Package ‘sptotal’

July 1, 2020

Type Package

Title Predicting Totals and Weighted Sums from Spatial Data

Date 2020-07-01

Version 0.1.0

Depends R (>= 3.5.0)

Description Performs predictions of totals and weighted sums, or finite population block kriging, on spatial data using the methods in Ver Hoef (2008) <doi:10.1007/s10651-007-0035-y>. The primary outputs are an estimate of the total, mean, or weighted sum in the region, an estimated prediction variance, and a plot of the predicted and observed values. This is useful primarily to users with ecological data that are counts or densities measured on some sites in a finite area of interest. Spatial prediction for the total count or average density in the entire region can then be done using the functions in this package.

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

BugReports https://github.com/highamm/sptotal/issues

Imports matrixcalc, mvtnorm, ggplot2, stats, viridis, graphics, tibble, sp

Suggests gstat, rmarkdown, knitr, testthat, prettydoc, rgeos

VignetteBuilder knitr, rmarkdown

NeedsCompilation no

Author Higham Matt [cre, aut],
Ver Hoef Jay [aut],
Frank Bryce [aut]

Maintainer Higham Matt <mhigham@stlawu.edu>

Repository CRAN

Date/Publication 2020-07-01 13:20:02 UTC
**R topics documented:**

- AIC.slmfit ................................................................. 2
- AKmoose .................................................................... 3
- check.variogram ......................................................... 4
- coef.slmfit ................................................................. 4
- corModelExponential ............................................... 5
- exampledataset .......................................................... 6
- geostatSim ................................................................. 6
- get.predinfo ............................................................... 7
- get.predplot ............................................................... 8
- GR2 ........................................................................... 9
- LLtoTM ....................................................................... 9
- loglik.slmfit ............................................................... 10
- m2LL.FPBK.nodet ....................................................... 11
- mginv ....................................................................... 12
- pointSimCSR ............................................................... 12
- pointSimSyst ............................................................... 13
- predict.slmfit ............................................................. 14
- print.summary.slmfit .................................................. 15
- residuals.slmfit .......................................................... 16
- simdata .................................................................... 16
- slmfit ....................................................................... 17
- sptotal ...................................................................... 19
- summary.slmfit .......................................................... 20
- USlakes ..................................................................... 20

**Index** 22

---

AIC.slmfit  
*Extract the AIC from a slmfit object for comparing models.*

### Description

Extract the AIC from a slmfit object for comparing models.

### Usage

```r
## S3 method for class 'slmfit'
AIC(object, ...)
```

### Arguments

- `object`  
a slmfit object
- `...`  
further arguments passed to or from other methods.
Value
The AIC value of the slmfit object. Here, AIC is computed as 2 times the negative log-likelihood plus 2 times the number of model parameters for Maximum Likelihood (ML) and 2 times the negative log-likelihood for REML. For REML, AIC should only be used to compare two models with the same covariates but different spatial covariance structures.

Examples

data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset, xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
AIC(slmobj)

AKmoose
Data Set with Alaska Moose Counts.

Description
A data set that can be used with the sptotal package. In this example, the counts are of moose on 860 sites of equal area.

Usage
AKmoose

Format
A SpatialPolygonsDataFrame object in the format of the (sp) package. The data frame AKmoose@data contains 860 rows and 4 columns:

CENTRLAT  The latitude of the centroid for each site
CENTRLON  The latitude of the centroid for each site
STRAT     A stratification variable
TOTAL     The total moose count on each site

Source
Alaska Department of Fish and Game, Division of Wildlife Conservation has released this data set under the CC0 (creative commons) license. To the extent possible under law, Alaska Department of Fish and Game, Division of Wildlife Conservation waives all copyright and related or neighboring rights to An Alaskan GSPE (Geospatial Population Estimator) Survey of Moose, AKmoose.rda. This work is published from: United States.

Examples

data(AKmoose)
names(AKmoose@data)
summary(AKmoose@data)
check.variogram  
*Plot an Empirical Variogram of Residuals*

**Description**

Plots an empirical variogram of the residuals from the spatial linear model with the fitted parametric model as a curve overtop of the binned points. By default, the empirical variogram only shows distances that are less than or equal to the maximum distance in the data set divided by 2. Therefore, it's possible that the REML-fitted model will not "fit" the points perfectly.

**Usage**

```r
check.variogram(object)
```

**Arguments**

- `object` is an object of class `slmfit`.

**Value**

A plot of the empirical variogram with the fitted model overlayed.

**Examples**

```r
data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset, xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
check.variogram(slmobj)
```

---

**coef.slmfit**  
*Extract Model Coefficients from a slmfit object*

**Description**

Extract Model Coefficients from a `slmfit` object

**Usage**

```r
## S3 method for class 'slmfit'
coef(object, ...)
```

**Arguments**

- `object` a `slmfit` object
- `...` further arguments passed to or from other methods.
corModelExponential

Value

a vector of fitted model coefficients.

Examples

data(exampledataset)  ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
coef(slmobj)

Spatial Correlation Models

Description

Note that, currently, only three of these models are implemented in the sptotal package: corModelExponential(), corModelGaussian(), and corModelSpherical().

Usage

   corModelExponential(distance.matrix, range)
   corModelGaussian(distance.matrix, range)
   corModelSpherical(distance.matrix, range)

Arguments

distance.matrix
   The distance matrix for sampled sites
range
   The range that determines how quickly covariance among sites tapers

Value

Correlation Matrix

Functions

• corModelExponential: Exponential Correlation Structure
• corModelGaussian: Gaussian Correlation Structure
• corModelSpherical: Spherical Correlation Structure
exampledataset  

Data Set with Uncorrelated Poisson Counts.

Description

A toy data set that can be used with the sptotal package. In this example, the true counts are actually uncorrelated, the covariates are generated as uniform random variables, and the sites fall on a regular grid.

Usage

exampledataset

Format

A data frame with 40 rows and 7 variables:

- counts: counts, with NA values for unsampled sites
- pred1: a possible predictor
- pred2: a second possible predictor
- xcoords: coordinates on the x-axis
- ycoords: coordinates on the y-axis
- dummyvar: an extra variable
- areavar: Variable for the area of each plot...

geostatSim  

Simulate geostatistical data on set of given locations

Description

Spatially correlated data are simulated assuming a multivariate normal random error vector. For simplicity, only "Exponential" and "Spherical" simulation options are given here.

Usage

geostatSim(
  loc.data,
  xcol = "x",
  ycol = "y",
  parsil = 1,
  range = 1,
  nugget = 0,
  minorp = 1,
  rotate = 90,
  extrap = NULL,
  CorModel = "Exponential"
)
get.predinfo

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>loc.data</td>
<td>data.frame with x- and y-coordinates of locations for simulated data</td>
</tr>
<tr>
<td>xcol</td>
<td>name of the column in loc.data with x-coordinates, default is &quot;x&quot;</td>
</tr>
<tr>
<td>ycol</td>
<td>name of the column loc.data with y-coordinates, default is &quot;y&quot;</td>
</tr>
<tr>
<td>parsil</td>
<td>partial sill of autocorrelation model, default = 1</td>
</tr>
<tr>
<td>range</td>
<td>range of autocorrelation model, default = 1</td>
</tr>
<tr>
<td>nugget</td>
<td>range of autocorrelation model, default = 0</td>
</tr>
<tr>
<td>minorp</td>
<td>proportion of range in x direction to that of y direction for unrotated anisotropic model, default = 1</td>
</tr>
<tr>
<td>rotate</td>
<td>rotation of anisotropic axes, default = 90</td>
</tr>
<tr>
<td>extrap</td>
<td>extra covariance parameter</td>
</tr>
<tr>
<td>CorModel</td>
<td>autocorrelation model, default = &quot;Exponential&quot;. Other possibilities are &quot;Spherical&quot;.</td>
</tr>
</tbody>
</table>

Value

data.frame of three columns, the original location data appended with a 3rd column of simulated geostatistical data

Author(s)

Jay Ver Hoef

Examples

locations <- expand.grid(1:10, 1:10)
geostatSim(locations, xcol = "Var1", ycol = "Var2", parsil = 4, range = 20, nugget = 1, CorModel = "Exponential")

corr <- c(0.1, 0.2, 0.3, 0.4)
get.predinfo(corr, conf_level = 0.9)

Description

Displays basic summary information in a tabular form.

Usage

gpred <- get.predinfo(x, conf_level = 0.9)

Arguments

x | the output of the predict.slmfit() function, of class sptotalPredOut
conf_level | is the confidence level for a normal-based confidence interval (default = 0.90).
get.predplot

Value

a list of three tables, including

- simptab, which contains the prediction and its standard error,
- confbounds, which contains a confidence interval for the prediction, and
- outptmat, a table of sampling information, including the number of sites sampled, the total number of sites, the total observed response, and the observed average density (equal to the average response if all site areas are equal).

Examples

data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
predobj <- predict(slmobj)
get.predinfo(predobj)

get.predplot

Create a default map from predictions

Description

Creates a default map for the predictions of unobserved sites. Note that all predictions are stored in a data frame in the output of predict.slmfit(). Therefore, if a user would like to create his or her own plot, he or she can easily do so using this data frame.

Usage

get.predplot(x)

Arguments

x

the output of the predict.slmfit() function, of class sptotalPredOut

Value

a plot with x-coordinates on the x-axis and y-coordinates on the y-axis that is coloured by predictions, with points with an X denoting that a site was sampled and filled circles denoting unsampled sites.

Examples

data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
predobj <- predict(slmobj)
get.predplot(predobj)
Computes the Generalized R-squared.

Description

Computes the Generalized R-squared.

Usage

GR2(object)

Arguments

object is an object of class slmfit.

Value

the value for generalized R-squared

Examples

data(exampledataset) # load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
                  xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
GR2(slmobj)

Convert Latitude and Longitude spatial coordinates to transverse Mercator (TM) projection coordinates with a user-defined central meridian.

Description

The resulting units from applying the function are kilometers.

Usage

LLtoTM(cm, lat, lon, xcol = "x", ycol = "y", minx = NULL, miny = NULL)
Arguments

- **cm** is the user defined central median. A common choice is the mean of the longitude values in your data set.
- **lat** is the vector of latitudes.
- **lon** is the vector of longitudes.
- **xcol** is the name of the output TM column of x coordinates.
- **ycol** is the name of the output TM column of y coordinates.
- **minx** is ‘NULL’ by default and sets the minimum x-coordinate value to 0. This is an optional minimum value for the x-coordinate vector.
- **miny** is ‘NULL’ by default and sets the minimum y-coordinate value to 0. This is an optional minimum value for the y-coordinate vector.

Details

This function only should only be used if the coordinates supplied by the user are latitude and longitude. The default TM projection here specifies that both the minimum x and y-coordinate values are 0 scaled to 1 km.

Value

A list with the TM coordinates as the first component of the list. The first component of the list contains x coordinates in the first column and y coordinates in the second column. The remaining elements of the list are the cm, minx, and miny values that were input.

Examples

```r
## Add transverse Mercator x and y coordinates to a data frame with
## latitude/longitude coordinates. Name these \code{xc_TM_} and \code{yc_TM_}.
examedataset$xc_TM_ <- LLtoTM(cm = base::mean(examedataset[,"xcoords"]),
lat = examedataset[,"ycoords"],
lon = examedataset[,"xcoords"]$xy[,1]
examedataset$yc_TM_ <- LLtoTM(cm = base::mean(examedataset[,"xcoords"]),
lat = examedataset[,"ycoords"],
lon = examedataset[,"xcoords"]$xy[,2]
```

loglik.slmfit

Extract Log-Likelihood from a fitted class slmfit object

Description

Extract Log-Likelihood from a fitted class slmfit object

Usage

loglik.slmfit(object, ...)

Arguments

object is a class slmfit object
...
additional arguments

Value

the log-likelihood of the model fit in the slmfit object

Examples

data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
loglik.slmfit(slmobj)

Description

The primary purpose of `m2LL.FPBK.nodet()` is to estimate the spatial covariance parameters using REML. This is a helper function to `slmfit()`.

Usage

`m2LL.FPBK.nodet(theta, zcol, XDesign, xcoord, ycoord, CorModel, estmethod)`

Arguments

`theta` is the parameter vector of (nugget, partialsill, range)
`zcol` is the response vector of densities
`XDesign` is the design matrix containing the covariates used to predict animal or plant abundance (including a column of 1’s for the intercept).
`xcoord` is a vector of the x spatial coordinates (in UTM)
`ycoord` is a vector of the y spatial coordinates (in UTM)
`CorModel` is the geostatistical spatial correlation model to be used. See the `corModels` documentation for possible models to use.
`estmethod` is either "REML" for restricted maximum likelihood or "ML" for maximum likelihood.

Value

A numeric output of minus 2 times the restricted log likelihood to be minimized by `optim` to obtain spatial parameter estimates.
Constructing the generalized inverse of a matrix

**Description**
Computes the generalized inverse of a matrix X. This function is used in the `m2LL.FPBK.nodet` functions in order to estimate the spatial covariance parameters.

**Usage**
```
mginv(X, tol = sqrt(.Machine$double.eps))
```

**Arguments**
- **X**: The matrix to be inverted
- **tol**: The tolerance of the estimation

**Value**
The generalized inverse matrix

---

**pointSimCSR**
simulate completely spatially random point patterns.

**Description**
simulates a completely spatially random point patterns. This function is only used in simulating data sets.

**Usage**
```
pointSimCSR(
  npoints = 100,
  lower_x_lim = 0,
  upper_x_lim = 1,
  lower_y_lim = 0,
  upper_y_lim = 1
)
```

**Arguments**
- **npoints**: number of points to add that are completely spatially random (CSR), default = 100
- **lower_x_lim**: left limit of boundary, default = 0
- **upper_x_lim**: right limit of boundary, default = 1
- **lower_y_lim**: lower limit of boundary, default = 0
- **upper_y_lim**: upper limit of boundary, default = 1
**pointSimSyst**

**Value**

data.frame of two columns, x-coordinate in the first, and y-coordinate in the second.

**Author(s)**

Jay Ver Hoef

---

**pointSimSyst**  
*Creates a systematic grid of points.*

**Description**

Creates a systematic grid of points. This function is only used in simulating data sets.

**Usage**

```r
pointSimSyst(
  nrow = 10,
  ncol = 10,
  lower_x_lim = 0,
  upper_x_lim = 1,
  lower_y_lim = 0,
  upper_y_lim = 1
)
```

**Arguments**

- `nrow`: the number of rows in the systematic grid, default is 10
- `ncol`: the number of cols in the systematic grid, default is 10
- `lower_x_lim`: the lower limit for x-coordinate, default is 0
- `upper_x_lim`: the upper limit for x-coordinate, default is 1
- `lower_y_lim`: the lower limit for y-coordinate, default is 0
- `upper_y_lim`: the upper limit for y-coordinate, default is 1

**Value**

A data.frame with x- and y-coordinates of simulated locations

**Author(s)**

Jay Ver Hoef
predict.slmfit

Perform Finite Population Block Kriging

Description

Uses an object of class slmfit from the slmfit() function to predict the response on the unsampled sites. The column of the data set that has the response should have numeric values for the observed response on the sampled sites and ‘NA’ for any site that was not sampled.

Usage

```r
## S3 method for class 'slmfit'
predict(object, wtscol = NULL, ...)
```

Arguments

- `object`: is an object generated from `slmfit()`
- `wtscol`: is the name of the column that contains the weights for prediction.
- `...`: further arguments passed to or from other methods. The default setting predicts the population total

Value

A list with

- the estimated population total
- the estimated prediction variance
- a data frame containing
  1. x-coordinates
  2. y-coordinates
  3. density predictions
  4. count predictions
  5. site-by-site density prediction variances
  6. site-by-site count prediction variances
  7. indicator variable for whether or not the each site was sampled
  8. estimated mean for each site
  9. area of each site
- vector with estimated covariance parameters
- the formula used to fit the model in `slmfit()`

Examples

```r
data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
predict(slmobj)
```
print.summary.slmfit  

Prints the summary of a fitted spatial linear model.

Description

This function uses the object that is output from `summary.slmfit()`.

Usage

```r
## S3 method for class 'summary.slmfit'
print(
  x,
  digits = max(3L, getOption("digits") - 3L),
  signif.stars = getOption("show.signif.stars"),
  ...
)
```

Arguments

- `x` is an summary object generated from `summary.slmfit()`
- `digits` is the number of digits to be displayed in the model output
- `signif.stars` is an option to show which predictors are significant.
- `...` further arguments passed to or from other methods. `summary.slmfit`.

Value

A list with

- model formula
- summary statistics for the residuals.
- a table of fixed effects estimates and associated standard errors.
- estimated spatial covariance parameter estimates.
- generalized r-squared value.

Examples

```r
data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
print(summary(slmobj))
```
residuals.slmfit Extract Model Residuals from an slmfit object.

Description

Extract Model Residuals from an slmfit object.

Usage

## S3 method for class 'slmfit'
residuals(object, cross.validation = FALSE, ...)

Arguments

object a slmfit object generated from the slmfit() function.
cross.validation

a logical (TRUE or FALSE) that indicates whether the residuals computed should be found using leave one out cross-validation. Set to FALSE by default.

...

further arguments passed to or from other methods.

Value

a vector of residuals, consisting of each observed response/density minus the estimated mean, or, in the case of cross-validation, the observed response/density minus the leave-one-out-cross-validation prediction.

Examples

data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset, xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
residuals(slmobj)
residuals(slmobj, cross.validation = TRUE)

simdata Simulated Spatially Autocorrelated Data.

Description

A simulated data set that can be used with the sptotal package.

Usage

simdata
Format

A data frame object including:

- x  The x-coordinate for each site
- y  The y-coordinate for each site
- X1 Simulated independent variable to be used as a predictor
- X2 Simulated independent variable to be used as a predictor
- X3 Simulated independent variable to be used as a predictor
- X4 Simulated independent variable to be used as a predictor
- X5 Simulated independent variable to be used as a predictor
- X6 Simulated spatially correlated random variable to be used as a predictor
- X7 Simulated spatially correlated random variable to be used as a predictor
- F1 Simulated factor variable to be used as a predictor
- F2 Simulated factor variable to be used as a predictor
- Z  The simulated response variable.
- wts1 Prediction weights if estimating an overall mean
- wts2 Prediction weights for estimating a total over a subset of 25 contiguous plots

Examples

data(simdata)
names(simdata)
summary(simdata)

slmfit

Fits a Spatial Linear Model

Description

Estimates regression coefficients and spatial autocorrelation parameters, given spatial coordinates and a model formula.

Usage

slmfit(
  formula,
  data,
  xcoordcol,
  ycoordcol,
  areacol = NULL,
  CorModel = "Exponential",
  estmethod = "REML",
  covestimates = c(NA, NA, NA)
)

Arguments

- **formula** is an R linear model formula specifying the response variable as well as covariates for predicting the response on the unsampled sites.
- **data** is the data set with the response column, the covariates to be used for the block kriging, and the spatial coordinates for all of the sites.
- **xcoordcol** is the name of the column in the data frame with x coordinates or longitudinal coordinates.
- **ycoordcol** is the name of the column in the data frame with y coordinates or latitudinal coordinates.
- **areacol** is the name of the column with the areas of the sites. By default, we assume that all sites have equal area, in which case a vector of 1’s is used as the areas.
- **CorModel** is the covariance structure. By default, CorModel is Exponential but other options include the Spherical and Gaussian.
- **estmethod** is either the default "REML" for restricted maximum likelihood to estimate the covariance parameters and regression coefficients or "ML" to estimate the covariance parameters and regression coefficients. This argument can also be set to "None", in which case covestimates must be provided.
- **covestimates** is an optional vector of covariance parameter estimates (nugget, partial sill, range). If these are given and estmethod = "None", the the provided vector are treated as the estimators to create the covariance structure.

Value

A list of class `slmfit` with

- the spatial covariance estimates
- the regression coefficient estimates
- the covariance matrix of the fixed effects
- minus two times the log-likelihood of the model
- the names of the predictors
- the sample size
- the name of the covariance model used
- a vector of residuals
- the design matrix
- a vector of the sampled densities
- a list containing

  1. formula, the model formula
  2. data, the data set input as the data argument
  3. xcoordcol, the name of the x-coordinate column
  4. ycoordcol, the name of the y-coordinate column
  5. estmethod, either REML or ML
  6. CorModel, the correlation model used
  7. estimated covariance matrix of all sites
  8. Inverted covariance matrix on the sampled sites
  9. the vector of areas.
Examples

data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
summary(slmobj)

sptotal: A package used for performing Finite Population Block Kriging (FPBK) on polygonal count data.

Description

The package provides an option to perform FPBK on counts assuming perfect detection of counts on the sites that were in the survey sample. The functions in the package use methods in (Ver Hoef, 2008, <doi:10.1007/s10651-007-0035-y>)

Details

critical

sptotal Main Functions:

slmfit fits a spatial linear model to the response on the observed/sampled sites. check.variogram can be used to construct an empirical variogram of the residuals of the spatial linear model.

predict.slmfit uses the spatial linear model fit from slmfit and finite population block kriging to predict the response at unobserved locations. A prediction for the total response as well as a prediction variance are given by default.

get.predinfo and get.predplot take the resulting object from predict.slmfit to construct (1) summary information, including the prediction, prediction variance, and a prediction interval as well as (2) a plot of the site-wise predictions.

Most of the remaining functions in the package are either helper functions or extra optional functions to extract various specific things from an slmfit object, such as residuals, AIC, log-likelihood, etc.

See the Vignette for more details: browseVignettes("sptotal") Reference for Mathematical Details:

summary.slmfit  

Summarizes a fitted spatial linear model.

Description
In conjunction with print.summary.slmfit(), the output looks similar to output from R’s standard lm() function.

Usage
## S3 method for class 'slmfit'
summary(object, ...)

Arguments

object is an object generated from slmfit() of class slmfit.

... further arguments passed to or from other methods.

Value

a list with

• model formula
• a table of fixed effects estimates and associated standard errors
• estimated spatial covariance parameter estimates
• residuals
• generalized r-squared.

Examples
data(exampledataset) ## load a toy data set
slmobj <- slmfit(formula = counts ~ pred1 + pred2, data = exampledataset,
                  xcoordcol = 'xcoords', ycoordcol = 'ycoords', areacol = 'areavar')
summary(slmobj)

USlakes  

Dissolved Organic Carbon in U.S. Lakes

Description
These data contain dissolved organic carbon (DOC) in National Lakes Data from the U.S. Environmental Protection Agency

Usage

USlakes
USlakes

Format

A data frame with 1206 rows and 9 variables:

- **XCOORD**  x-coordinate from US Contiguous Albers Equal Area Conic projection
- **YCOORD**  y-coordinate from US Contiguous Albers Equal Area Conic projection
- **DOC_RESULT**  Analyte value, in mg/L, for Dissolved Organic Carbon
- **ELEVATION**  Elevation at lake coordinates (LAT_DD_N83, LON_DD_N83) from NHD Digital Elevation Map layer
- **FCIBIG_LIT**  Fish cover: index of fish cover due to large structures in the littoral zone
- **RVFCGNDDBARE_RIP**  riparian zone and vegetation: fraction of ground lacking cover in the riparian zone
- **RVFCGNDWOODY_RIP**  riparian zone and vegetation: fraction of ground cover by woody vegetation in the riparian zone
- **RVFPUNDWOODY_RIP**  riparian zone and vegetation: fraction of understory with nonwoody cover present in the riparian zone
- **UID**  A unique lake identifier in the EPA lake survey datebases

Source

National Aquatic Resource Surveys webpage. We combined site data, DOC data, and habitat metrics to create a data set of 1206 lakes in the conterminous United States.

Examples

data(USlakes)
names(USlakes)
summary(USlakes)
Index

* datasets
  AKmoose, 3
  exampledataset, 6
  simdata, 16
  USlakes, 20

AIC.slmfit, 2
  AKmoose, 3

check.variogram, 4, 19
coef.slmfit, 4
corModelExponential, 5
corModelGaussian (corModelExponential), 5
corModelSpherical
  (corModelExponential), 5

exampledataset, 6

geostatSim, 6
get.predinfo, 7, 19
get.plot, 8, 19
GR2, 9

LLtoTM, 9
loglik.slmfit, 10

m2LL.FPBK.nodet, 11
mginv, 12

pointSimCSR, 12
pointSimSyst, 13
predict.slmfit, 7, 8, 14, 19
print.summary.slmfit, 15

residuals.slmfit, 16

simdata, 16
slmfit, 2, 4, 14, 16, 17, 19, 20
sptotal, 19
summary.slmfit, 15, 20

USlakes, 20