 'ELEVATION' : elevation of the gauging station, [m a.s.l.].
*) 'BASIN_ID' : identifier of the subbasin in which the gauging station s located.
*) 'BASIN_NAME' : name of the subbasin in which the gauging station s located.

Details

Projection: EPSG:4326

Source

Downloaded ('Red de Control Meteorologico') from the web site of the Confederacion Hidrografica del Ebro (CHE) http://www.chebro.es/ (original link http://oph.chebro.es/ContenidoCartoClimatologia.htm, last accessed [March 2008]), and then the name of 7 selected fields were translated into English language.

These data are intended to be used for research purposes only, being distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY.

`ValparaisoPPts` *Daily Precipitation Time Series for Valparaiso Region (Chile)*

Description

Daily time series for the year 1983 on 34 rain gauges of the Valparaiso region (Chile), with more than 90% of days with information (without missing values)

Usage

```
data(ValparaisoPPts)
```

Format

A zoo object with 34 columns (one for each rain gauge) and 365 rows (one for each day in 1983). `colnames(ValparaisoPPts)` must coincide with the *ID* column in *ValparaisoPPgis*.

Details

Daily time series of ground-based daily precipitation for 1900-2018 were downloaded from a dataset of 816 rain gauges from the Center of Climate and Resilience Research (CR2; https://www.cr2.cl/datos-de-precipitacion/).
The 34 stations in this dataset wer selected because they had less than 10

Source

The CR2 dataset unifies individual datasets provided by Dirección General de Aguas (DGA) and Dirección Meteorológica de Chile (DMC), the Chilean water and meteorological agencies, respectively.
These data are intended to be used for research purposes only, being distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY.

Index

∗ **datasets**
ValparaisoPPgis, 8
ValparaisoPPts, 9
∗ **manip**
RFmerge, 4
∗ **package**
RFmerge-package, 2

crop, *7*

makeCluster, *6*
makeForkCluster, *6*

nlyr, *5*, *7*

rast, *7*
resample, *7*
RFmerge, 4
RFmerge-package, 2
rotate, *7*

terra, *7*

ValparaisoPPgis, 8
ValparaisoPPts, 9