Package ‘micromap’
February 8, 2018

Version 1.9.3
Date 2018-02-05
Title Linked Micromap Plots
Depends R (>= 2.10), maptools, RColorBrewer, rgdal, sp
Imports ggplot2 (>= 2.0.0), grid
Description This group of functions simplifies the creation of linked micromap plots.
License GPL (>= 2)
URL http://www.jstatsoft.org/v63/i02/
Author Quinn Payton [aut],
  Tony Olsen [aut],
  Marc Weber [ctb],
  Michael McManus [ctb],
  Tom Kincaid [ctb],
  Marcus W. Beck [cre, ctb]
Maintainer Marcus W. Beck <marcusb@sccwrp.org>
NeedsCompilation no
Repository CRAN
Suggests knitr, rgeos
RoxygenNote 6.0.1
Date/Publication 2018-02-08 18:53:05 UTC

R topics documented:

alterForMedian ......................................................... 2
assimilatePlot .......................................................... 3
create_map_table ......................................................... 3
edPov ................................................................. 4
labels_att ............................................................. 5
labels_build ............................................................ 5
lmgroupedplot .......................................................... 6
Description

Adds a median line to the DF table.

Usage

alterForMedian(DF, a)

Arguments

DF the DF table created by create_DF_rank.
a the attributes list, which contains the user specified choice of whether to include a median row.

Value

Returns the DF table.

Note

See the Introduction Guide for a full list of the options available for altering micromaps.

Author(s)

Quinn Payton <Payton.Quinn@epa.gov>
assimilatePlot

Cleans a User Made Panel

Description

Sends a user made panel out to the graph "cleaning functions" in lmplot in order to properly display a user made panel seamlessly into the rest of an lmplot.

Usage

assimilatePlot(pl, i, a, limsx=NA, limsy=NA)

Arguments

pl the lmplot object.
i the panel number.
a the attribute list.
limsx limits of the x axis if desired.
limsy limits of the y axis if desired.

Value

Returns a cleaned plot object.

Note

See the Introduction Guide for a full list of the options available for altering micromaps.

Author(s)

Quinn Payton <Payton.Quinn@epa.gov>

create_map_table

Prepares A Polygon Table

Description

Takes the user supplied polygon table and prepares it to be plotted correctly with the lmplot function.

Usage

create_map_table(tmp.map, IDcolumn=NA, poly.thresh=.0001)
Arguments

- `tmp.map`: a shapefile to be parsed into a flat table for use with `ggplot2`.
- `IDcolumn`: specify which column in the data file to use as a unique identifier.
- `poly.thresh`: specify the minimum polygon area which to keep. All polygons with less area will be dropped.

Value

A flat table of all the polygons in the shapefile with an entry for each vertex.

Author(s)

Quinn Payton <Payton.Quinn@epa.gov>

Examples

```r
data("USstates")
head(USstates@data)
statePolys <- create_map_table(USstates, IDcolumn="ST")
head(statePolys)
```

---

edPov  

Example Dataset: Education and Poverty Levels

Description

A dataset comparing education and poverty levels among the states.

Usage

```r
data(edPov)
```

Format

A data frame with 51 rows and the following 5 columns:

- **state**: full state name.
- **ed**: education rate.
- **pov**: poverty rate.
- **region**: major US region.
- **StateAb**: abbreviated state name.

Note

This is just an example data set.
*labels_att*

**Examples**

```r
data(edPov)
head(edPov)
```

---

**labels_att**  
*Default Attribute Lists*

**Description**

Creates a list of default attribute lists for each panel type.

**Usage**

```r
labels_att(show=FALSE)
```

**Arguments**

- `show`: If `show` equals `FALSE`, then the resulting list is returned invisibly.

**Value**

A list of defaults to be stored in the overall attribute list.

**Author(s)**

Quinn Payton <Payton.Quinn@epa.gov>

---

**labels_build**  
*Builds panels*

**Description**

Creates a panel of the user specified type using the attribute list to make adjustments.

**Usage**

```r
labels_build(pl, p, DF, att)
```

**Arguments**

- `pl`: the `lmplot` object being constructed.
- `p`: number of the panel within the plot.
- `DF`: the data table of statistics to be referenced.
- `att`: the attribute list to specify visual and graphical characteristics.
Value
Returns a ggplot object to be printed later in the `lmplot` function.

Author(s)
Quinn Payton <Payton.Quinn@epa.gov>

### lmgroupedplot

**Linked Micromaps**

Description
Creates a linked micromap, displaying specified polygons and their associated statistical summary displays; differentiated by color.

Usage

```r
lmgroupedplot(stat.data, map.data, panel.types, panel.data, map.link = NULL, nPanels = length(panel.types), grp.by, cat, colors = brewer.pal(10, "Spectral"), map.color = "lightyellow", map.all = FALSE, print.file = "no", print.res = NA, panel.att = vector("list", nPanels), plot.header = NA, plot.header.size = NA, plot.header.color = NA, plot.footer = NA, plot.footer.size = NA, plot.footer.color = NA, plot.width = 7, plot.height = 7, map.spacing = 1, plot.grp.spacing = 1, plot.panel.spacing = 1, plot.panel.margins = c(0, 0, 1, 0), ...)
```

```r
lmplot(stat.data, map.data = NULL, panel.types, panel.data, map.link = NULL, nPanels = length(panel.types), ord.by, rev.ord = FALSE, grouping, median.row = FALSE, vertical.align = "top", median.color = gray(0.5), colors = brewer.pal(max(grouping), "Spectral"), map.all = FALSE, map.color = "lightgray", two.ended.maps = FALSE, print.file = "no", print.res = 300, panel.att = vector("list", nPanels), plot.header = NA, plot.header.size = NA, plot.header.color = NA, plot.footer = NA, plot.footer.size = NA, plot.footer.color = NA, plot.width = 7, plot.height = 7, map.spacing = 1, plot.grp.spacing = 1, plot.panel.spacing = 1, plot.panel.margins = c(0, 0, 1, 0), ...)
```

```r
mmgroupedplot(stat.data, map.data, panel.types, panel.data, map.link = NULL, nPanels = length(panel.types), grp.by, cat, colors = brewer.pal(10, "Spectral"), map.color = "lightyellow", map.all = FALSE, print.file = "no", print.res = NA, panel.att = vector("list", nPanels), plot.header = NA, plot.header.size = NA, plot.header.color = NA, plot.footer = NA, plot.footer.size = NA, plot.footer.color = NA, plot.width = 7, plot.height = 7, map.spacing = 1, plot.grp.spacing = 1, plot.panel.spacing = 1, plot.panel.margins = c(0, 0, 1, 0))
```
mmpplot(map.data, ...)

## S3 method for class 'SpatialPolygonsDataFrame'
mmpplot(map.data, ...)

## Default S3 method:
mmpplot(map.data, stat.data, panel.types, panel.data, map.link, 
nPanels = length(panel.types), ord.by, rev.ord = FALSE, grouping, 
median.row = FALSE, vertical.align = "top", median.color = gray(0.5), 
median.text.color = "black", median.text.size = 1, 
median.text.label = "Median", colors = brewer.pal(max(grouping), 
"Spectral"), map.all = FALSE, map.color2 = "lightgray", 
two.ended.maps = FALSE, trans = "identity", print.file = "no", 
print.res = 300, panel.att = vector("list", nPanels), plot.header = NA, 
plot.header.size = NA, plot.header.color = NA, plot.footer = NA, 
plot.footer.size = NA, plot.footer.color = NA, plot.width = 7, 
plot.height = 7, map.spacing = 1, plot.pGrp.spacing = 1, 
plot.panel.spacing = 1, plot.panel.margins = c(0, 0, 1, 0), ...)

### Arguments

- stat.data: table of statistics for display
- map.data: table of polygons to be associated with each item in stat.data.
- panel.types: vector of panel types to specify the layout of the plot (e.g. c('map', 'labels', 'dot.cl'))
- panel.data: a list (of lists) of data to be used with each panel (e.g. list(NA, 'Names', list('lower.bound','estimate','upper.bound'))).
- map.link: a vector with the name of the columns from stat.data and map.data, respectively, on which to join.
- nPanels: the number of panels, which is not expected to be set by the user. The default is the length of panel.types.
- grp.by: The column name from stat.data with which to order the lines of the output graphic for a standard ImPlot or identifier column on which to group the categorized lmPlot.
- cat: category column within stats table for a categorization type lmPlot.
- colors: a vector of colors for the perceptual groups. The default is brewer.pal(max(grouping), 'Spectral') for lmPlot and brewer.pal(10, 'Spectral') for lmgroupedplot. The colors are passed to colorRampPalette to create a continuous color vector equal in length to the groupings.
- map.color: the color to fill in previously displayed polygons.
- map.all: by default, lmPlot will only plot the polygons associated with data in the stats table; map.all = TRUE will show all the polygons in the polygon table regardless of whether they are actively referred to.
- print.file: name of the file being created. The extension (.pdf, .tiff, jpeg, .png) tells lmPlot which image creation tool to use.
print.res the resolution of the image to use.
panel.att a list of panel specific attributes to be altered (see lmplot documentation).
plot.header the overall title to be placed on the lmPLot.
plot.header.size size of the overall title to be placed on the lmPLot.
plot.header.color color of the overall title to be placed on the lmPLot.
plot.footer the overall footer to be placed under the lmPLot.
plot.footer.size size of the overall footer to be placed under the lmPLot.
plot.footer.color color of the overall footer to be placed under the lmPLot.
plot.width width of the overall plot in inches. Defaults to 7.
plot.height height of the overall plot in inches. Defaults to 7.
map.spacing the vertical spacing between maps measured in lines. Perceptual group spacing does not affect map spacing so as to leave the maps as large as possible. The user can increase map spacing using this argument. Defaults to 1.
plot.grp.spacing the vertical spacing between groups measured in lines. Defaults to 1.
plot.panel.spacing the vertical spacing between panels measured in lines. Defaults to 1.
plot.panel.margins the horizontal spacing between panels measured in lines. THIS IS LEGACY CODE AND SHOULD NOT BE USED.
... Additional arguments passed to or from other methods.
ord.by The column name from stat.data with which to order the lines of the output graphic for a standard lmPLot or identifier column on which to group the categorized lmPLot.
rev.ord specifies whether the plot should be displayed in reverse order of the ranking column. The default is FALSE.
grouping the number of lines per perceptual group (for the standard lmplot only). Can be a single number to have the same number in each group or a vector of numbers for unequal groupings.
median.row specifies whether a median row should be included. If an odd number of data lines are supplied, a data line itself will be used as the median, otherwise median entries will be calculated from the supplied data. Note that without a median row maps are forced into proper size.
vertical.align controls vertical alignment of the median row.
median.color specifies color of the median row.
map.color2 the color to fill in previously displayed polygons.
two.ended.maps the resulting micromaps will highlight previously referenced polygons (see map.color2) up to the median perceptual group then switch to highlighting all polygons that are still to be referenced later.
lmgroupedplot

plot.pGrp.spacing
the vertical spacing between perceptual groups measured in lines. Defaults to 1.

median.text.color
specifies color of text in the median row.

median.text.size
specifies size of text in the median row.

median.text.label
specifies the label for text in the median row.

trans
chr string for axis transformations, passed to scale_x_continuous. Acceptable values are "asn", "atanh", "boxcox", "exp", "identity", "log", "log10", "log1p", "log2", "logit", "probability", "probit", "reciprocal", "reverse" or "sqrt". One value will be recycled to all panels as needed, otherwise one per panel can be used in a combined string. Applies only to panels with axes.

Value
A list of ggplot2 objects with entries for each individual panel.

Note
See the Introduction Guide for a full list of the options available for altering micromaps.

Author(s)
Quinn Payton <Payton.Quinn@epa.gov>

Examples

# initial example

data("USstates")
head(USstates@data)
statePolys <- create_map_table(USstates, 'ST')
head(statePolys)
data("edPov")

# basic figure 1
lmplot(stat.data = edPov,
map.data = statePolys,
panel.types = c('labels', 'dot', 'dot','map'),
panel.data = list('state','pov','ed', NA),
ord.by = 'pov',
grouping = 5, median.row = TRUE,
map.link = c('StateAb','ID'))

## Not run:
# publication figure 1a
lmplot(stat.data = edPov, map.data = statePolys ,
panel.types = c('labels', 'dot', 'dot','map'),
panel.data = list('state','pov','ed', NA),
ord.by = 'pov',
grouping = 5,
  median.row = TRUE,
map.link = c('StateAb','ID'),

plot.height = 9,
colors = c('red','orange','green','blue','purple'),
map.color2 = 'lightgray',

panel.att = list(  
  list(1, header = 'States', panel.width = .8, align = 'left',
    text.size = .9),
  list(2, header = 'Percent Living Below \n Poverty Level',
    graphbgcolor = 'lightgray', point.size = 1.5,
    xaxis.ticks = list(10,15,20), xaxis.labels = list(10,15,20),
    xaxis.title = 'Percent'),
  list(3, header = 'Percent Adults With\n\n4+ Years of College',
    graphbgcolor = 'lightgray', point.size = 1.5,
    xaxis.ticks = list(0,20,30,40), xaxis.labels = list(0,20,30,40),
    xaxis.title = 'Percent'),
  list(4, header = 'Light Gray Means\nHighlighted Above',
    inactive.border.color = gray(.7), inactive.border.size = 2,
    panel.width = .8))

edPov$points <- 0

# publication figure 1b
lmplot (stat.data = edPov, map.data = statePolys,
panel.types = c('dot', 'labels', 'dot', 'dot', 'map'),
panel.data = list('points', 'state', 'pov', 'ed', NA),
map.link = c('StateAb','ID'),
ord.by = 'pov',
grouping = 5,
median.row = TRUE,

plot.height = 9,
colors = c('red','orange','green','blue','purple'),
map.color2 = 'lightgray',

panel.att = list(  
  list(1, panel.width = .15, point.type = 20,
    graph.border.color = 'white',
    xaxis.text.display = FALSE, xaxis.line.display = FALSE,
    graph.grid.major = FALSE),
  list(2, header = 'States', panel.width = .8,
    align = 'left', text.size = .9),
  list(3, header = 'Percent Living Below\nPoverty Level',
    graphbgcolor = 'lightgray', point.size = 1.5,
    xaxis.ticks = list(10,15,20),
    xaxis.labels = list(10,15,20),
    xaxis.title = 'Percent'),
list(4, header = 'Percent Adults With\n4+ Years of College',
    graphbgcolor = 'lightgray', point.size = 1.5,
xaxis.ticks = list(20, 30, 40),
xaxis.labels = list(20, 30, 40),
xaxis.title = 'Percent'),

list(5, header = 'Light Gray Means\nHighlighted Above',
inactive.border.color = gray(.7), inactive.border.size = 2,
panel.width = .8)))

# publication figure lc
myPlot <- lmplot(stat.data = edPov, map.data = statePolys,
        panel.types = c('map', 'dot', 'labels', 'dot', 'dot'),
        panel.data = list(NA, 'points', 'state', 'pov', 'ed'),
        map.link = c('StateAb', 'ID'),
        ord.by = 'pov',
        grouping = 5,
        median.row = TRUE,
        plot.height = 9,
        colors = c('red', 'orange', 'green', 'blue', 'purple'),
        map.color2 = 'lightgray',
        panel.att = list(list(2, panel.width = .15, point.type = 20,
                graph.border.color = 'white',
                xaxis.text.display = FALSE, xaxis.line.display = FALSE,
                graph.grid.major = FALSE),
        list(3, header = 'States', panel.width = .8,
            align = 'left', text.size = .9),
        list(4, header = 'Percent Living Below\nPoverty Level',
            graphbgcolor = 'lightgray', point.size = 1.5,
            xaxis.ticks = list(10, 15, 20), xaxis.labels = list(10, 15, 20),
            xaxis.title = 'Percent'),
        list(5, header = 'Percent Adults With\n4+ Years of College',
            graphbgcolor = 'lightgray', point.size = 1.5,
            xaxis.ticks = list(20, 30, 40),
            xaxis.labels = list(20, 30, 40),
            xaxis.title = 'Percent'),
        list(1, header = 'Light Gray Means\nHighlighted Above',
            inactive.border.color = gray(.7), inactive.border.size = 2,
            panel.width = .8)))

print(myPlot, name = 'myExhibit.tiff', res = 300)

## End(Not run)
Example Dataset: Lung Cancer Mortality

**Description**
Lung mortality rates for all 50 states and Washington D.C.

**Usage**
data(lungMort)

**Format**
A data frame with 51 rows and the following 14 columns:

- **StateAb**: abbreviation of state name.
- **Rate_95**: mortality rate estimate for 1995.
- **Count_95**: estimated number of cases in 1995.
- **Lower_95**: lower bound of a 95 percent confidence interval for 1995 mortality rate.
- **Upper_95**: lower bound of a 95 percent confidence interval for 1995 mortality rate.
- **Pop_95**: population in 1995.
- **StdErr_95**: estimated standard error of the estimated mortality rate in 2000.
- **Rate_00**: mortality rate estimate for 2000.
- **Count_00**: estimated number of cases in 2000.
- **Lower_00**: lower bound of a 95 percent confidence interval for 2000 mortality rate.
- **Upper_00**: lower bound of a 95 percent confidence interval for 2000 mortality rate.
- **Pop_00**: population in 2000.
- **StdErr_00**: estimated standard error of the estimated mortality rate in 2000.
- **State**: full state name.

**Note**
This is just an example data set.

**Examples**
data(lungMort)
head(lungMort)
Example Dataset: Oregon Level 3 Ecoregion Shapefile

Description
An object of class 'SpatialPolygonsDataFrame' that was created from a shapefile detailing boundaries of the level 3 ecoregion borders of Oregon.

Usage
```r
data(OrEcoLevel3)
```

Format
The format is: Formal class 'SpatialPointsDataFrame' [package "sp"].

Note
This is just an example data set.

Examples
```r
data(OrEcoLevel3)
spplot(OrEcoLevel3, zcol="US_L3NAME", col.regions=rainbow(9, s=0.75))
```

print.mm

Print Implot Object to a File

Description
Takes a stored Implot object of class mm and prints to a file whose type is implied by the file name.

Usage
```r
## S3 method for class 'mm'
print(x, name, res, ...)
```

Arguments
- `x`: name of the Implot object.
- `name`: name of the file to be produced. tiff, png, jpeg, and pdf are recognized.
- `res`: resolution in dpi.
- `...`: Additional arguments. (Currently ignored.)
**printlmplot**

*Print lmplot Object to a File*

**Description**

Takes a stored lmplot object and uses the print method for objects of class mm to print to a file whose type is implied by the file name.

**Usage**

```
printlmplot(pobject, name, res)
```

**Arguments**

- `pobject` name of the lmplot object.
- `name` name of the file to be produced. Tiff, png, jpeg, and pdf are recognized.
- `res` resolution in dpi.

**Value**

Returns nothing.

**Author(s)**

Quinn Payton <Payton.Quinn@epa.gov>

---

**right**

*Substring Function*

**Description**

Mimics the Excel Function of the Same Name

**Usage**

```
right(txt, i)
```
sample_att

Arguments

txt string to be substringed.
i number of characters to substring.

Value

A character value containing the substring.

Author(s)

Quinn Payton <Payton.Quinn@epa.gov>

Description

Creates a sample attribute list for use in the creation of new panel types.

Usage

sample_att(size=1, type=rep("standard", size), ord.by=NA, grouping=5, colors=brewer.pal(max(grouping), "Spectral"), plot.pGrp.spacing=0.05, plot.panel.margins=c(0,1,0,0), panel.data=list(NA), median.row=FALSE, show=FALSE)

Arguments

size specify the number of attributes lists to be created.
type match the attributes to which type of default panel.
ord.by argument inherited from lmplot/lmGroupedplot.
grouping argument inherited from lmplot/lmGroupedplot.
colors argument inherited from lmplot/lmGroupedplot.
plot.pGrp.spacing argument inherited from lmplot/lmGroupedplot.
plot.panel.margins argument inherited from lmplot/lmGroupedplot.
panel.data argument inherited from lmplot/lmGroupedplot.
median.row argument inherited from lmplot/lmGroupedplot.
show controls whether to print the resulting attribute list.

Value

A list of attributes to be used in the creation of a panel
Note

See the Introduction Guide for a full list of the options available for altering micromaps.

Author(s)

Quinn Payton <payton.Quinn@epa.gov>

---

**statesFlatfile**  
*Example Dataset: A Table of State Polygons*

---

**Description**

Polygon vertices for each state.

**Usage**

```r
data(statesFlatfile)
```

**Format**

A data frame with 434 rows and the following 4 columns:

- **ID**  abbreviation or state name.
- **coordsx**  x coordinates for each polygon vertice.
- **coordsy**  y coordinates for each polygon vertice.
- **poly**  individual polygon identifier within each state.

**Note**

This is just an example data set.

**Examples**

```r
data(statesFlatfile)
head(statesFlatfile)
```
Example Dataset: U.S. States Polygons

Description
An object of class 'SpatialPolygonsDataFrame' that was created from a shapefile of U.S. state borders.

Usage
data(USstates)

Format
The format is: Formal class 'SpatialPolygonsDataFrame' [package "sp"].

Note
This is just an example data set.

Examples
data(USstates)  
plot(USstates)

Example Dataset: Vegetation Coverage Percentages

Description
A subset of a larger vegetation coverage analysis dataset.

Usage
data(vegCov)

Format
A data frame with 12 rows and the following 13 columns:

Type the population name.
Subpopulation the subpopulation name, either national or level 3 ecoregion.
Indicator the ecological variable (indicator)
Category category of disturbance.
NResp sample size.
**Estimate.P** estimated percentage of vegetation coverage with a given classification.

**StdError.P** standard error of the percentage estimate.

**LCB95Pct.P** lower bound of a 95 percent confidence interval for the estimated percentage.

**UCB95Pct.P** upper bound of a 95 percent confidence interval for the estimated percentage.

**Estimate.U** estimated area of vegetation coverage with a given classification.

**StdError.U** standard error of the area estimate.

**LCB95Pct.U** lower bound of a 95 percent confidence interval for the estimated area.

**UCB95Pct.U** upper bound of a 95 percent confidence interval for the estimated area.

**Note**

This is just an example data set.

**Examples**

```r
data(vegCov)
head(vegCov)
```

---

**WSA3**

*Example Dataset: Major U.S. EcoRegions*

**Description**

An object of class 'SpatialPolygonsDataFrame' that was created from a shapefile of the U.S. level 3 ecoregion boundaries.

**Usage**

```r
data(WSA3)
```

**Format**

The format is: Formal class 'SpatialPointsDataFrame' [package "sp"].

**Note**

This is just an example data set.

**Examples**

```r
data(WSA3)
spplot(WSA3, zcol="WSA_3_NM", col.regions=rainbow(3, s=0.75))
```
Index

*Topic **datasets**
  edPov, 4
  lungMort, 12
  OrEcoLevel3, 13
  statesFlatfile, 16
  USstates, 17
  vegCov, 17
  WSA3, 18

*Topic **hplot**
  lmgroupedplot, 6

alterForMedian, 2
assimilatePlot, 3

bar_att (labels_att), 5
bar_build (labels_build), 5
bar_cl_att (labels_att), 5
bar_cl_build (labels_build), 5
box_summary_att (labels_att), 5
box_summary_build (labels_build), 5

colorRampPalette, 7
create_map_table, 3

dot_att (labels_att), 5
dot_build (labels_build), 5
dot_cl_att (labels_att), 5
dot_cl_build (labels_build), 5

edPov, 4

labels_att, 5
labels_build, 5
lmgroupedplot, 6
lmplot (lmgroupedplot), 6
lungMort, 12

map_att (labels_att), 5
migroupedplot (lmgroupedplot), 6
mmplot (lmgroupedplot), 6

OrEcoLevel3, 13
print.mm, 13
printLMPlot, 14
ranks_att (labels_att), 5
ranks_build (labels_build), 5
right, 14
sample_att, 15
scale_x_continuous, 9
standard_att (labels_att), 5
statesFlatfile, 16

USstates, 17
vegCov, 17
WSA3, 18