Package 'MapGAM'

May 3, 2015

Type Package

Title Mapping Smoothed Effect Estimates from Individual-Level Data

Version 0.7-5

Date 2015-05-02

Author Veronica Vieira, Scott Bartell, and Robin Bliss

Maintainer Scott Bartell <sbartell@uci.edu>

Description Contains functions for mapping odds ratios or other effect estimates using individual-level data such as case-control study data, using generalized additive models (GAMs) for smoothing with a two-dimensional predictor (e.g., geolocation or exposure to chemical mixtures) while adjusting for confounding variables, using methods described by Kelsall and Diggle (1998) and Webster at al. (2006). Includes convenient functions for mapping, efficient control sampling, and permutation tests for the null hypothesis that the two-dimensional predictor is not associated with the outcome variable (adjusting for confounders).

License GPL-3

Depends R (>= 2.10.0), sp, gam

Imports maptools

Suggests maps, mapproj, PBSmapping

NeedsCompilation no

Repository CRAN

Date/Publication 2015-05-03 07:07:27

R topics documented:

`MapGAM-package` *Mapping Smoothed Effect Estimates from Individual-Level Spatial Data*

Description

Contains functions for mapping odds ratios or other effect estimates using individual-level data such as case-control study data, using generalized additive models (GAMs) for smoothing with a two-dimensional predictor (e.g., geolocation or exposure to chemical mixtures) while adjusting for confounding variables, using methods described by Kelsall and Diggle (1998) and Webster at al. (2006). Includes convenient functions for mapping, efficient control sampling, and permutation tests for the null hypothesis that the two-dimensional predictor is not associated with the outcome variable (adjusting for confounders).

Details

Package:	MapGAM
Type:	Package
Version:	0.7-5
Date:	2014-05-02
License:	GPL-3

Typical spatial applications will start with the predgrid function to create a regular grid of points within the study area, with optional map boundaries (e.g., a country, state, or regional map). Crude or adjusted odds ratios (or linear predictors) are then estimated at each grid point using the modgam function to smooth by geolocation. Finally, the predicted values (and optionally, "clusters"–areas of signficantly increased or decreased values determined via permutation tests) are plotted using the colormap function. The trimdata and sampcont functions can be used to restrict data to those within map boundaries and to conduct simple or spatiotemporal stratified sampling from eligible controls. The optspan function can be used to find an optimal span size for the LOESS smoother used by the modgam function; it is automatically used within the modgam function when the span size is not provided by the user. These functions can also be applied to non-spatial data when two-dimensional smoothing is of interest, such as investigation of the effects of a mixture of two chemicals.

Author(s)

Veronica Vieira, Scott Bartell, and Robin Bliss

Send bug reports to <sbartell@uci.edu>.

References

Hastie TJ, Tibshirani RJ. Generalized Additive Models. (Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Boca Raton, Florida, 1990).

Kelsall J, Diggle P. Spatial variation in risk of disease: a nonparametric binary regression approach. J Roy Stat Soc C-App 1998, 47:559-573.

Vieira V, Webster T, Weinberg J, Aschengrau A, Ozonoff D. Spatial analysis of lung, colorectal, and breast cancer on Cape Cod: An application of generalized additive models to case-control data. Environmental Health 2005, 4:11.

Webster T, Vieira V, Weinberg J, Aschengrau A. Method for mapping population-based case-control studies using Generalized Additive Models. International Journal of Health Geographics 2006, 5:26.

Young RL, Weinberg J, Vieira V, Ozonoff A, Webster TF. A power comparison of generalized additive models and the spatial scan statistic in a case-control setting. International Journal of Health Geographics 2010, 9:37.

http://www.busrp.org/projects/project-2-analyzing-patterns-in-epidemiologic-and-toxicologic-data

See Also

trimdata, sampcont, predgrid, optspan, modgam, colormap.

Examples

```
# Load synthetic data and a preformatted base map
data(MAmap)
data(MAdata)

# Create a grid on the base map (PBSmapping package recommended)
if(require(PBSmapping)) gamgrid <- predgrid(MAdata, MAmap) else
 gamgrid <- predgrid(MAdata)
# Fit a GAM with a smooth term for spatial location
fit1 <- modgam(MAdata, gamgrid, m="crude", sp=0.5)
# Display odds ratio estimates on the base map
colormap(fit1, MAmap)

#### See colormap and modgam help files for more examples
```

---

beertweets *Geocoded Tweets with Beer Indicator*

Description

Geocoded tweets from Twitter, with an indicator variable for any mention of beer, time stamp, and state of origin

Usage

```
data(MAmap)
```

Format

A data frame with 10000 observations on 5 variables:

`beer` 1 for tweets about beer, 0 for other tweets

`longitude` geocoded longitude

`latitude` geocoded latitude

`state` a factor with the name of the state

`time` a list of POSIXlt format dates and times for the tweets

Details

A sample of geocoded tweets from within in the contiguous US, from June to October of 2012. Tweets mentioning beer (cases) are oversampled by a factor of 100. Geocoding is typically at the level of city or town; tweets that could not be geocoded were excluded from this data set.

Source

Dr. Matthew Zook, University of Kentucky

http://www.floatingsheep.org/2012/07/church-or-beer-americans-on-twitter.html

Examples

```
data(MAmap)
plot(MAmap)
```

---

colormap *Maps Predicted Values and Clusters for modgam Objects*

Description

Displays a color image map, including a legend, scale bar, and optional North arrow, showing crude or adjusted odds ratios (or linear predictors) for a grid of points. Irregular grids are allowed. Also draws contour lines for regions of signficantly increased or decreased values of the outcome variable ("clusters"), if permutation ranks are provided. Designed to display `modgam` objects but can be used with other model results if the modgamobj list contains the correct elements.

Usage

```
colormap(modgamobj, map = NULL, add=F, contours="none", mapmin = NULL, mapmax = NULL,
         arrow = T, axes = F, ptsize = 0.9, alpha = 0.05)
```

Arguments

modgamobj	(Required) A list containing at least these two elements: "grid" (a 2 column data frame with X and Y coordinates) and "fit" (a vector of linear predictors, with length equal to the number of grid points). If the list contains an element named "OR" (a vector of odds ratios, with length equal to the number of grid points) then "OR" will be used instead of "fit". If the list contains the element "pointwise" (a vector of percentile ranks generated by permutation test) then those values will be used to generate contours. The correct list format is provided as output from the modgam function (see examples).
map	Can be used to map predicted values on a base map from the map function in the **maps** package, or on a base map produced by the readShapePoly function in **maptools** package. ReadShapePoly reads maps from external files in the ESRI shapefile format. map=NULL produces a color image without any base map.
add	Use add=T to add the color map to an existing plot. This will often result in loss of the legend and scale, which are added outside of the normal map boundaries. add is ignored when a map is provided using the map argument.
contours	Use contours="response" to add contour lines for the predicted response, for example to draw isoboles for mixtures of exposures. Use contours="permrank" to add contour lines for pointwise p-values computed from the permutation ranks, at alpha/2 and (1-alpha)/2. The default is "none" which produces no contour lines.
mapmin	The minimum value for the color scale legend
mapmax	The maximum value for the color scale legend
arrow	Use arrow=T to add a North arrow to the map.
axes	Use axes=T to add axes to the map (useful for chemical mixture isoboles).
ptsize	The size of the points used to fill in colors on the map. Increase to remove white space inside the map or decrease to remove color outside the map boundaries. NOTE: white space can also be eliminated by increasing the grid size in predgrid, which is often preferable as it results in a higher resolution map.
alpha	The nominal pointwise type I error rate; only used when contours="permrank".

Value

Produces an image map showing crude or adjusted linear predictors (or odds ratios). If the base map is in readShapePoly format a scale bar is included. The scale bar assumes that the X and Y coordinates are provided in meters.

Warning

Note that the contour lines use a pointwise nominal type I error rate of alpha; the chance of a type I error occurring for **at least** one point on the map is much higher, typically approaching 100% at alpha=0.05, because a spatial prediction grid generally contains many points.

Author(s)

Veronica Vieira, Scott Bartell, and Robin Bliss

Send bug reports to <sbartell@uci.edu>.

See Also

trimdata, predgrid, optspan, modgam.

Examples

```
### Load readShapePoly base map and data
data(MAmap)
data(MAdata)
# Create a grid on the base map (PBSmapping package recommended)
if(require(PBSmapping)) MAgrid <- predgrid(MAdata, MAmap) else
MAgrid <- predgrid(MAdata)
# fit crude GAM model to the MA data using span size of 50%
fit1 <- modgam(MAdata, MAgrid, m="crude", sp=0.5)
# Plot a map showing crude odds ratios
colormap(fit1, MAmap)

#### A detailed example including map projections and data trimming
# NOTE: this example requires the maps, mapproj, and PBSmapping packages
# Convert base map and beer tweet data locations to US Albers projection
# projected coords yield better distance estimates than (lat,long)
if(require(maps) & require(mapproj) & require(PBSmapping)) {
USmap <- map("state",projection="albers",parameters=c(29.5,45.5),
plot=FALSE,fill=TRUE,col="transparent")
data(beertweets)
case <- beertweets$beer
    # Reuse last map projection to convert data coordinates
XY <- mapproject(beertweets$longitude,beertweets$latitude)[1:2]
beerproj <- data.frame(case, XY[1], XY[2])
# Generate grid on the US map, trimmed to range of beer data
USgrid <- predgrid(beerproj, USmap)
    # Fit unadjusted model--geolocation only
fit2 <- modgam(beerproj, USgrid, m="unadjusted", sp=0.2)
dev.new(width=7,height=5)
    colormap(fit2, USmap)
title(main="Beer Tweet Odds Ratios")
}
```

MAdata *Synthetic Case-Control Data for Massachusetts*

Description

90 cases and 910 controls with random smoking covarate values and random geolocations within Massachusetts, geocoded on a Lambert projection (in meters). MAmap is a map of Massachusetts using the same projection.

Usage

```
data(MAdata)
```

Format

A data frame with 1000 observations on the following 6 variables.

`Case` 0 for controls, 1 for cases

`Xcoord` projected X coordinate

`Ycoord` projected Y coordinate

`Smoking` 0 for nonsmokers, 1 for smokers

`mercury` continuous variable for mercury exposure

`selenium` continuous variable for selenium exposure

Details

Lambert conformal conic projection for the State of Massachusetts, using standard parallels 41.71666667 and 42.68333333. The latitude of origin is 41.0, the central meridian is -71.5, and the projection units are meters (False Easting: 200000 m; False Northing: 750000 m).

Source

2010 ISEE/ISES GamMAP workshop: http://www.cireeh.org/pmwiki.php/Main/Gam-mapWorkshop

http://www.busrp.org/projects/project-2-analyzing-patterns-in-epidemiologic-and-toxicologic-data (Mercury and selenium exposure variables added in December 2013)

Examples

```
data(MAdata)
summary(MAdata)
attach(MAdata)
# map participants, cases in red and controls in black
plot(Xcoord,Ycoord,col=Case+1)
```

MAmap *Map of Massachusetts*

Description

A map of the outline of MA in SpatialPolygonsDataFrame format, converted from an ESRI shapefile using the `readShapePoly` function in the **maptools** package.

Usage

```
data(MAmap)
```

Format

The format is class SpatialPolygonsDataFrame (package "sp")

Details

Lambert conformal conic projection for the State of Massachusetts, using standard parallels 41.71666667 and 42.68333333. The latitude of origin is 41.0, the central meridian is -71.5, and the projection units are meters (False Easting: 200000 m; False Northing: 750000 m).

Source

Dr. Veronica Vieira, University of California, Irvine

Examples

```
data(MAmap)
plot(MAmap)
```

---

modgam — *Fit a Generalized Additive Model with a Two-Dimensional Smooth (GAM)*

Description

Fits a crude or adjusted regression on a user-supplied grid for spatial analysis using a generalized additive model with a two-dimensional LOESS smooth for geolocation (or any other two-dimensional predictor). Includes optional permutation tests for global and local tests of the null hypothesis that the two-dimensional predictor (e.g., geolocation) is not associated with the outcome. Most applications will pass the output of this function to `colormap` to map the resulting odds ratios or other effect estimates.

Usage

```
modgam(rdata, rgrid, family=binomial, permute = 0, conditional = TRUE,
       m = "adjusted", sp = NULL, keep = FALSE, verbose = TRUE, ...)
```

Arguments

`rdata`	Data set (required). The data must be structured so that the outcome is in the 1st column and the X and Y coordinates for two-dimensional predictor (e.g., geolocation) are in the 2nd and 3rd columns, respectively. If more than three columns are provided and `m = "adjusted"`, the additional columns will be entered as linear predictors in the GAM model.
`rgrid`	A data frame containing the values at which to generate predictions (required). X and Y coordinates for the two-dimensional predictor must be in the 1st and 2nd columns. Additional covariates values for predictions may be provided in additional columns of rgrid. If `m = "adjusted"` and rdata includes covariates

not present in rgrid, then for each missing covariate the median value will be taken from rdata. The `predgrid` function can be used to supply an appropriate rgrid for rdata, with the grid clipped according to any specified base map (see examples below).

`family` A description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See `family` for details of family functions. Note that unlike other packages, MapGAM defaults to the binomial family and logit link, i.e., a logistic model.)

`permute` The number of permutations of the data set for testing the significance of the two-dimensional predictor. `permute = 0` (default) produces no permutation tests. If `permute` > 0, the paired coordinates for the two-dimensional predictor are randomly permuted in order to simulate the distribution of results under the null hypothesis that location is not associated with the outcome. 1000 permutations are recommended for reasonable accuracy of p-values. WARNING: Because each permutation requires refitting the GAM, permutation tests can be quite slow.

`conditional` Logical value indicating whether to run a conditional or unconditional permutation test; this argument is used only when `permute` > 0. The default is a conditional permutation test: the span size is held fixed throughout all permutations even if `sp = NULL` to optimize the span size for the original data (see Young et al., 2012). The unconditional permutation test repeats the span size optimization for each permutation, which is more conservative but takes about 20 times longer to compute. If `conditional = FALSE` the sp argument is ignored when fitting both the original and permuted data sets.

`m` Model type for the GAM. Options are "crude" or "adjusted" (default). If "crude", only the smooth for the two-dimensional predictor is included in the model (i.e., "crude" is a synonym for "unadjusted"). If "adjusted", all covariates in the data set (columns >= 4) are included in the model.

`sp` Span size for the LOESS smooth of the two-dimensional predictor. If `sp = NULL` (default) then an optimal span size will be determined using the optspan function.

`keep` Logical value indicating whether to store and return the pointwise odds ratios, and if conditional = FALSE the span size, for every permuted data set. These values aren't necessary for mapping or cluster identification, and storing them slows the permutation tests, so the default is `keep = FALSE`.

`verbose` Logical value indicating whether to print the GAM model statement, the percentile rank for the global deviance statistic in the permutation test, and the progress of the permutation test (report completion of every 10 permutations). The default is `verbose = TRUE`.

`...` Further arguments to be passed to the `gam` function (e.g., weights).

Details

The model used to fit the data is a generalized additive model with a LOESS smooth for a two-dimensional predictor such as geolocation (Hastie and Tibshirani, 1990; Kelsall and Diggle, 1998;

Webster et al., 2006). Although any family and link function supported by the gam function is supported, the default binomial family with logit link yields the following model:

$$\ln\left(\frac{\pi_i}{1-\pi_i}\right) = S(x_i, y_i) + \mathbf{Z_i}\boldsymbol{\beta}$$

where π_i is the probability that the outcome is 1 for participant *i*, x_i and y_i are predictor coordinates for participant *i* (i.e., projected distance east and north, respectively, from an arbitrarily defined origin location), $S(.,.)$ is a 2-dimensional smoothing function (currently LOESS), $\mathbf{Z_i}$ is a row vector of covariate values for participant *i*, and $\boldsymbol{\beta}$ is a vector of unknown regression parameters (including an intercept). When a permutation test is requested, for each permutation the paired X and Y coordinates in the data set are randomly reassigned to participants, consistent with the null hypothesis that the geolocation (or another two-dimensional predictor entered in place of X and Y) is not associated with the outcome. Note that this procedure intentionally preserves associations between other covariates and the outcome variable so the permutation test reflects the significance of geolocation. See the references for more details.

Value

`grid` A data frame with X and Y coordinates from rgrid.

`m` Whether the GAM was unadjusted (only spatial location as a predictor) or adjusted (included other covariates in addition to spatial location). "Crude" is a synonym for "unadjusted."

`span` The span size used for the LOESS smooth. If `keep = TRUE` and `conditional = FALSE`, a vector with optimized span sizes for the original data set and each of the permuted data sets.

`gamobj` The GAM model object from the fit to the data.

`family` The family and link function used for the model.

`fit` Predicted values (on the linear predictor scale) for each point on the grid, from the smoothed GAM model. For example, for a binomial family logit link model the fit values are the log odds of the outcome. Predicted responses can be obtained from the fit values by applying the inverse link function.

`OR` Odds ratios for each grid point, comparing fits from the smoothed GAM model with a referent model without the two-dimensional predictor but otherwise identical to the GAM. These are only provided when the binomial family and logit link are selected.

If `permute > 0` then the following values are also provided:

`global` The p-value based on the deviance statistic comparing models with and without geolocation (or other two-dimensional predictor), using the distribution of deviance statistics from the permuted data sets to represent the null distribution. For a test of H0: geolocation is unassociated with the outcome variable (adjusting for any covariates if `m = "adjusted"`), reject H0 if the percentile rank is below alpha. WARNING: by default `modgam` uses a conditional permutation test which produces inflated type I error rates; Young et al. (2012) recommend using alpha=0.025 to limit the type I error rate to approximately 5%.

`pointwise` For each point on the grid, the percentile rank of the local linear predictor for the model compared to the local linear predictor distribution from the permuted data sets. This result is needed to define clusters of the outcome (e.g., spatial regions with statistically significant outcomes–e.g., unusually high or low risks.)

If `permute` > 0 and `keep = T` then the following values are also provided:

`permutations` A matrix containing null permuted values for each point on the grid, with the results for each permutation in a separate column. For the binomial family and logit link these are provided as odds ratios, otherwise they are reported as linear predictors.

Warning

Permutation tests are computationally intensive, often requiring several hours or more.

Author(s)

Veronica Vieira, Scott Bartell, and Robin Bliss

Send bug reports to <sbartell@uci.edu>.

References

Hastie TJ, Tibshirani RJ. Generalized Additive Models. (Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Boca Raton, Florida, 1990).

Kelsall J, Diggle P. Spatial variation in risk of disease: a nonparametric binary regression approach. J Roy Stat Soc C-App 1998, 47:559-573.

Vieira V, Webster T, Weinberg J, Aschengrau A, Ozonoff D. Spatial analysis of lung, colorectal, and breast cancer on Cape Cod: An application of generalized additive models to case-control data. Environmental Health 2005, 4:11.

Webster T, Vieira V, Weinberg J, Aschengrau A. Method for mapping population-based case-control studies using Generalized Additive Models. International Journal of Health Geographics 2006, 5:26.

Young RL, Weinberg J, Vieira V, Ozonoff A, Webster TF. A power comparison of generalized additive models and the spatial scan statistic in a case-control setting. International Journal of Health Geographics 2010, 9:37.

See Also

`predgrid`, `optspan`, `colormap`, `readShapePoly`.

Examples

```
# Load base map in SpatialPolygonsDataFrame format
# This map was read from ESRI shapefiles using the readShapePoly function
data(MAmap)
# Load data and create grid on base map
data(MAdata)
```

```
gamgrid <- predgrid(MAdata, MAmap)    # requires PBSmapping package
# Fit crude logistic GAM to the MA data using span size of 50%
# and predict odds ratios for every point on gamgrid
fit1 <- modgam(MAdata, gamgrid, m="crude", sp=0.5)
# Summary statistics for pointwise crude odds ratios
summary(fit1$OR)
# Summary stats for pointwise crude log odds (linear predictor)
summary(fit1$fit)

# fit adjusted GAM using span size of 50%,
# including a (too small) conditional permutation test
fit2 <- modgam(MAdata, gamgrid, permute=25, m="adjusted", sp=0.5)

# Detailed example with a continuous outcome variable, map projections,
# and data trimming:  investigating tweet times by geolocation
# NOTE: this example requires the maps, mapproj, and PBSmapping packages
# Convert base map and beer tweet data locations to US Albers projection
# for better distance estimates than using (lat,long) as (X,Y) coords
if(require(maps) & require(mapproj) & require(PBSmapping)) {
USmap <- map("state",projection="albers",parameters=c(29.5,45.5),
plot=FALSE,fill=TRUE,col="transparent")
data(beertweets)
# Reuse last map projection to convert data coordinates
XY <- mapproject(beertweets$longitude,beertweets$latitude)[1:2]
jtime <- julian(beertweets$time)
# Convert tweet dates and times to time of day (24-hour)
tweettime <- as.numeric(jtime-trunc(jtime))*24
beerproj <- data.frame(tweettime, XY[1], XY[2], beertweets$beer)
# Generate grid on the US map, trimmed to range of beer data
USgrid <- predgrid(beerproj, USmap)
# Fit adjusted model--adjusting for beer indicator variable
fit3 <- modgam(beerproj, USgrid, family=gaussian, m="adjusted", sp=0.2)
# Get summary statistics for predicted tweet times across grid points
summary(fit3$fit)
}

# Smoothing for two-dimensional chemical exposure instead of geolocation
# case status in 1st column, mercury and selenium in 2nd and 3rd columns
ma2 <- MAdata[,c(1,5:6)]
expgrid <- predgrid(ma2)
fit4 <- modgam(ma2,expgrid,sp=.5,m="crude")
summary(fit4$OR)
# plot the results, with mercury on the X axis and selenium on the Y axis
colormap(fit4, contours="response", arrow=FALSE, axes=TRUE)
```

---

optspan *Determine the Optimal Span Size for modgam*

Description

Determines the optimal span size for modgam, a spatial generalized additive model (GAM) with a two-dimensional LOESS smooth for location, by minimizing the AIC. Currently evaulates span sizes from 5% to 95% in increments of 5%.

Usage

```
optspan(rdata, m = "adjusted", family=binomial(), verbose = TRUE, ...)
```

Arguments

rdata	Data set (required). The data must be structured so that the outcome variable is in the 1st column and X and Y location values are in the 2nd and 3rd columns. Any additional columns will be entered as covariates with linear effects in the gam model.
m	Model type. Options are "crude" or "adjusted" (default). If "crude", only the spatial smooth term for location is included in the model. If "adjusted", all covariates in the data set (columns >= 4) are included in the model.
family	A family function or the result of a call to a family function. (See family for details of family functions.
verbose	Logical argument; if TRUE shows the AIC for each candidate span size.
...	Any additional arguments to pass to the gam function.

Value

The optimal span size, determined by minimizing the AIC

Note

This function does not return model predictions–only the optimal span size. To obtain model predictions use the modgam function.

Author(s)

Veronica Vieira, Scott Bartell, and Robin Bliss

Send bug reports to <sbartell@uci.edu>.

See Also

predgrid, modgam, colormap.

Examples

```
data(MAdata)
optspan(MAdata, m="crude")
```

predgrid *Create a Grid and Clip it to a Map and Data Bounds*

Description

Creates a data frame containing a rectangular grid of points to cover the range of X and Y coordinates provided in a data set, and trims the grid so that the points do not extend beyond the boundaries shown on a map.

Usage

```
predgrid(dataXY, map = NULL, nrow = 100, ncol = 100)
```

Arguments

dataXY	A data frame with X and Y coordinates (required). If the data frame has more than 2 columns, the X and Y coordinates must be in the 2nd and 3rd columns, respectively (the format required for the modgam function). If the data frame has only 2 columns, the X coordinates must be in the 1st column and Y coordinates in the 2nd column.
map	A map for clipping the grid, provided in a recognized format. Supported classes include "map", the format produced by the maps package, and "SpatialPolygonsDataFrame", which can be produced from an ESRI shapefile using the readShapePoly function in the maptools package. If map = NULL (default), a rectangular grid is produced covering the range of geolocations within dataXY.
nrow	number of rows in the grid (default=100)
ncol	number of columns in the grid (default=100)

Details

predgrid creates a grid of dimensions nrow*ncol using the range of X and Y coordinates (e.g., longitude and latitude) in the data frame supplied as dataXY. If the map argument is used, the function trimdata is used to clip the grid. Users should be sure to use the same projection for the map and the data; putting both on the same plot can help reveal differing projections. If the map centroid is not in the range of the data a warning message is printed; this might indicate differing projections but can occur naturally when the data were not sampled from the entire extent of the map or when map boundaries are concave.

Value

A data frame with X and Y coordinates in the first two columns. The column names from dataXY are used for the output. If the columns of dataXY are unnamed then the names "X" and "Y" are assigned to the data frame.

Author(s)

Veronica Vieira, Scott Bartell, and Robin Bliss

Send bug reports to <sbartell@uci.edu>.

See Also

trimdata, optspan, modgam, colormap.

Examples

```
# define a rectangular 100x100 grid covering the MA data
data(MAdata)
gamgrid <- predgrid(MAdata)
# plot the grid points
plot(gamgrid$Xcoord, gamgrid$Ycoord, cex=0.1, col="red")
# and the data locations
points(MAdata$Xcoord,MAdata$Ycoord)

# But that grid extends beyond the state boundaries and into the ocean!
# Better to also clip the grid to a map of MA using the following code
# which requires the PBSmapping package:
if (require(PBSmapping)) {
# Clip a 50x50 grid covering the MA data to a map of MA
data(MAmap)
gamgrid2 <- predgrid(MAdata, map=MAmap, nrow=50, ncol=50)
# plot the MA map and grid points
plot(MAmap)
points(gamgrid2$Xcoord, gamgrid2$Ycoord, cex=0.1, col="red")
}
```

---

sampcont *Unmatched Control Sampling*

Description

Take all cases and a random sample of controls from a data frame. Simple random sampling and stratified random sampling are available. For statified random sampling, strata can be defined by region, or by region and time. If no specific regions are specified then the function will create a regular grid for sampling.

Usage

```
sampcont(rdata, type = "stratified", regions = NULL, times = NULL, n = 1,
         nrow = 100, ncol = 100)
```

Arguments

rdata	A data frame with the outcome (coded as 0/1) in the 1st column, and the geocoordinates (e.g., X and Y) in the 2nd and 3rd columns. Additional columns are not used in the sampling scheme but are retained in the sampled data frame.
type	"stratified" (default) or "simple". If "simple" then a simple random sample of n controls (rows of rdata with outcome=0) is obtained. If "stratified" then a stratified random sample of controls is obtained, with up to n controls per stratum. Sampling strata are defined by the regions and times arguments. All cases (rows with outcome=1) are taken for the sample regardless of the value supplied for type.
regions	A vector of length equal to the number of rows in rdata, used to construct sampling strata. Only used if type = "stratified". If regions = NULL and the PBSmapping package is available then the function will define regions as a vector of specific grid cells on a regular grid with nrow rows and ncol columns. If times = NULL then the nonempty regions are used as the sampling strata. If times is a vector, then the sampling strata are all nonempty combinations of regions and times.
times	A vector of length equal to the number of rows in rdata, used to construct sampling strata. If times = NULL then the sampling strata are defined only by the regions argument. If times is a vector, then the sampling strata are all nonempty combinations of regions and times. Continuous times should generally be binned before being passed through this argument, as there are no efficiency gains if each value in times is unique.
n	The number of controls to sample from the eligible controls in each stratum. All available controls will be taken for strata with fewer than n eligible controls.
nrow	The number of rows used to create a regular grid for sampling regions. Only used when regions = NULL.
ncol	The number of columns used to create a regular grid for sampling regions. Only used when regions = NULL.

Value

rdata	A data frame with all cases and a random sample of controls.
w	Inverse probability weights for the rows in rdata. Important to include as weights in subsequent analyses.
ncont	The total number of controls in the sample.

Author(s)

Scott Bartell <sbartell@uci.edu>.

See Also

modgam

Examples

```
#### load beertweets data, which has 719 cases and 9281 controls
data(beertweets)
# take a simple random sample of 1000 controls
samp1 <- sampcont(beertweets, type="simple", n=1000)

# take a stratified random sample of controls on a 80x50 grid
# requires PBSmapping package
samp2 <- NULL

if(require(PBSmapping)) samp2 <- sampcont(beertweets, nrow=80, ncol=50)

# Compare locations for the two sampling designs (cases in red)
par(mfrow=c(2,1), mar=c(0,3,4,3))
plot(samp1$rdata$longitude, samp1$rdata$latitude, col=3-samp1$rdata$beer,
cex=0.5, type="p", axes=FALSE, ann=FALSE)
# Show US base map if maps package is available
mapUS <- require(maps)
if (mapUS) map("state", add=TRUE)
title("Simple Random Sample, 1000 Controls")

if (!is.null(samp2)) {
plot(samp2$rdata$longitude, samp2$rdata$latitude,
col=3-samp2$rdata$beer, cex=0.5, type="p", axes=FALSE,
ann=FALSE)
if (mapUS) map("state", add=TRUE)
title(paste("Spatially Stratified Sample,",samp2$ncont,"Controls"))
}

par(mfrow=c(1,1))

## Note that weights are needed in statistical analyses
# Prevalence of cases in sample--not in source data
mean(samp1$rdata$beer)
# Estimated prevalence of cases in source data
weighted.mean(samp1$rdata$beer, w=samp1$w)
## Do beer tweet odds differ below the 36.5 degree parallel?
# Using full data
glm(beer~I(latitude<36.5), family=binomial, data=beertweets)
# Stratified sample requires sampling weights
if (!is.null(samp2)) glm(beer~I(latitude<36.5), family=binomial,
data=samp2$rdata, weights=samp2$w)
```

`trimdata` *Trim a Data Set To Map Boundaries*

Description

Takes a subset of a data frame: returns rows with geolocations inside the boundaries determined by a map. If `rectangle=FALSE`, strict map boundaries are used. If `rectangle=TRUE`, a rectangular

boundary is determined based on the range of X and Y coordinates for the map, along with an optional buffer.

Usage

```
trimdata(rdata, map, Xcol=2, Ycol=3, rectangle=F, buffer=0.05)
```

Arguments

rdata	Data set (required). If rdata has only 2 columns then they are assumed to be the X and Y coordinates, in that order. If rdata has more than 2 columns then identify the positions of the X and Y coordinates with Xcol and Ycol, respectively.
map	A map for clipping the grid, provided in a recognized format. Supported classes include "map", the format produced by the maps package, and "SpatialPolygonsDataFrame", which can be produced from an ESRI shapefile using the readShapePoly function in the maptools package.
Xcol	The column number of rdata for the X coordinates of geolocation. Only used if rdata has >2 columns.
Ycol	The column number of rdata for the Y coordinates of geolocation. Only used if rdata has >2 columns.
rectangle	If rectangle=FALSE (default), only data with geolocations strictly within the map boundaries are retained. If rectangle=TRUE, only data with geolocations within a rectangular boundary are retained. The rectangular boundary coordinates are determined using the ranges of X and Y coordinates for the map, along with an optional buffer.
buffer	A fraction of the map range to add to either side of the rectangular boundary before trimming the data (default 5%). The buffer argument is ignored if rectangle=FALSE. If a vector of length 2 is supplied, the first value as the buffer fraction for the X coordinate range and the second value is used as the buffer fraction for the Y coordinate range.

Details

Various functions from the PBSmapping package and maptools package are used to convert map formats, check whether the map and data are aligned, and clip the grid. Without the PBSmapping package only rectangle=TRUE will work. Be sure to use fill=T when using the map function to produce maps for trimdata: maps produced using the default fill=F argument do not work properly with this function. If the map centroid is not in the range of the data and the PBSmapping package is available a warning message is printed; this might indicate differing projections but can occur naturally when the data were not sampled from the entire extent of the map or when map boundaries are concave.

Value

A subset of the rdata data frame containing only those rows with geolocations within the specified boundaries.

Author(s)

Veronica Vieira, Scott Bartell, and Robin Bliss

Send bug reports to <sbartell@uci.edu>.

See Also

predgrid, optspan, modgam, colormap.

Examples

```
# These examples require the "PBSmapping" and "maps" packages
if (require(maps) & require(PBSmapping)) {
data(beertweets)
dim(beertweets)

### Trim data to US base map, and plot them
basemap1 <- map("usa", fill=TRUE, col="transparent")
dUS <- trimdata(beertweets, basemap1)
# Plot tweet locations (beer tweets in red)
points(dUS$longitude, dUS$latitude, col=dUS$beer+1, cex=0.5)

### Trim data to Texas base map, and plot them
basemap2 <- map("state", regions="texas", fill=TRUE, col="transparent")
dTX <- trimdata(beertweets, basemap2)
# Plot tweet locations (beer tweets in red)
points(dTX$longitude, dTX$latitude, col=dTX$beer+1, cex=0.5)
}
```

Index