Package 'EchoNet2Fish'

November 2, 2015

Title Estimate Fish Abundance from Acoustic Echoes and Net Catch

Version 0.0.0.9004

Description EchoNet2Fish estimates fish abundance from acoustic echoes and net catch.

Depends R (>= 3.2.2)

Imports rtf, maps, mapdata

Suggests magrittr, testthat

License GPL

LazyData TRUE

URL https://github.com/JVAdams/EchoNet2Fish

RoxygenNote 5.0.0

NeedsCompilation no

Author Jean V. Adams [aut, cre]

Maintainer Jean V. Adams <jvadams@usgs.gov>

R topics documented:

combinecsv *Combine Several Comma Delimited Files into a Single Data Frame*

Description

Combine all csv files in a given directory into a single data frame.

Usage

```
combinecsv(myDir, addSource = TRUE, column1name = "Region_ID")
```

Arguments

myDir	A character scalar naming the directory in which the csv files are stored. Should end in a forward slash, e.g., "C:/temp/". All the csv files should have the same number of columns with the same header row of column names.
addSource	A logical scalar indicating whether a new column, named "source", should be added to the data frame identifying the source file, default TRUE.
column1name	A character scalar assigning a name to the first column in the data frame (writing over whatever name is there already), default "Region_ID".

Details

The column1name argument is needed to handle occasional problems with byte order marks at the beginning of the csv files, which can result in strange characters being added to the name of the first column. See, for example, this link.

Value

A data frame with the information from all the csv files combined.

EchoEnv *EchoNet2Fish Package Local Environment*

Description

An environment local to the EchoNet2Fish package, used to hold objects outside of the individual package functions

Format

An environment.

Source

Post from Hadley Wickham to r-help on 2 Dec 2014 [link].

endrtf *Write and Close an RTF Document*

Description

Write and close an rtf (rich text format) document.

Usage

```
endrtf(rtf = doc, details = FALSE, ...)
```

Arguments

rtf	An rtf object, default doc.
details	Logical scalar indicating if session details should be added to the end of the document, default FALSE.
...	Additional parameters to addPageBreak.

References

This is a copy of the endrtf function from the [GLFC] package.

See Also

startrtf for an example, heading, para, tabl, figu, RTF, addPageBreak.

figu *Add a Figure to an RTF Document*

Description

Add a figure to an rtf (rich text format) document.

Usage

```
figu(..., FIG = fig, rtf = doc, figid = "Figure ",
  fign = EchoEnv$figcount, boldt = TRUE, capunder = TRUE, w = NULL,
  h = NULL, rf = 300, newpage = "none", omi = c(1, 1, 1, 1))
```

Arguments

...	One or more character scalars (separated by commas) of text to use for the figure caption.
FIG	A function to create a figure which will be added to the document, default fig.
rtf	An rtf object, default doc.
figid	Character scalar of caption identifier, default "Figure ".
fign	Numeric scalar of figure number to use in caption, default EchoEnv$figcount.
boldt	Logical scalar indicating if figure number should use bold font, default TRUE.

capunder	Logical scalar indicating if caption should appear under the figure (TRUE, the default) or on top of the figure (FALSE).
w	Numeric scalar width of figure in inches, default 6.5.
h	Numeric scalar height of figure in inches, default 8.
rf	Numeric scalar resolution of figure, default 300.
newpage	Character scalar indicating if the figure should start on a new page in the document "port" for a new portrait page, "land" for a new landscape page, and "none" for no new page (the default).
omi	Numeric vector, length 4, width of document page margins in inches (bottom, left, top, right), default c(1, 1, 1, 1).

Details

The figure and caption are written to the rtf file. The size of a new page is assumed to be 8.5 by 11 inches.

Value

A 1 is added to the numeric vector of length 1, `EchoEnv$figcount`, stored in the working directory to keep track of the number of figures written to the rtf document, and label the captions accordingly.

References

This is a copy of the `figu` function from the [GLFC] package.

See Also

`startrtf` for an example, `heading`, `para`, `tabl`, `endrtf`, `RTF`.

`getpackages` *Get Packages*

Description

Installs (if necessary) and attaches the specified packages.

Usage

```
getpackages(want)
```

Arguments

want	A character vector of package names.

References

This is a copy of the `getpackages` function from the [jvamisc] package.

heading *Add a Heading to an RTF Document*

Description

Add a text heading to an rtf (rich text format) document.

Usage

```
heading(words, htype = 1, rtf = doc)
```

Arguments

words	Character scalar text of heading to add to document.
htype	Integer scalar heading type, 1=bold and font size 12, 2=bold and font size 10, 3=italics and font size 10, default 1.
rtf	An rtf object, default doc.

Details

The specified heading is written to the rtf file.

References

This is a copy of the heading function from the [GLFC] package.

See Also

startrtf for an example, para, tabl, figu, endrtf, RTF.

Lakenames *Great Lakes Names*

Description

A vector with the names of the five Great Lakes.

Format

A character vector, length 5.

mapSymbols *Draw a Map using Different Colored Symbols*

Description

Draw a map using different colored symbols for data exploration purposes.

Usage

```
mapSymbols(lat, long, colorz, main = "", pch = 1, cex = 1.5, xla = 0,
  yla = xla)
```

Arguments

lat	A numeric vector of latitudes in decimal degrees.
long	A numeric vector of longitudes in decimal degrees. Same length as lat.
colorz	A vector of character or numeric colors to use, either of length 1 or the same length as lat and long.
main	A character scalar of the main title of the plot, default "".
pch	A vector of plotting characters or symbols, either of length 1 or the same length as lat and long, default 1. See points.
cex	A numeric vector giving the amount by which plotting characters and symbols should be scaled relative to the default, either of length 1 or the same length as lat and long, default 1.5.
xla	A numeric scalar giving an added margin of decimal degrees to be mapped beyond the range of longitudes in long, default 0.
yla	A numeric scalar giving an added margin of decimal degrees to be mapped beyond the range of latitudes in lat, default xla.

Details

The column1name argument is needed to handle occasional problems with byte order marks at the beginning of the csv files, which can result in strange characters being added to the name of the first column. See, for example, this link.

Examples

```
## Not run:
 latitude <- c(43.25, 45.73, 45.71, 44.84)
 longitude <- c(-82.30, -80.85, -84.03, -80.39)
 basincode <- c(1, 2, 1, 2)
 mapSymbols(lat=latitude, long=longitude, colorz=basincode+3,
  pch=16, xla=0.4)

## End(Not run)
```

mapText *Add Text to a Map*

Description

Add identifying text to a map based on a single grouping variable.

Usage

```
mapText(lat, long, group, cex = 1.5, ...)
```

Arguments

lat	A numeric vector of latitudes in decimal degrees.
long	A numeric vector of longitudes in decimal degrees. Same length aslat.
group	A character or numeric vector of group identifiers, the same length as lat and long.
cex	A numeric scalar giving the amount by which plotting characters should be scaled relative to the default, default 1.5.
...	Additional arguments to text.

Details

Group identifiers are added to a map, typically created with mapSymbols, at the midpoint of the range of each groups' latitudes and longitudes

Examples

```
## Not run:
 latitude <- c(43.25, 45.73, 45.71, 44.84)
 longitude <- c(-82.30, -80.85, -84.03, -80.39)
 basincode <- c(1, 2, 1, 2)
 basin <- c("Main", "GBay", "Main", "GBay")
 mapSymbols(lat=latitude, long=longitude, colorz=basincode+3,
  pch=16, xla=0.4)
 mapText(lat=latitude, long=longitude, group=basin)

## End(Not run)
```

midpoint *Midpoint Between the Minimum and the Maximum*

Description

Calculate the midpoint between the minimum and the maximum of a vector.

Usage

```
midpoint(x)
```

Arguments

x	A numeric vector.

Value

A numeric scalar representing the midpoint between the minimum and the maximum of x, ignoring missing values.

Examples

```
midpoint(c(10:20, 90))
```

---

para *Add a Paragraph to an RTF Document*

Description

Add a paragraph to an rtf (rich text format) document.

Usage

```
para(..., rtf = doc, bold = FALSE, italic = FALSE)
```

Arguments

...	One or more character scalars (separated by commas) of text to add to document as a single paragraph.
rtf	An rtf object, default doc.
bold	Logical scalar indicating if paragraph should use bold font, default FALSE.
italic	Logical scalar indicating if paragraph should use italic font, default FALSE.

Details

The specified heading is written to the rtf file.

References

This is a copy of the para function from the [GLFC] package.

See Also

startrtf for an example, heading, tabl, figu, endrtf, RTF.

plotIntLay *Plot Acoustic Survey Data using Different Colored Symbols*

Description

Plot acoustic survey data, interval vs. layer, using different colored symbols for data exploration purposes. Place multiple group-specific plots on one page, using the same x- and y-scales.

Usage

```
plotIntLay(interval, layer, group, grouporder = sort(unique(group)), colorz,
  main = "")
```

Arguments

`interval`	A numeric vector of intervals along the length of an acoustic transect.
`layer`	A numeric vector of layers from surface to bottom along the vertical water column of an acoustic transect, all values should be <= 0, the same length as `interval`.
`group`	A vector of group identifiers, the same length as `interval`.
`grouporder`	A vector of unique group identifiers, providing the order that each group will be plotted, the same length as `unique(group)`, default `sort(unique(group))`.
`colorz`	A vector of character or numeric colors to use, the same length as `interval`.
`main`	A character scalar of the main title of the plot, default "".

Details

The `column1name` argument is needed to handle occasional problems with byte order marks at the beginning of the csv files, which can result in strange characters being added to the name of the first column. See, for example, this link.

plotValues *Test for and Plot Errors in Acoustic Survey Values*

Description

Test for and plot errors in acoustic survey data, based on reported lows, highs, and in-between values.

Usage

```
plotValues(low, high, between, lowhighKnown = TRUE, varname = "Varname",
  test = FALSE, ...)
```

Arguments

low	A numeric vector of low values.
high	A numeric vector of high values, the same length as low.
between	A numeric vector of in between values, the same length as low.
lowhighKnown	A logical scalar indicating whether the vector representing the lows and the vector representing the highs are known, default TRUE. If FALSE, the low (and high) value is calculated as the elementwise mininum (and maximumu) of the three vectors, low, high, and between.
varname	A character scalar identifying what the values represent, used as the y-axis label if test=FALSE, default "Varname".
test	A logical scalar indicating whether to conduct a test for errors (TRUE) or to draw a plot of the results (FALSE, the default).
...	Additional arguments to plot.

Value

If test = TRUE, a logical scalar is returned indicating whether there were errors in the values (TRUE) or not (FALSE). If test = FALSE, a figure is drawn, but no value is returned.

---

recode *Recode Values*

Description

Assign new values to a vector.

Usage

```
recode(x, old, new, must.match = FALSE)
```

Arguments

x	A vector whose values will be recoded, can be character, numeric, or factor.
old	A vector of the unique values currently in the vector.
new	A vector of values which should replace the current ones.
must.match	A logical scalar indicating whether only those elements of the original vector with values in old should be returned (TRUE), or all values should be returned (FALSE, default) though some may be unchanged.

Value

A vector the same length as x (unless must.match=TRUE), with old values replaced by new values.

References

This is a copy of the recode function from the [jvamisc] package.

Examples

```
recode(c(1,1,1,2,3,4,1,10,3), 1:3, 1001:1003)
recode(c(1,1,1,2,3,4,1,10,3), 1:3, 1001:1003, must.match=TRUE)
```

`sliceCat` *Categorize Observations as Slices for Matching Acoustic Densities and Trawl Catches*

Description

Categorize observations as spatial slices for matching fish densities estimated from acoustic transects and speciec compositions estimated from midwater trawl catches.

Usage

```
sliceCat(sliceDef, fdp = NULL, bdp = NULL, lat = NULL, reg = NULL)
```

Arguments

`sliceDef`	A list of at least two named sub-lists defining the slices into which observations will be classified. Each sub-list contains one or more named numeric vectors of length two, identifying the parameter (the name of the vector) and the range of values that contribute to the slice definition. Each interval is closed on the left and open on the right (see Details). The name of each sub-list is the name of the slice to be assigned. See Examples.
`fdp`	A numeric vector of fishing depths (the distance from the surface of the water to the depth of a fish in the water) corresponding to the observations which are to be categorized into slices. Only necessary if required by `sliceDef`, default NULL. default
`bdp`	A numeric vector of bottom depths (the distance from the surface of the water to the substrate) corresponding to the observations which are to be categorized into slices. Only necessary if required by `sliceDef`, default NULL.
`lat`	A numeric vector of latitudes corresponding to the observations which are to be categorized into slices. Only necessary if required by `sliceDef`, default NULL.
`reg`	A character vector of regions corresponding to the observations which are to be categorized into slices. Only necessary if required by `sliceDef`, default NULL.

Details

Each interval of sliceDef is closed on the left and open on the right. In other words, if you assign an interval of fdp=c(10, 20), observations >= 10 and < 20 will be considered for inclusion in that slice.

All observation variables (`fdp`, `bdp`, `lat`, `lat`), if not NULL, must be the same length.

Value

A character vector the same length as the observations variables (`fdp`, `bdp`, `lat`, `reg`), identifying the slice to which each observation belongs.

Examples

```
myslicedef <- list(
  epiNear = list( fdp=c(0,   4), bdp=c(0,   6) ),
  epiOff =  list( fdp=c(0,   4), bdp=c(6, Inf) ),
  hypo =    list( fdp=c(4, Inf) )
)
```

```
fishingD <- 1:7
bottomD <- c(2, 10, 4, 12, 6, 14, 8)
slice <- sliceCat(myslicedef, fdp=fishingD, bdp=bottomD)
data.frame(fishingD, bottomD, slice)
```

`startrtf` *Create an RTF Document*

Description

Create an rtf (rich text format) document.

Usage

```
startrtf(file = NULL, dir = getwd(), width = 8.5, height = 11,
  omi = c(1, 1, 1, 1), quiet = FALSE)
```

Arguments

`file`	Character scalar name of document, default "RGeneratedDocument" with `Sys.Date()` suffix.
`dir`	Character scalar name of directory where document should be stored, default `getwd()`.
`width`	Numeric scalar width of document page in inches, default 8.5.
`height`	Numeric scalar height of document page in inches, default 11.
`omi`	Numeric vector, length 4, width of document page margins in inches (bottom, left, top, right), default c(1, 1, 1, 1).
`quiet`	Logical scalar indicating if name of new rtf document should be printed to command line, default FALSE.

Details

The rtf file may be written to until the `endrtf()` function is run. If you assign your rtf file to an object called doc, you can use the default settings in other **GLFC** rtf functions.

Value

An rtf file is created in the specified directory. An object of class rtf is created. This object is referred to in other functions to write to the file. In addition, two numeric vectors of length 1, `tabcount` and `figcount`, are written to the working directory to keep track of the number of tables and figures written to the rtf document, and label the captions accordingly.

References

This is a copy of the `startrtf` function from the [GLFC] package.

See Also

`heading`, `para`, `tabl`, `figu`, `endrtf`, `RTF`.

Examples

```
## Not run:
# open a Word-friendly rtf file
today <- Sys.Date()
doc <- startrtf(file=paste("Example", today))
# add headings
heading("Title")
heading(paste("Author", today, sep=" - "), 2)
# add a paragraph
para("This is how write a paragraph.")
# reference a table
para("This is how you reference a table (Table ", EchoEnv$tabcount, ").")
# add the table
tab <- matrix(sample(20), ncol=5,
 dimnames=list(paste("Row", 1:4), paste("Column", 1:5)))
tabl("A silly table.")
# reference a figure
para("And this is how you reference a figure (Figure ",
 EchoEnv$figcount, ").")
# add the figure
fig <- function() {
  par(mar=c(4, 4, 1, 1))
  plot(1:10, 1:10, xlab="X", ylab="Y")
}
figu("A silly plot.", h=4, w=4)
# save the rtf file
endrtf()

## End(Not run)
```

---

tabl *Add a Table to an RTF Document*

Description

Add a table to an rtf (rich text format) document.

Usage

```
tabl(..., TAB = tab, rtf = doc, fontt = 8, row.names = TRUE,
  tabc = EchoEnv$tabcount, boldt = TRUE, newpage = "none", omi = c(1, 1,
  1, 1))
```

Arguments

...	One or more character scalars (separated by commas) of text to use for the table caption.
TAB	A matrix, data frame, or table to be added to the document as a table, default tab.
rtf	An rtf object, default doc.
fontt	Numeric scalar font size for table caption, default 8.

row.names	Logical scalar whether to include the row.names of TAB in the table, default TRUE.
tabc	Numeric scalar table number to use in caption, default EchoEnv$tabcount.
boldt	Logical scalar indicating if table number should use bold font, default TRUE.
newpage	Character scalar indicating if the table should start on a new page in the document "port" for a new portrait page, "land" for a new landscape page, and "none" for no new page (the default).
omi	Numeric vector, length 4, width of document page margins in inches (bottom, left, top, right), default c(1, 1, 1, 1).

Details

The table and caption are written to the rtf file. The size of a new page is assumed to be 8.5 by 11 inches.

Value

A 1 is added to the numeric vector of length 1, EchoEnv$tabcount, stored in the working directory to keep track of the number of tables written to the rtf document, and label the captions accordingly.

References

This is a copy of the tabl function from the [GLFC] package.

See Also

startrtf for an example, heading, para, figu, endrtf, RTF.

Index