

Package: amc (via r-universe)

October 9, 2026

Type Package

Title Alex's Miscellaneous Code

Description A collection of variously useful functions and utilities.

Version 1.0.3

Date 2025-07-28

Depends R (>= 4.3)

Imports base64enc, data.table, digest, dismo, httr, methods,
reproducible, raster (>= 3.3-13), sf, sp, terra, tools, utils

Suggests covr, datasets, rstudioapi, spelling, testthat

Remotes PredictiveEcology/reproducible@development

BugReports <https://github.com/achubaty/amc/issues>

Language en-CA

License GPL-3

ByteCompile yes

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Roxygen list(markdown = TRUE)

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev make default-jdk libuv1-
dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

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

Date/Publication 2025-07-29 04:19:13 UTC

RemoteUrl <https://github.com/achubaty/amc>

RemoteRef development

RemoteSha 09a6f01363335d991d8139dffaac029881133f5d

Contents

.gc	2
binstr	3
detachAllPackages	4
detachPackage	4
dir.copy	5
dl.data	5
dt2raster	6
flink	7
geometricMean	8
get_deps	9
getFileName	10
guesstimate	11
hill	12
inRange	12
isRstudio	13
loadkNNageMap	13
loadObjects	14
loadStudyArea	15
logit	15
min_r_version	16
mosaic2	16
outerBuffer	17
pkgSrc	18
rescale	19
rndstr	20
source_github	21
sysmem	21
td	22
Index	23

.gc	<i>Manual garbage collection</i>
-----	----------------------------------

Description

This shouldn't be necessary, since R (usually) handles this correctly and automatically. However, sometimes when working with large geospatial data (e.g., using raster and sp packages) it can help to free recently unallocated memory manually.

Usage

.gc()

Author(s)

Alex Chubaty

See Also[gc\(\)](#)

binstr	<i>Convert integer to binary string</i>
--------	---

Description

Convert integer to binary string

Usage

```
binstr(i, maxBits = NA)
```

Arguments

i	Positive integer $\leq 2^{53}$ ($\leq 9.007199\text{e}+15$).
maxBits	Maximum number of bits to print (default NA).

Value

Character vector.

Author(s)

Alex Chubaty

Examples

```
x <- sample(0:9999, 10000)
y <- binstr(x) # length is 14 bits

## Not run:
# alternate (but slower) conversion to binary string
R.utils::intToBin(x)

## End(Not run)

# convert binary string to integer value (very fast)
strtoi(y, base = 2)
strtoi(substr(y, 1, 4), base = 2)
strtoi(substr(y, 5, 8), base = 2)
strtoi(substr(y, 9, 11), base = 2)
strtoi(substr(y, 12, 14), base = 2)

# see also `binary()` and `unbinary()` in the `composition` package (requires x11)
```

detachAllPackages	<i>Forcibly detach all packages</i>
-------------------	-------------------------------------

Description

Based on <https://stackoverflow.com/a/39235076/1380598>.

Usage

```
detachAllPackages()
```

Author(s)

mmfrgmpds

See Also

[detach\(\)](#), [detachPackage\(\)](#)

detachPackage	<i>Detach and unload a package</i>
---------------	------------------------------------

Description

A simple wrapper to detach using unload = TRUE.

Usage

```
detachPackage(package)
```

Arguments

package	The name of a currently attached package.
---------	---

Author(s)

Alex Chubaty

See Also

[detach\(\)](#), [detachAllPackages\(\)](#)

dir.copy	<i>Copy folders with links</i>
----------	--------------------------------

Description

Copies folders like `file.copy` except it replicates links correctly on unix-like systems. Based on <http://stackoverflow.com/a/30107868/1380598>.

Usage

```
dir.copy(from, to)
```

Arguments

from	character indicating the path to the directory to be copied.
to	character indicating the path to which the directory will be copied.

Value

Logical indicating success or failure.

Author(s)

Zach Foster
Alex Chubaty

dl.data	<i>Intelligently download data</i>
---------	------------------------------------

Description

Only downloads the specified files if it is not found locally. Optionally unzips the files.

Usage

```
dl.data(urls, dest = ".", checksum = TRUE, unzip = FALSE)
```

Arguments

urls	A character vector of data file URLs.
dest	The directory path in which data should be downloaded.
checksum	Logical indicating whether downloaded files should be checksummed.
unzip	Logical indicating whether the file should be unzipped after download.

Author(s)

Alex Chubaty and Eliot McIntire

dt2raster	<i>Convert data.table to a RasterLayer for plotting, etc.</i>
-----------	---

Description

Convert data.table to a RasterLayer for plotting, etc.

Usage

```
dt2raster(dt, r, val)
```

Arguments

dt	data.table object with columns ID, or both X and Y, and the values to assign to the raster specified by column val.
r	Raster* object.
val	The name of the column in dt containing the values for the raster.

Value

A RasterLayer object.

Author(s)

Alex Chubaty

Examples

```
library(data.table)
library(sp)
library(raster)

r <- raster(nrows = 10, ncols = 10)
r[] <- 10

# using x,y coordinates
#dt1 <- data.table(X = , Y = , value = r[])

# using pixel ids
dt2 <- data.table(ID = 1L:ncell(r), VALUE = r[])
dt2[, VALUE := sample(1L:10L, ncell(r), replace = TRUE)]

if (interactive())
  plot(dt2raster(dt2, r, "VALUE"))
```

flink	Create a link to a file
-------	-------------------------

Description

Creates a symbolic link (symlink) to a file if possible, possibly falling back to a hard link. Hard links are for files only, and won't work across different physical drives. Symlinks won't work on Windows without admin privileges.

Usage

```
flink(from, to, symlink = TRUE)
```

Arguments

from, to	character vectors, containing file names or paths (can alternatively be the path to a single existing directory).
symlink	Logical indicating whether to use symlink (instead of hardlink). Default TRUE.

Note

Use caution with files-backed objects (e.g., rasters). See examples.

Author(s)

Alex Chubaty

See Also

[file.link\(\)](#), [file.symlink\(\)](#), [Sys.readlink\(\)](#)

Examples

```
if (require("datasets", quietly = TRUE)) {
  library(raster)

  tmpDir <- file.path(tempdir(), 'symlink-test') |>
    normalizePath(winslash = '/', mustWork = FALSE)
  dir.create(tmpDir)

  f0 <- file.path(tmpDir, "file0.csv")
  write.csv(iris, f0)

  d1 <- file.path(tmpDir, "dir1")
  dir.create(d1)
  write.csv(iris, file.path(d1, "file1.csv"))

  d2 <- file.path(tmpDir, "dir2")
  f2 <- file.path(tmpDir, "file2.csv")
}
```

```

## create a link to the the directory; d2 should look like d1
flink(d1, d2) ## symlink
dir.exists(d2) ## TRUE
identical(d1, Sys.readlink(d2)) ## TRUE
file.exists(file.path(d2, "file1.csv")) ## TRUE

## create link to a file
flink(f0, f2) ## symlink
file.exists(f2) ## TRUE
identical(read.csv(f0), read.csv(f2)) ## TRUE

## deleting the link shouldn't delete the original file
unlink(d2, recursive = TRUE)
file.exists(file.path(d2, "file1.csv")) ## FALSE
file.exists(file.path(d1, "file1.csv")) ## TRUE

unlink(f2)
file.exists(f2) ## FALSE
file.exists(f0) ## TRUE

## using rasters and other file-backed objects
f3 <- system.file("external/test.grd", package = "raster")
r3 <- raster(f3)
f4 <- file.path(tmpDir, "raster4.grd")
flink(f3, f4, FALSE) ## hardlink the grd and gri files
flink(extension(f3, "gri"), extension(f4, "gri"), FALSE)

file.exists(f4) ## TRUE
file.exists(extension(f4, "gri")) ## TRUE
r4 <- raster(f4) ## hardlink

f5 <- file.path(tmpDir, "raster5.grd")
flink(f3, f5, TRUE) ## symlink the grd and gri files
flink(extension(f3, "gri"), extension(f5, "gri"), TRUE)
file.exists(f5) ## TRUE
file.exists(extension(f5, "gri")) ## TRUE
r5 <- raster(f5) ## symlink works
identical(r3, r5) ## TRUE

## cleanup
unlink(tmpDir, recursive = TRUE)
}

```

geometricMean

Geometric and harmonic mean

Description

Description needed.

Usage

```
geometricMean(x, ...)
```

```
harmonicMean(x, ...)
```

Arguments

x A numeric vector.

... Additional arguments to prod or mean.

Value

A numeric vector of length one.

Note

these have not been thoroughly tested to handle NA values, etc.

Author(s)

Alex Chubaty

Examples

```
series <- 1:10  
mean(series)  
geometricMean(series)  
harmonicMean(series)
```

get_deps

Get package dependencies (offline)

Description

Read a package's dependencies from file, rather than searching CRAN. Based on <http://stackoverflow.com/a/30225680/1380598>.

Usage

```
get_deps(path, dependencies = NA)
```

Arguments

path	A local file path to a package directory.
dependencies	Logical indicating whether to also install uninstalled packages which these packages depend on/link to/import/suggest (and so on recursively). Can also be a character vector, a subset of <code>c("Depends", "Imports", "LinkingTo", "Suggests", "Enhances")</code> . The default, <code>NA</code> , means <code>c("Depends", "Imports", "LinkingTo")</code> . <code>TRUE</code> means to use <code>c("Depends", "Imports", "LinkingTo", "Suggests")</code> .

Value

A character vector of package dependencies.

Author(s)

Josh O'Brien
Alex Chubaty

Examples

```
get_deps(system.file(package = "amc"))
get_deps(system.file(package = "amc"), TRUE)
```

getFileName	<i>Get the name of a source-ed file</i>
-------------	---

Description

Use `getFileName` in a file that is source-ed. Based on <http://stackoverflow.com/a/1816487/1380598>.

Usage

```
getFileName(fullname)

## S4 method for signature 'logical'
getFileName(fullname)
```

Arguments

fullname	Logical (default <code>FALSE</code>) indicating whether the full path should be returned.
----------	--

Value

Character string representing the filename.

Author(s)

Alex Chubaty

guesstimate*Guesstimate the number of CPUs for cluster operations*

Description

Take a wild stab at guessing how many CPUs to use in cluster when you have some idea of how much RAM is needed per CPU.

Usage

```
guesstimate(ram, prop = 0.8, units = "gb")
```

Arguments

ram	How much ram is required per CPU.
prop	Proportion of overall RAM to devote to R. Default 0.80.
units	Units of memory. One of either "KB", "MB", "GB".

Details

Tries to be conservative by assuming no more than 80% system memory use.

Value

Integer. Number of CPUs to allocate to cluster.

Note

You should take these numbers with several grains of salt.

Author(s)

Alex Chubaty

Examples

```
## Not run:  
guesstimate(4)  
guesstimate(4, 0.90, "MB")  
  
## End(Not run)
```

hill	<i>Hill function</i>
------	----------------------

Description

Hill function

Usage

hill(a, b, z)

Arguments

a	DESCRIPTION NEEDED
b	DESCRIPTION NEEDED
z	DESCRIPTION NEEDED

Value

DESCRIPTION NEEDED

Author(s)

Devin Goodsman

inRange	<i>Test whether a number lies within range [a,b]</i>
---------	--

Description

Default values of a=0; b=1 allow for quick test if x is a probability.

Usage

inRange(x, a = 0, b = 1)

Arguments

x	values to be tested
a	lower bound (default 0)
b	upper bound (default 1)

Value

Logical vectors. NA values in x are retained.

Author(s)

Alex Chubaty

Examples

```
set.seed(100)
x <- stats::rnorm(4) ## -0.50219235  0.13153117 -0.07891709  0.88678481
inRange(x, 0, 1)     ## FALSE  TRUE FALSE  TRUE
```

isRstudio

*Check whether R is running in an Rstudio session***Description**Based on <https://stackoverflow.com/q/12389158/1380598>.**Usage**

isRstudio()

loadkNNageMap

*Load kNN stand age map***Description**

Load kNN stand age map

Usage

loadkNNageMap(path, url = NULL, studyArea = NULL, ...)

Arguments

path	file path where raster will be saved.
url	URL from which to download the data (default provided if NULL).
studyArea	SpatialPolygonsDataFrame giving the study area for which to extract ages.
...	Additional arguments passed to Cache (only userTags currently used).

loadObjects

*Load, save, and remove .RData objects***Description**

Wrapper functions to [load\(\)](#), [save\(\)](#), and [unlink\(\)](#), permitting lists of objects to be loaded/saved/deleted all at once.

Usage

```
loadObjects(
  objects,
  path = NULL,
  ext = ".RData",
  quiet = TRUE,
  envir = parent.frame()
)
```

```
saveObjects(
  objects,
  path = NULL,
  ext = ".RData",
  quiet = TRUE,
  envir = parent.frame()
)
```

```
rmObjects(objects, path = NULL, ext = ".RData", quiet = TRUE)
```

Arguments

objects	A character list or character vector of object names
path	The filepath to the directory in which to save or from which to load the objects. The path should be constructed using file.path() .
ext	The file extension to use (default is .RData).
quiet	Logical. Should output be suppressed? Default is TRUE.
envir	The environment in which to look for and load objects (default: the environment from which the function was called).

Details

By default, the extension .RData is used.

Value

Invisibly if quiet=TRUE. Either a list of objects loaded, empty list if saved, or if removed either 0 for success, 1 for failure.

Author(s)

Alex Chubaty

See Also

[file.path\(\)](#), [load\(\)](#), [save\(\)](#), [unlink\(\)](#)

loadStudyArea	<i>Load a study area from file</i>
---------------	------------------------------------

Description

Simple wrapper around [sf::st_read\(\)](#) to load a kml or shapefile, and optionally reproject it.

Usage

```
loadStudyArea(path = NULL, filename = NULL, proj = NULL)
```

Arguments

path	path to directory containing the file
filename	the name of the file
proj	(optional) a crs projection string to reproject the study area to.

Value

An sf object.

logit	<i>Logit function</i>
-------	-----------------------

Description

Logit function

Usage

```
logit(p)
```

Arguments

p	DESCRIPTION NEEDED
---	--------------------

Value

DESCRIPTION NEEDED

min_r_version	<i>Determine a package's minimum R version requirement based on its dependencies</i>
---------------	--

Description

Based on <https://stackoverflow.com/q/38686427>.

Usage

```
min_r_version(package = NULL, exclude_main_pkg = TRUE)
```

Arguments

package	Character string giving the name of a package whose dependencies should be checked.
exclude_main_pkg	Logical indicating whether package should be excluded from the check. Default TRUE.

Author(s)

hrbrmstr and Jack Wasey

mosaic2	<i>Merge Raster* objects using a function for overlapping areas</i>
---------	---

Description

Provides a wrapper around `raster::mosaic()` that cleans up any temporary intermediate files used, and sets the layer name of the resulting raster.

Usage

```
mosaic2(x, y, ...)

## S4 method for signature 'RasterLayer,RasterLayer'
mosaic2(
  x,
  y,
  ...,
  fun,
  tolerance = 0.05,
  filename = NULL,
  layerName = "layer",
```

```

    inRAM = FALSE
)

## S4 method for signature 'SpatRaster,SpatRaster'
mosaic2(
  x,
  y,
  ...,
  fun,
  tolerance = 0.05,
  filename = NULL,
  layerName = "layer",
  inRAM = FALSE
)

```

Arguments

x	Raster* object
y	Raster* object
...	Additional Raster or Extent objects.
fun	Function (e.g., mean, min, or max, that accepts a <code>na.rm</code> argument).
tolerance	Numeric. Permissible difference in origin (relative to the cell resolution). See all.equal() .
filename	Character. Output filename (optional).
layerName	Character. Name of the resulting raster layer.
inRAM	Logical (default FALSE) indicating whether function operations should be performed in memory or, if TRUE, using temporary files.

Author(s)

Alex Chubaty

outerBuffer	<i>Draw convex hull around polygons</i>
-------------	---

Description

Draws a convex hull around vertice points of a polygon shapefile, creating a single polygon. If a buffer distance is supplied, will buffer the convex hull inwards or outwards depending on the sign of the distance value.

Usage

```
outerBuffer(x, b = NULL)
```

Arguments

- x** A `SpatialPolygons*` object
- b** Optional. Distance to buffer. If the value is negative, the buffer will be drawn inwards.

Value

A `SpatialPolygons` object.

Author(s)

Ceres Barros and Alex Chubaty

See Also

[raster::buffer\(\)](#)

pkgSrc

Determine source of installed packages

Description

Which packages were installed from CRAN, GitHub, Bioconductor, etc.?

Usage

```
pkgSrc(pkg, lib.loc = NULL)
```

Arguments

- pkg** a character string with the package name.
- lib.loc** a character vector of directory names of R libraries, or `NULL`. The default value of `NULL` corresponds to all libraries currently known. If the default is used, the loaded packages and namespaces are searched before the libraries.

Examples

```
pkgs <- as.data.frame(installed.packages(), stringsAsFactors = FALSE)
ids <- which(!(pkgs$Priority %in% c("base", "recommended")))
pkgs <- pkgs[ids, ]
pkgs <- pkgs$Package
pkgs[pkgSrc(pkgs) == "CRAN"]
```

rescale	<i>Rescale values to a new range</i>
---------	--------------------------------------

Description

Rescale values to a new range

Usage

```
rescale(x, to, from, ...)

## S3 method for class 'numeric'
rescale(x, to = c(0, 1), from = range(x, na.rm = TRUE, finite = TRUE), ...)

## S3 method for class 'RasterLayer'
rescale(
  x,
  to = c(0, 1),
  from = range(getValues(x), na.rm = TRUE, finite = TRUE),
  ...
)

## S3 method for class 'SpatRaster'
rescale(
  x,
  to = c(0, 1),
  from = range(values(x), na.rm = TRUE, finite = TRUE),
  ...
)
```

Arguments

x	A numeric vector or Raster* object.
to	The lower and upper bounds of the new range. Default c(0, 1).
from	(optional) The lower and upper bounds of the old range (calculated from x).
...	Additional arguments (not used).

Value

A new object whose values have been rescaled.

Note

Objects with values that are all equal (e.g., all zeroes) will be returned as-is. This behaviour differs from `scales::rescale` which would return a value of 0.5.

Examples

```

rescale(50, from = c(0, 100), to = c(0, 1)) ## 0.5

x <- 0:100
rescale(x) ## defaults to new range [0,1]
rescale(x, c(-1, 1))

f <- system.file("external/test.grd", package = "raster")
r <- raster::raster(f)
rescale(r) ## defaults to new range [0,1]
rescale(r, c(-1, 1))

f <- system.file("ex/test.grd", package = "terra")
r <- terra::rast(f)
rescale(r) ## defaults to new range [0,1]
rescale(r, c(-1, 1))

```

rndstr

Generate random strings

Description

Generate a vector of random alphanumeric strings each of an arbitrary length.

Usage

```
rndstr(n = 1, len = 8)
```

Arguments

n	Number of strings to generate (default 1). Will attempt to coerce to integer value.
len	Length of strings to generate (default 8). Will attempt to coerce to integer value.

Value

Character vector of random strings.

Author(s)

Alex Chubaty

Examples

```

set.seed(11)
rndstr()
rndstr(len = 10)
rndstr(n = 5, len = 10)
rndstr(n = 5)

```

source_github	<i>Source a file hosted in a public or private GitHub repo</i>
---------------	--

Description

Source a file hosted in a public or private GitHub repo

Usage

```
source_github(repo, branch = "master", file, auth = Sys.getenv("GITHUB_PAT"))
```

Arguments

repo	Name of the GitHub repository in the form "user/repo".
branch	Branch from which to source the file (default master).
file	Filename to source, including relative path.
auth	Personal Access Token to use for authorization. Required to access files in private repositories. By default, checks for GITHUB_PAT environment variable. See https://help.github.com/articles/creating-an-access-token-for-command-line-use/ .

Author(s)

Alex Chubaty

Examples

```
## Not run:
repo = "PredictiveEcology/SpaDES"
branch = "development"
file = "_ignore/thinSpatialPolygons.R"
auth = "" ## your Personal Access Token

source_github(repo, branch, file, auth)

## End(Not run)
```

systemem	<i>Check system memory</i>
----------	----------------------------

Description

This tells you the **TOTAL** system memory (RAM) available. Other processes running on the computer will eat into this total, and as such, you should take these numbers with a grain of salt.

Usage

```
sysmem(x = "GB")
```

Arguments

x Units to use for output. One of either "KB", "MB", "GB".

Value

Total amount of system memory (RAM) in units.

Author(s)

Alex Chubaty

Examples

```
sysmem()
```

td

Temporary directory and file creation

Description

These are wrappers around `tempdir` and `tempfile` that creates the directory or file, to ensure a correctly normalized filepath (i.e., on macOS).

Usage

```
td(dir = tempdir())
```

```
tf(ext = ".tif", dir = td())
```

Arguments

dir Path to use as temporary directory. A subdirectory will be created. Default is to use the R session temporary directory.

ext File extension to give to the newly create file.

Value

Character string indicating the filepath to the newly created file.

Author(s)

Alex Chubaty

Index

.gc, [2](#)
all.equal(), [17](#)
binstr, [3](#)

detach(), [4](#)
detachAllPackages, [4](#)
detachAllPackages(), [4](#)
detachPackage, [4](#)
detachPackage(), [4](#)
dir.copy, [5](#)
dl.data, [5](#)
dt2raster, [6](#)

file.link(), [7](#)
file.path(), [14](#), [15](#)
file.symlink(), [7](#)
flink, [7](#)

gc(), [3](#)
geometricMean, [8](#)
get_deps, [9](#)
getFileName, [10](#)
getFileName, logical-method
 (getFileName), [10](#)
guesstimate, [11](#)

harmonicMean (geometricMean), [8](#)
hill, [12](#)

inRange, [12](#)
isRstudio, [13](#)

load(), [14](#), [15](#)
loadkNNageMap, [13](#)
loadObjects, [14](#)
loadStudyArea, [15](#)
logit, [15](#)

min_r_version, [16](#)

mosaic2, [16](#)
mosaic2, RasterLayer, RasterLayer-method
 (mosaic2), [16](#)
mosaic2, SpatRaster, SpatRaster-method
 (mosaic2), [16](#)

outerBuffer, [17](#)

pkgSrc, [18](#)

raster::buffer(), [18](#)
raster::mosaic(), [16](#)
rescale, [19](#)
rmObjects (loadObjects), [14](#)
rndstr, [20](#)

save(), [14](#), [15](#)
saveObjects (loadObjects), [14](#)
sf::st_read(), [15](#)
source_github, [21](#)
Sys.readlink(), [7](#)
systemem, [21](#)

td, [22](#)
tf (td), [22](#)

unlink(), [14](#), [15](#)