Package: usefulFuns (via r-universe)

August 28, 2024

Type Package

Title Useful functions for my modules and packages

Version 0.1.1

Author Tati Micheletti

Maintainer Tati Micheletti <`tati.micheletti@gmail.com`>

Description A few functions and wrappers around useful code.

License GPL (>=2)

Encoding UTF-8

Language en-CA

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Depends R (>= 3.6)

Imports animation, crayon, data.table, effsize, fasterize, future,
    future.apply, ggplot2, googledrive, gridExtra, hexSticker,
    LandR, magrittr, methods, pryr, quickPlot, raster, rasterVis,
    RColorBrewer, reproducible, Require, sp, sf, showtext,
    SpaDES.core, SpaDES.tools, stats, sysfonts

Suggests covr, lintr, remotes, spelling

Remotes achubaty/amc@development, PredictiveEcology/LandR@development,
    PredictiveEcology/reproducible@development

Collate 'RSFplot.R' 'avrgTimeComparison.R'
    'biomassPerSpeciesYearGIF.R' 'biomassYearGIF.R'
    'cbindFromList.R' 'substrBoth.R' 'grepMulti.R' 'helpersBirds.R'
    'bootstrapPercentChanges.R' 'bringObjectTS.R' 'burnFromAge.R'
    'calculatePixelsInaRule.R' 'changeTraits.R'
    'checkPackagesVersions.R' 'classifyWetlands.R'
    'nameAndBringOn.R' 'createDynamicLayersRSF.R'
    'createEquation.R' 'createModObject.R' 'createModels.R'
    'createShrubHerbLayers.R' 'createStaticLayersRSF.R'
    'defineStudyArea.R' 'extractDisturbanceFast.R'

'forestAgePlot.R' 'getLayers.R' 'makeReclassifyMatrix.R'
'meanValuesTime.R' 'moduleSticker.R' 'plotBurnSummary.R'
'plotLeadingVegetationType.R' 'plotVegetationBiomass.R'
'prepInputStack.R' 'prepInputsLayers_DUCKS.R'
'prepareClimateLayers.R' 'provinceBCRStudyArea.R'
'reviseSpeciesTraits.R' 'subsetNonNARas.R'
'totalBiomassPerSpecies.R' 'usefulFuns-package.R'

Repository https://predictiveecology.r-universe.dev

RemoteUrl https://github.com/PredictiveEcology/usefulFuns

RemoteRef development

RemoteSha bc28411fd29cfc385e1bc08fc39b85d222ca4d8c

Contents

Index

---

usefulFuns-package   usefulFuns *package*

Description

A few functions and wrappers around useful code.

---

avrgTimeComparison   *Compare two or more scenarios for averages through time*

Description

Compare two or more scenarios for averages through time

Usage

```
avrgTimeComparison(
  ...,
  upload,
  outputFolder,
  comparisonID,
  folderID,
  plotCI = TRUE
)
```

Arguments

...	List or tables of results coming from meanValuesTime to be compared.
upload	logical. Should the raster be uploaded to Google Drive? Only works if the raster exists OR is set to be written.
outputFolder	character. Path to the folder where it should be saved.
comparisonID	character. Name to identify the comparison (for file name)
folderID	character. Google folder id to upload to. Only needs to be provided if upload == TRUE. Default is 'NULL"
plotCI	logical. Should the plot have confidence interval?

Value

list of significant species or scenarios with indication of increasing or decreasing

Author(s)

Tati Micheletti

---

biomassPerSpeciesYearGIF
                          Creates a GIF of biomass change

Description

Creates a GIF of biomass change

Usage

```
biomassPerSpeciesYearGIF(dataPath, years = NULL, uploadTo)
```

Arguments

dataPath	character. Path to data
years	numeric. Years available/intended to be used for the animated gif.
uploadTo	character. Google drive folder id. (i.e. "1ZqPVs33HxnnmjLUW94i7AuwAS-nloPGH")

Value

list of plots

Author(s)

Tati Micheletti

---

biomassYearGIF   *Creates a GIF of total biomass change*

Description

Creates a GIF of total biomass change

Usage

```
biomassYearGIF(dataPath, years = NULL, uploadTo)
```

Arguments

dataPath	character. Path to data.
years	numeric. Years available/intended to be used for the animated gif.
uploadTo	character. Google drive folder id. (i.e. "1ZqPVs33HxnnmjLUW94i7AuwAS-nloPGH")

Value

RasterStack

Author(s)

Tati Micheletti

---

bootstrapPercentChanges

Bootstrap rasters for testing significance on comparable rasters of different species or scenarios

---

Description

Bootstrap rasters for testing significance on comparable rasters of different species or scenarios

Usage

```
bootstrapPercentChanges(
  dataPath,
  years = c(2001, 2100),
  sampleSize = "auto",
  n = 100,
  shp = NULL,
  species = NULL,
  useFuture = FALSE
)
```

Arguments

dataPath	character. Path to raster data.
years	numeric. Years to compare. Currently this function only compares 2 years.
sampleSize	numeric or "auto" (default). What is the sample size (i.e. number of pixels) we want to use on the bootstrapping? If "auto" it calculates internally Cohen's D And Hedges G Effect Size.
n	numeric. Default is 100. How many iterations (random selection of sampleSize pixels) should be done?
shp	character or shapefile. If you wish to calculate these metrics for separate areas. Needs to match the rasters. Default is NULL (i.e. the whole raster is only one area)

species   character. Default is NULL. Which species should this function be ran ?

useFuture   logical. Should use **future** to parallelize? Requires future and future_apply
                 packages.

Value

list of significant species or scenarios with indication of increasing or decreasing

Author(s)

Tati Micheletti

---

bringObjectTS   *Bring objects that represent time series into a list*

Description

Bring objects that represent time series into a list

Usage

```
bringObjectTS(path, rastersNamePattern)
```

Arguments

path   path to the directory containing the files to be brought as list.

rastersNamePattern
                 Character vector of the name pattern to be used for the search.

Value

A list of the objects that were read from disk. Currently the function only works with .tif (i.e.
raster) and .rds objects.

Author(s)

Tati Micheletti

---

calculatePixelsInaRule

Calculate pixels in a rule

Description

Uses a set of rules to calculate how many non-NA pixels in a raster follow those rules, using a shapefile to subset those of interest

Usage

```
calculatePixelsInaRule(ras, rule, pol, shp, ...)
```

Arguments

ras	RasterLayer to calculate the number of pixels that follow a given rule
rule	character string of rule to determine which pixels should be computed
pol	numeric. Representation of the polygon of interest in pol for the calculation
shp	shapefile indicating the different areas to subset the pixels to consider in the calculation
...	Additional arguments

Value

A list of the percent disturbance, total pixels not NA and how many pixels are disturbed according to the rules.

Author(s)

Tati Micheletti

---

cbindFromList *Column binds a list of* data.frame*s or* data.table*s*

Description

Column binds a list of data.frames or data.tables

Usage

```
cbindFromList(lst)
```

Arguments

lst	List of data.tables or data.frames to column bind.

Value

Returns a merged `data.table` with unique columns

Author(s)

Tati Micheletti

---

changeTraits	*Provides an easy way of modifying species traits table in* LandR *by multiplication of original traits by a specified factor*

Description

Provides an easy way of modifying species traits table in `LandR` by multiplication of original traits by a specified factor

Usage

```
changeTraits(speciesTable, param, facMult, species)
```

Arguments

speciesTable	`speciesTable` object (`data.table`) from `LandR`. Contains different traits in columns and a row for each species.
param	Character vector of the trait to be modified.
facMult	numeric. Factor by which the specified trait should be multiplied.
species	character string of the Species for which the trait should be modified

Value

A `data.table` updated with the new parameters.

Author(s)

Tati Micheletti

---

checkPackagesVersions   *All packages' versioning*

Description

Returns a data.table of versions of all packages attached to your current R session, or used in a simulation - including git commit if a package was installed from e.g., GitHub.

Usage

```
checkPackagesVersions(simList = NULL, filePath = NULL)
```

Arguments

simList	simList object resulting from a SpaDES.core::spades() call. Default NULL.
filePath	Character string. If passed, the table is written to the specified file path, as an RDS file. Default NULL.

Details

If you don't provide it, the function will return the information regarding your current R session's information. When running a simulation, it is automatically be created by the spades() and attached as the invisible object .packagesVersions.

Author(s)

Tati Micheletti

---

classifyWetlands   *Classify wetlands (really!) using the wetlands layer set as input and a either LCC05 or LCC2010*

Description

Classify wetlands (really!) using the wetlands layer set as input and a either LCC05 or LCC2010

Usage

```
classifyWetlands(
  LCC,
  wetLayerInput,
  pathData,
  studyArea = NULL,
  RasterToMatch = NULL
)
```

Arguments

`LCC`	numeric. 2005 (250 m resolution) or 2010 (30 m resolution) landcover rasters.
`wetLayerInput`	Which wetland should be used as input (raster with projection). It was originally designed to work with the DUCKS Unlimited Waterland layer (30 m) but can work with any water layers that have the following code: `possibleLakes == 0`, `water bodies == 1`, `wetlands == 2`, `uplands > 2`.
`pathData`	Where the layers are stored and/or should be saved to
`studyArea`	If the layer should be cropped and masked after classification. Optional.
`RasterToMatch`	raster to match the new layer after classification to. Optional.

Value

As with `archivist::cache`, returns the value of the function call or the cached version (i.e., the result from a previous call to this same cached function with identical arguments).

Author(s)

Tati Micheletti

`createDynamicLayersRSF`

Create the necessary layers for predictive caribou RSF module.

Description

Intended for use within a SpaDES module.

Usage

```
createDynamicLayersRSF(
  ageMap,
  biomassMap,
  biomassMapName,
  oldBurnTime,
  oldBurnName,
  newBurnName,
  roadDensity,
  roadDensityName,
  waterRaster,
  waterRasterName,
  RTM
)

createStaticLayersRSF(
  elevation,
```

```
    vrug,
    LCC,
    shrubName,
    herbName,
    elevationName,
    vrugName,
    reclassLCC05,
    dynamicLayers,
    RTM,
    destinationPath
)
```

Arguments

`ageMap`	RasterLayer of forest age.
`biomassMap`	RasterLayerof forest biomass.
`biomassMapName`	character. Name of the forest biomass layer.
`oldBurnTime`	numeric. Definition of the initial interval considered to be old burn. The end of this time is 20 years later (i.e. 40-60 years).
`oldBurnName`	character. Name of the old burn layer in the model.
`newBurnName`	character. Name of the new burn layer in the model.
`roadDensity`	Anthropogenic disturbance (raster) layer. Currently, road density layer used for both RSF and demographic models.
`roadDensityName`	
	character. Name of the road Density layer in the model.
`waterRaster`	RasterLayer indicating water bodies.
`waterRasterName`	
	character. Name of the water layer in the model.
`RTM`	RasterLayer template for these layers to match.
`elevation`	RasterLayer of elevation
`vrug`	RasterLayer of ruggedness
`LCC`	RasterLayer of landcover classes 2005
`shrubName`	character. Name of the shrub layer in the model.
`herbName`	character. Name of the herb layer in the model.
`elevationName`	character. Name of the elevation layer in the model.
`vrugName`	character. Name of the vrug layer in the model.
`reclassLCC05`	List with reclassification for LCC05 values (i.e. LCC05 classes that should be classified as shrub or herbs)
`dynamicLayers`	RasterStack to be stacked with the `staticLayers` created in the current function to be passed for the model prediction.
`destinationPath`	
	TODO

Value

`RasterStack` of layers

`RasterStack`

Author(s)

Tati Micheletti

---

createEquation | *Write an equation based on a* `data.frame` *(or* `data.table`*), bootstrapping a set of covariates around the standard error present in the table.*

Description

Write an equation based on a `data.frame` (or `data.table`), bootstrapping a set of covariates around the standard error present in the table.

Usage

```
createEquation(model, replicates = 100)
```

Arguments

model   `data.frame`with columns `Coefficient`, and `Value` and `StdErr` for a given parameter. The model table NEEDS to have at least two rows, one being the `Intercept` as `Coefficient`. If `Intercept` is not found, it enters `browser()` mode.

replicates   numeric. Number of repetitions to be added to the bootstrapping.

Value

As with `archivist::cache`, returns the value of the function call or the cached version (i.e., the result from a previous call to this same cached function with identical arguments).

Author(s)

Tati Micheletti

---

createModels	*Create caribou models based on 2011 ECCC report on population demographics and RSF.*

Description

Uses `createEquation` internally.

Usage

```
createModels(caribouCoefTable = NULL, modelsToUse = "M3")
```

Arguments

caribouCoefTable

> `data.table` of the coefficient values for all models available. If not provided, will use as default the one from ECCC 2011 for population demographics.

modelsToUse   Character string of the model to be used. Currently only "M3" and "M7" (demographics) and "TaigaPlains" (RSF) have been implemented. Default "M3".

Value

named list of the quote of the model that can be parsed to generate the simulated data bootstrapped using the model coefficients.

Author(s)

Tati Micheletti

---

createModObject	*Create a simulation model object*

Description

Check a `simList` for a specific object at a specific time, or searches for it in an input folder (i.e. saved outputs). Simulates the existence of a `simList` with specific objects in time `t`.

Usage

```
createModObject(data, sim = NULL, pathInput, currentTime, fun = readRDS)
```

Arguments

`data`	character string of the object or file name to be searched for
`sim`	`simList` object. Default `NULL`.
`pathInput`	path to the directory containing the files to be searched for.
`currentTime`	Numeric. Current time to be used to search for the object.
`fun`	function to be used to recover the file. Default is `readRDS`

Value

A list of the objects that were read from disk. Currently the function only works with `.tif` (i.e. raster) and `.rds` objects.

Author(s)

Tati Micheletti

---

createShrubHerbLayers   *Creates the necessary shrub or herb layers for predictive caribou RSF module.*

Description

Creates the necessary shrub or herb layers for predictive caribou RSF module.

Usage

```
createShrubHerbLayers(
  landCoverECCC,
  reclassLCC05,
  layerName,
  includeCrops = FALSE
)
```

Arguments

`landCoverECCC`	RasterLayer LCC2005 reclassified to ECCC landcover.
`reclassLCC05`	List with reclassification for LCC05 values (i.e. LCC05 classes that should be classified as shrub or herbs)
`layerName`	character. Name of the shrub layer in the model.
`includeCrops`	Logical. Should this layer include crops?

Value

`RasterLayer` of herbs or shrubs

Author(s)

Tati Micheletti

---

defineStudyArea   *Preparing study area for Canadian projects*

Description

Downloads, reprojects, crops and masks to specific areas in Canada such as: boreal, random areas, provinces and territories, or any of the last in the boreal.

Usage

```
defineStudyArea(
  testArea = NULL,
  specificTestArea = NULL,
  mapSubset = NULL,
  destinationFolder = tempdir()
)
```

Arguments

testArea   Logical. Indicates if the test area should be anything other than NULL. Default is
                  NULL.

specificTestArea

                  A character string with a province or territory name, or `'boreal'`. (following
                  Brandt *et al.*, 2013). Default is NULL.

mapSubset   If `specificTestArea` is supplied as `'boreal'`, this can be set as a character
                  string with a province or territory that is contained in the boreal or `'Canada'` if
                  the whole Canadian boreal is wanted. Default is NULL.

destinationFolder

                  Path to where to save downloaded files. Default is `tempdir()`.

Author(s)

Tati Micheletti

Examples

```
## rp is the whole North American boreal region
rp <- defineStudyArea(testArea = TRUE, specificTestArea = "boreal", mapSubset = NULL)

## Alberta
rp <- defineStudyArea(testArea = TRUE, specificTestArea = "Alberta", mapSubset = NULL)

## Alberta inside boreal extension
rp <- defineStudyArea(testArea = TRUE, specificTestArea = "boreal", mapSubset = "Alberta")
```

---

extractDisturbanceFast

Extract disturbance of anthropogenic and fire

Description

Extract disturbance of anthropogenic and fire

Usage

```
extractDisturbanceFast(
  ageMap,
  caribouShapefile,
  recoveryTime = 40,
  anthropogenicLayer = NULL,
  waterRaster,
  rasterToMatch,
  destinationPath
)
```

Arguments

ageMap	RasterLayer. Map with forest age.
caribouShapefile	
	Shapefile with polygons for which we want to calculate lambda for the caribou demographic model.
recoveryTime	numeric. Recovery time in years that the forest needs to support Caribou. Default = 40.
anthropogenicLayer	
	Anthropogenic disturbance (raster) layer. Currently, road density layer used for both RSF and demographic models.
waterRaster	RasterLayer indicating water bodies.
rasterToMatch	RasterLayer template for these layers to match.
destinationPath	
	TODO

Value

A list of the anthropogenic and fire disturbances as percent (0-100)

Author(s)

Tati Micheletti

---

forestAgePlot   *Plots of forest age*

Description

Plots of forest age

Usage

```
forestAgePlot(
  dataPath,
  typeSim,
  addCaribousuitability = FALSE,
  overwrite = FALSE
)
```

Arguments

dataPath   character. Path to data

typeSim   character. Which simulation is it? i.e., 'LandR_SCFM' or 'LandR.CS_fS'

addCaribousuitability

                     logical. Should the plot show which forest ages are better or worse for Caribou in a colour-coded way?

overwrite   logical.

Value

plot

Author(s)

Tati Micheletti

---

getLayers   *Get the necessary layers for predictive modules such as* caribouPopGrowth *and* caribouRSF.

Description

Get the necessary layers for predictive modules such as caribouPopGrowth and caribouRSF.

Usage

```
getLayers(
  currentTime,
  cohortData,
  pixelGroupMap,
  startTime,
  endTime,
  recoveryTime = 40,
  listSACaribou,
  anthropogenicLayer,
  roadDensity,
  waterRaster,
  isRSF = FALSE,
  decidousSp = NULL,
  oldBurnTime = NULL,
  elevation = NULL,
  vrug = NULL,
  LCC05 = NULL,
  reclassLCC05 = NULL,
  rasterToMatch = NULL,
  destinationPath
)
```

Arguments

`currentTime`	numeric. Current time being used (i.e. `time(sim)`).
`cohortData`	`data.table`. Output from `LandR_Biomass` module.
`pixelGroupMap`	raster to identify the `cohortData`.
`startTime`	numeric. start time of the simulation. Needed to verify and potentially adjust relative simulation times.
`endTime`	numeric. end time of the simulation. Needed to verify and potentially adjust relative simulation times.
`recoveryTime`	numeric. Recovery time in years that the forest needs to support Caribou. Default = 40.
`listSACaribou`	list of shapefiles with polygons for which we want to calculate lambda for the caribou demographic model.
`anthropogenicLayer`	
	Anthropogenic disturbance (raster) layer. Currently, 500 m buffered anthropogenic disturbance for demographic models.
`roadDensity`	Anthropogenic disturbance (raster) layer. Currently, road density layer used for RSF models.
`waterRaster`	`RasterLayer` indicating water bodies.
`isRSF`	logical. Identify if it should get the layers for the RSF or demographic model.
`decidousSp`	binary raster layer indicating if the dominant biomass in a pixel belongs to a deciduous species.

oldBurnTime	numeric. Definition of the initial interval considered to be old burn. The end of this time is 20 years later (i.e. 40-60 years).
elevation	RasterLayer of elevation
vrug	RasterLayer of ruggedness
LCC05	RasterLayer of landcover classes 2005
reclassLCC05	List with reclassification for LCC05 values (i.e. LCC05 classes that should be classified as shrub or herbs)
rasterToMatch	RasterLayer template for these layers to match.
destinationPath	
	TODO

Value

TODO

Author(s)

Tati Micheletti

---

grepMulti *Grep using multiple patterns*

Description

Works similarly to grepl, but for multiple patterns and returning the object.

Usage

```
grepMulti(x, patterns, unwanted = NULL)
```

Arguments

x	object where to look for patterns.
patterns	Character vector of patterns to look for objects.
unwanted	Character vector of patterns to exclude from search.

Value

The objects with specified patterns combined

Author(s)

Tati Micheletti

---

makeReclassifyMatrix   *Create a matrix to use for reclassification of rasters*

Description

Create a matrix to use for reclassification of rasters

Usage

```
makeReclassifyMatrix(table, originalCol, reclassifiedTo)
```

Arguments

table	`data.frame` or `data.table` that has the original classification and the desired classification to replace the first.
originalCol	character. Column in the table that identifies the original classification.
reclassifiedTo	character. Column in the table that identifies the desired classification.

Value

Matrix to use for reclassification of rasters

Author(s)

Tati Micheletti

---

meanValuesTime   *Calculates the mean value of rasters through time*

Description

Calculates the mean value of rasters through time

Usage

```
meanValuesTime(ras, scenario, initialTime)
```

Arguments

ras	`RasterStack`. Time series used to calculate the mean value through time.
scenario	character. Which scenario are you running i.e., `LandR.CS_fS` Needs to match the rasters. Default is NULL (i.e., the whole raster is only one area)
initialTime	numeric. Format of the first year of analysis.

Value

table with `average`, `SD` and `CI95%`

Author(s)

Tati Micheletti

---

modulesAvailable   *Show the modules that have stickers available*

Description

Show the modules that have stickers available

Usage

```
modulesAvailable()
```

Value

A character vector of all modules that have stickers available.

Author(s)

Tati Micheletti

---

moduleSticker   *Function to create stickers for* SpaDES *modules*

Description

Function to create stickers for SpaDES modules

Usage

```
moduleSticker(
  moduleTable = NULL,
  moduleName,
  directory = getwd(),
  useCache = NULL,
  savedSticker = NULL,
  ...
)
```

Arguments

`moduleTable`	Table with the information to generate the sticker (i.e. figure name, height, colours, etc)
`moduleName`	character string of the module's name.
`directory`	character string of the directory where the sticker should be saved
`useCache`	logical. Should the process of downloading and loading the table be cached?
`savedSticker`	Character string. Name of the file to be saved. If not provided, saves with the name of the figure. Useful if more than one stickers have the same figure
`...`	Other parameters for `ggplot2` (i.e. `fontface`, `lineheight`)

Value

This function returns the location where the sticker is available

Author(s)

Tati Micheletti

---

nameAndBringOn	*Name a raster,* `postProcess` *it using RTM, and brings it to memory*

Description

Name a raster, `postProcess` it using RTM, and brings it to memory

Usage

```
nameAndBringOn(ras, name, RTM = NULL)
```

Arguments

`ras`	RasterLayer.
`name`	character. Name of the raster layer.
`RTM`	RasterLayer template. If a RTM is passed, the function masks the raster to it, converting non-NA to 0.

Value

RasterStack

Author(s)

Tati Micheletti

---

plotBurnSummary *Plots summary of burns*

Description

Plots summary of burns

Usage

```
plotBurnSummary(
  dataPath,
  typeSim,
  lastYear,
  theObject = NULL,
  overwrite = FALSE
)
```

Arguments

dataPath   character. Path to data

typeSim   character. Which simulation is it? i.e., `'LandR_SCFM'` or `'LandR.CS_fS'`

lastYear   TODO

theObject   TODO

overwrite   logical. Default FALSE

Value

plot

Author(s)

Tati Micheletti

---

plotLeadingVegetationType
                    Plots Leading Vegetation Type using cohortData *and* pixelGroupMap

---

Description

Plots Leading Vegetation Type using cohortData and pixelGroupMap

Usage

```
plotLeadingVegetationType(
  dataPath,
  typeSim,
  colNA = "grey85",
  saveRAS = TRUE,
  overwrite = FALSE
)
```

Arguments

dataPath	character. Path to data
typeSim	character. Which simulation is it? i.e. 'LandR_SCFM' or 'LandR.CS_fS'.
colNA	character. The colour to use for NA.
saveRAS	logical. Save the raster for posterior use?
overwrite	logical.

Value

plot

Author(s)

Tati Micheletti

---

plotVegetationBiomass   *Plots vegetation biomass type using* cohortData *and* pixelGroupMap

---

Description

Plots vegetation biomass type using cohortData and pixelGroupMap

Usage

```
plotVegetationBiomass(
  years = c(2011, 2100),
  dataPath,
  typeSim,
  colNA = "grey85",
  saveRAS = TRUE,
  overwrite = FALSE
)
```

Arguments

years	numeric. Years available/intended to be used.
dataPath	character. Path to data.
typeSim	character. Which simulation is it? i.e., 'LandR_SCFM' or 'LandR.CS_fS'.
colNA	character. The colour to use for NA.
saveRAS	logical. Save the raster for posterior use?
overwrite	logical.

Value

plot

Author(s)

Tati Micheletti

---

prepareClimateLayers   *Create raster stack for climate sensitive models*

Description

Designed primarily for NWT project, but somewhat flexible.

Usage

```
prepareClimateLayers(
  pathInputs = NULL,
  variables = NULL,
  years = NULL,
  GDriveFolder = NULL,
  climateFilePath = NULL,
  fileResolution = NULL,
  authEmail = NULL,
  RCP = NULL,
  climateModel = NULL,
  ensemble = NULL,
  rasterToMatch = NULL,
  studyArea = NULL,
  model = NULL,
  droughtMonths = 4:9,
  returnCalculatedLayersForFireSense = FALSE,
  yearsWithClimateProjections = 2011:2100,
  overwrite = FALSE,
  overwriteOriginalData = FALSE
)
```

Arguments

pathInputs	Default NULL (TODO: description needed)
variables	Character string of the variables to be used, i.e. c("PPT", "Tmax").
years	Character string of the years to use. i.e. c(2011:20100).
GDriveFolder	Character string of the folder in Google Drive to upload the layers to. Handy for shared projects.
climateFilePath	
	Character string of the path to the climate file in google drive (i.e. "https://drive.google.com/open?i for Canada3ArcMinute.7z)
fileResolution	Character string of the for naming purposes (i.e. 3ArcMinute)
authEmail	Character string of Google email for authentication for non-interactive use.
RCP	Character string of RCP to be used (i.e. 45)
climateModel	Character string of climate mode to be used (i.e. CanESM2)
ensemble	Character string of climate ensemble to be used (i.e. r11i1p1)
rasterToMatch	RasterLayer template for these layers to match
studyArea	shapefile of study area
model	For naming and shortcut for variables: i.e., "birds" or "fireSense". If providing the variables to be produced, don't use "birds" nor "fireSense" here.
droughtMonths	Numeric. Months to calculate Monthly Drought Code (MDC) i.e. 4:9.
returnCalculatedLayersForFireSense	
	Logical. Should it calculate MDC (TRUE) or return the original variables (FALSE)? Default is FALSE.
yearsWithClimateProjections	
	Numeric. The user can pass the years that have climate projection in the data. Default to 2011:2100.
overwrite	logical. Default FALSE. Should the layers be overwritten if exist?
overwriteOriginalData	
	logical. Default FALSE. If changes happen in the original layer (the one provided in climateFilePath), set this to TRUE to overwrite the zip files downloaded and return the calculated MDC layers already, not the original stack with Tmax and PPT.

Value

A list of all years, with each year being the local path for the raster stack that contains all variables.

Author(s)

Tati Micheletti

prepInputsLayers_DUCKS

Prepare DUCKS layer

Description

Intended to prepare the DUCKS Unlimited Hybrid Wetland v. 2.1 layer for different purposes. The output is a `RasterLayer` cropped and reprojected to the `studyArea`, and resampled to the `rasterToMatch` if any of these are provided.

Usage

```
prepInputsLayers_DUCKS(
  destinationPath,
  lccLayer = "2005",
  url = NULL,
  archive = NULL,
  targetFile = NULL,
  studyArea = NULL,
  rasterToMatch = NULL,
  overwrite = TRUE
)
```

Arguments

destinationPath

Path where to save the downloaded file.

lccLayer   Which year should be used as a base for the vegetation layer? Default to 2005

url   The url from where the layer should be downloaded from. If `NULL`, the default is used.

archive   TODO

targetFile   TODO

studyArea   Study area for which the layer should be cropped to

rasterToMatch TODO

overwrite   Logical indicating whether to overwrite the previously processed file.

Value

RasterLayer

---

prepInputStack   *Simple wrapper around* prepInputs *for a stack of raster layers*

Description

Simple wrapper around prepInputs for a stack of raster layers

Usage

```
prepInputStack(...)
```

Arguments

...	Arguments passed to fun (i.e,. user supplied), postProcess and Cache. Since ... is passed to postProcess, these will also be passed into the inner functions, e.g., cropInputs. User should supply several named arguments here, including: targetFile, archive, url, alsoExtract, destinationPath, fun, quick, purge, overwrite, and useCache. See details and examples.

Value

RasterStack

Author(s)

Tati Micheletti

---

provinceBCRStudyArea   *Preparing study area based on BCR and Canadian or American provinces or states.*

Description

Downloads, reprojects, crops and masks to specific areas in Canada such as: BCR6, random areas, provinces and territories, or any of the last in the BCR6.

Usage

```
provinceBCRStudyArea(bcr = NULL, province = NULL, country, ...)
```

Arguments

`bcr`	Numeric. Bird Conservation Region in North America that you want to crop for. Default is `NULL`. If 'NULL", it returns only the shapefile of the province.
`province`	Character string. Province or territory. Default is `NULL`. If null, returns the map of ??.
`country`	Character string. 3 letter ISO for a specific country. The complete list of countries can be seen by calling `raster::getData('ISO3')`.
`...`	Arguments to be passed to `prepInputs` or `Cache` (i.e. `targetFile`, `cacheId`, `destinationPath`, `overwrite`, etc.).

Details

If you provide a BCR that is outside of a province area, the object returned is `NULL`.

Author(s)

Tati Micheletti

---

reviseSpeciesTraits	*Bootstraps rasters for testing significance on comparable rasters of different species or scenarios*

Description

Bootstraps rasters for testing significance on comparable rasters of different species or scenarios

Usage

```
reviseSpeciesTraits(
  speciesTable = NULL,
  updatedTable = NULL,
  destinationPath = tempdir(),
  returnTable = FALSE,
 species = c("BETU.PAP", "LARI.LAR", "PICE.GLA", "PICE.MAR", "PINU.BAN", "POPU.TRE",
    "PINU.CON")
)
```

Arguments

`speciesTable`	`data.table`. Table to be updated. If `returnTable == TRUE` this can be NULL. In this case, the function will use the ORIGINAL species table traits (from LANDIS-II).
`updatedTable`	`data.table`. Table used to update `speciesTable`. If NULL, the function uses a default.
`destinationPath`	
	character. Where to save the table. Default is `tempdir()`.

returnTable	logical. If TRUE, it uses the ORIGINAL species table traits from LANDIS-II. If FALSE, you have to supply your own table.
species	character. Species to update the table for.

Value

Table of updated tree species traits to be used in LandR biomass simulation

Author(s)

Tati Micheletti

RSFplot	*ECCC's RSF plot style*

Description

ECCC's RSF plot style

Usage

```
RSFplot(
  ras,
  upload = FALSE,
  writeReclasRas = FALSE,
  outputFolder = tempdir(),
  rasName,
  folderID = NULL
)
```

Arguments

ras	RasterLayer. Layer to generate 10 binned green to red (caribou RSF) map.
upload	logical. Should the raster be uploaded to Google Drive? Only works if the raster exists OR is set to be written.
writeReclasRas	logical. Default is FALSE. Should the raster be written to disk?
outputFolder	character. Path to the folder where it should be saved.
rasName	character. Name of the raster to be saved.
folderID	character. Google folder id to upload to. Only needs to be provided if upload == TRUE. Default is 'NULL'

Value

Used for side effects of generating plots that have similar binning system to ECCC's 2011 Caribou RSF, and optionally uploading them to Google Drive.

Author(s)

Tati Micheletti

---

subsetNonNARas   *Get all the values of a raster and remove the NA's, keeping pixel ID.*

Description

Get all the values of a raster and remove the NA's, keeping pixel ID.

Usage

```
subsetNonNARas(ras, N = NULL)
```

Arguments

ras   RasterLayer.

N   numeric. If passed, subsets N pixels from the raster. If NULL, returns the whole
                   non-NA raster values. Default is NULL.

Value

data.table of pixel values and pixel ID

Author(s)

Tati Micheletti

---

substrBoth   *Get a sub-string based on the number of characters and the side to
                  start*

Description

Get a sub-string based on the number of characters and the side to start

Usage

```
substrBoth(strng, howManyCharacters, fromEnd = TRUE)
```

Arguments

strng   String from which to grab a subset

howManyCharacters

                  numeric. How many characters should be returned in the sub-string?

fromEnd   logical. Default is TRUE. Should the subset start in the end of the string?

Value

character string of the subset.

Author(s)

Tati Micheletti

---

totalBiomassPerSpecies

Plots biomass per species: proportional or absolute, and total or just overstory

---

Description

Plots biomass per species: proportional or absolute, and total or just overstory

Usage

```
totalBiomassPerSpecies(
  dataPath,
  typeSim,
  proportional = FALSE,
  columnsType = FALSE,
  overstory = FALSE,
  overwrite = FALSE,
  maxVal = 2e+10
)
```

Arguments

dataPath	character. Path to data
typeSim	character. Which type of simulation is it? i.e., 'LandR_SCFM' or 'LandR.CS_fS'
proportional	logical. Should the plot be of the proportional biomass?
columnsType	logical. Should the plot be continuous (lines) or columns?
overstory	logical. Should the plot be of the overstory biomass?
overwrite	logical.
maxVal	numeric. Max value for y axis. Passing this ensures that both overstory and all biomass plots are comparable Default to 1e10.

Value

plot

Author(s)

Tati Micheletti

Index