Package 'nnDiag'

January 2, 2012

Title k-Nearest Neighbor Diagnostic Tools

Version 0.0-5

Date 2010-01-11

Author Brian Walters <walte137@msu.edu>, Andrew O. Finley <finleya@msu.edu>, Zhen Zhang <zhangz19@stt.msu.edu>

Maintainer Brian Walters <walte137@msu.edu>

Depends sp, rgdal, spBayes, gplots, vcd, MASS, MBA, fields, yaImpute,geoR

Description A suite of graphical diagnostic tools to evaluate kNN classifications.

URL http://blue.for.msu.edu/NAFIS/software.html

LazyLoad yes

ZipData no

Repository CRAN

Date/Publication 2010-01-11 18:10:16

R topics documented:

`nnDiag-package` *A suite of graphical diagnostic tools to evaluate kNN classifications.*

Description

Contains diagnostic functions that can be applied to k-nearest neighbor classifications, both continuous and categorical data.

Details

Package:	nnDiag
Type:	Package
Version:	0.0-5
Date:	2010-01-11
License:	GPL (>= 2)
LazyLoad:	yes

Author(s)

Brian Walters <walte137@msu.edu>, Andrew O. Finley <finleya@msu.edu>, Zhen Zhang <zhangz19@stt.msu.edu>

Maintainer: Brian Walters

References

Hudson, W.D. and Ramm, C.W. (1987) Correct Formulation of the Kappa Coefficient of Agreement, *Photogrammetric Engineering and Remote Sensing*. **53**, 421–422.

McRoberts, R.E., Tomppo, E.O., Finley, A.O., Heikkinen, J. (2007) Estimating areal means and variances of forest attributes using the k-Nearest Neighbors technique and satellite imagery, *Remote Sensing of Environment*. **111**, 466–480.

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

Rosenfield, G.H. and Fitzpatrick-Lins, K. (1986) A Coefficient of Agreement as a Measure of Thematic Classification Accuracy, *Photogrammetric Engineering and Remote Sensing*. **52**, 223–227.

Story, M. and Congalton, R.G. (1986) Accuracy Assessment: A User's Perspective, *Photogrammetric Engineering and Remote Sensing*. **52**, 397–399.

`aoiMaker` *Create Random Areas of Interest (AOI) From a Larger Image*

Description

The purpose of this function is to create Areas of Interest (AOI) from a larger image that include a minimum number of reference set elements. The object created can then be used in kNN diagnostic tests.

Usage

```
aoiMaker(ref.coords, num.aoi, image, mask = NULL, aoi.size = 1000, min.points = 25, seed = NULL, verbose
```

Arguments

`ref.coords`	matrix of reference point coordinates
`num.aoi`	a single integer, number of AOIs desired
`image`	the image to make AOIs from, as an object of class `GDALReadOnlyDataset`
`mask`	optional mask image, as an object of class `GDALReadOnlyDataset`
`aoi.size`	a single integer, size of the AOI in pixels
`min.points`	a single integer, minimum number of reference points in each AOI desired
`seed`	a single value, interpreted as an integer
`verbose`	logical indicating whether to display a progress bar

Details

The `ref.coords` must be a two column matrix with an X and Y location.

Both the `image` and `mask` have to be in the class `"GDALReadOnlyDataset"`. This can be achieved by using the command `GDAL.open` from the package **rgdal**.

Each AOI created by this function is a square with the length of each side being the `aoi.size` in number of pixels. For example, if the desired AOI size was 300 meters on a side and the `image` had 30 meter pixels the `aoi.size` should be 100.

The minimum number of reference points within each AOI cannot be less than 10. An error will display if `min.points` is set less than that.

The seed option is available so identical AOIs can be made from different images. See `set.seed` for more information.

Value

Returns an object of `class` `"nnDaoi"`, which is a list containing the following components:

`AOI.spatial`	a list of the AOIs. Each element in the list is an object of class `SpatialGridDataFrame`
`AOI.data`	a list of the pixel values contained in each AOI. Each element in the list is a matrix.

`refPoint.index` a list with the index of reference points contained in each AOI. Each element in the list is a vector.

`ref.coordinates`
a matrix of the reference point coordinates

Note

Plotting an "nnDaoi" object will display the set of reference points and outlines of each AOI.

Author(s)

Brian Walters <walte137@msu.edu>

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

See Also

Functions that use "nnDaoi" objects: arealBias, extrap.

Examples

```
data(LuceVolume_subset)
LuceVolImg <- GDAL.open(system.file("data/LuceSubset_Volume.tif", package = "nnDiag"))
LuceTasscap <- GDAL.open(system.file("data/LuceSubset_Tcap.tif", package = "nnDiag"))
LuceMask <- GDAL.open(system.file("data/LuceSubset_Mask.tif", package = "nnDiag"))

coords <- as.matrix(LuceVolume_subset[,10:11])

vol.aoi <- aoiMaker(coords, 3, LuceVolImg, aoi.size = 20, min.points = 4, seed = 89)
plot(vol.aoi)

tcap.aoi <- aoiMaker(coords, 3, LuceTasscap, LuceMask, aoi.size = 20, min.points = 4, seed = 89)
plot(tcap.aoi)
```

---

arealBias *Bias Assessment Using an Areal Approach*

Description

Using the AOIs created from aoiMaker, this function makes an object that can be plotted to assess the areal bias of kNN classification. See McRoberts (2009) for a description of the plot.

Usage

```
arealBias(object, reference.set)
```

Arguments

`object`	object of class "`nnDaoi`"
`reference.set`	vector of reference set observations

Value

Returns an object of `class` "`nnDarbias`", which is a list containing the following:

`knn.estimates` vector of the mean values of each AOI

`probability.estimates`
vector of the mean value of the reference points contained within each AOI

`lower.confidence`
vector of the lower confidence interval values

`upper.confidence`
vector of the upper confidence interval values

Author(s)

Brian Walters <walte137@msu.edu>

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

See Also

`aoiMaker`, `bias`

Examples

```
data(LuceVolume_subset)
LuceVolImg <- GDAL.open(system.file("data/LuceSubset_Volume.tif", package = "nnDiag"))
LuceMask <- GDAL.open(system.file("data/LuceSubset_Mask.tif", package = "nnDiag"))

## First create \code{"nnDaoi"} object
coords <- as.matrix(LuceVolume_subset[,10:11])
vol.aoi <- aoiMaker(coords, 3, LuceVolImg, LuceMask, aoi.size = 20,
min.points = 4)

ref.vol <- LuceVolume_subset$ref.volume

x <- arealBias(vol.aoi, ref.vol)
plot(x)
```

`bias` *Bias Assessment*

Description

Assesses bias by the relationship between observations and predictions, whether in ordered groups or by individual data elements.

Usage

```
bias(object, mode = "groups")
```

Arguments

`object`	object of class "`nnDgrps`"
`mode`	can either be set to "`groups`" to assess means of groups of observations against means of predictions, or "`points`" to assess each observation against its corresponding prediction.

Value

An object of `class` "`nnDbias`".

When mode = "`groups`", object is a list that contains the following components:

`mean.predictions`	a vector of means of the groups of predictions.
`mean.observations`	a vector of means of the groups of observations.

When mode = "`points`" the list contains:

`predictions`	vector of predictions.
`observations`	vector of observations.

Author(s)

Brian Walters <`walte137@msu.edu`>

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

See Also

`grouper`, `arealBias`

Examples

```
data(LuceVolume)
data(LuceVolume_indx)

##First use grouper to make the \code{"nnDgrps"} object
grps <- grouper(LuceVolume$ref.volume, LuceVolume$pred.vol_k18, LuceVolume_indx)

##Using mode \code{"groups"}
bg <- bias(grps, mode = "groups")
plot(bg)

##Using mode \code{"points"}
bp <- bias(grps, mode = "points")
plot(bp)
```

`categorical` *Diagnostics for categorical variable classifications*

Description

Given a reference set of categorical variable observations and its corresponding predictions this function produces the classification confusion matrix, accuracy and kappa coefficient. Accuracy and kappa are given as overall, user's and producer's.

Usage

```
categorical(reference.set, predicted.set, class.names = NULL)
```

Arguments

`reference.set`	vector of observed values.
`predicted.set`	vector of predicted values.
`class.names`	optional $n \times 2$ matrix of class names. The first column being the class numbers with the second column being their corresponding class names.

Details

Using the optional argument `class.names` will insert the name of each class into the output, otherwise the class numbers extracted from the reference set are used to identify classes.

Value

Function returns an object of `class` `"nnDcat"`, which is a list containing the following components:

`Confusion.Matrix`
: matrix comparing the reference set to the predicted set. The major diagonal is agreement between the two sets.

`Overall.Accuracy`
: overall accuracy of the classification.

`Users.Accuracy`
: user's accuracy for each class.

`Producers.Accuracy`
: producer's accuracy for each class.

`Overall.Kappa`
: kappa coefficient of the overall classification.

`Users.Cond.Kappa`
: user's conditional kappa coefficient for each class.

`Producers.Cond.Kappa`
: producer's conditional kappa coefficient for each class.

Author(s)

Brian Walters <walte137@msu.edu>

References

Hudson, W.D. and Ramm, C.W. (1987) Correct Formulation of the Kappa Coefficient of Agreement, *Photogrammetric Engineering and Remote Sensing*. **53**, 421–422.

Rosenfield, G.H. and Fitzpatrick-Lins, K. (1986) A Coefficient of Agreement as a Measure of Thematic Classification Accuracy, *Photogrammetric Engineering and Remote Sensing*. **52**, 223–227.

Story, M. and Congalton, R.G. (1986) Accuracy Assessment: A User's Perspective, *Photogrammetric Engineering and Remote Sensing*. **52**, 397–399.

Examples

```
data(LuceForgrp)

x <- categorical(LuceForgrp$ref.forgrp, LuceForgrp$pred.forgrp)
x

## Add class names
clsnms <- cbind(c(100,120,380,400,700,800,900,999), c("WRJ Pine",
"Spruce/Fir", "Exotic SW", "Oak/Pine", "Elm/Ash/Cottonwood",
"Maple/Beech/Birch", "Aspen/Birch", "Nonstocked"))

x <- categorical(LuceForgrp$ref.forgrp, LuceForgrp$pred.forgrp, class.names = clsnms)
x
```

`extrap` *Extrapolations Test*

Description

Comparison of the range of observations in the feature space of both the reference set and target set. Any target set pixel whose value is beyond the range of the reference set's feature space requires an extrapolation during kNN classification. This function prepares an object to plot the comparison and find if and where an extrapolation is necessary.

Usage

```
extrap(image, refSet.spectral, mask = NULL, verbose = TRUE)
```

Arguments

image	the target set image, either an object of class GDALReadOnlyDataset or an object of class "nnDaoi"
refSet.spectral	matrix of the feature space values of the reference set
mask	optional mask, as an object of class GDALReadOnlyDataset
verbose	logical indicating whether to display a progress bar

Details

Both the image and mask have to be in the class "GDALReadOnlyDataset". This can be achieved by using the command GDAL.open from the package **rgdal**.

Value

Returns an object of class "nnDext", which is a list containing the following components:

image.range	a matrix of the range of spectral values found on the image or in the AOIs for each spectral layer
refSet.range	a matrix of the range of spectral values for the reference set of data elements

Author(s)

Brian Walters <walte137@msu.edu>

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

See Also

aoiMaker

Examples

```
data(LuceVolume_subset)
LuceTasscap <- GDAL.open(system.file("data/LuceSubset_Tcap.tif", package = "nnDiag"))
LuceMask <- GDAL.open(system.file("data/LuceSubset_Mask.tif", package = "nnDiag"))

spect <- as.matrix(LuceVolume_subset[12:14])
## Not run:
x <- extrap(LuceTasscap, spect, LuceMask)
plot(x)

## Using an \code{"nnDaoi"} object
```

```
coords <- as.matrix(LuceVolume_subset[,10:11])
tcap.aoi <- aoiMaker(coords, 3, LuceTasscap, LuceMask, aoi.size = 20, min.points = 4)

x <- extrap(tcap.aoi, spect)
plot(x)
## End(Not run)
```

---

grouper *Produce Ordered Groups of Data Elements*

Description

The purpose of this function is to produce an object that can be used in many of the diagnostic tools from this package. It orders the reference set elements with respect to predictions and breaks them into groups with an arbitrary number of elements. Also included in the output is the number of times each element was used as a neighbor in the kNN classification and the residuals.

Usage

```
grouper(reference.set, predicted.set, nnIndex, group.size = 25, best = TRUE)
```

Arguments

`reference.set`	vector of observed values
`predicted.set`	vector of predicted values
`nnIndex`	data.frame of nearest neighbors index
`group.size`	a single integer, number of elements to be in each group
`best`	logical indicating whether the function will use the exact `group.size` input or if it will find the "best" size nearest to the `group.size` input.

Details

The `nnIndex` matrix will have a column for each corresponding k used in the kNN classification. Each element in the matrix should be in reference to the `reference.set` vector position of the observed value.

If the remainder number of data elements do not fill a full group, the rest of that group will have `NA`s as place fillers. The `best` argument is to reduce the number of `NA`s in the last group. When `best = TRUE` the function will find the "best" group size, which is near the `group.size` input, that best fits the number of elements in the reference set. If `best` is set to `FALSE` it will use the `group.size` input exactly.

Value

Returns an object of class "nnDgrps", which is a list containing the following components:

- `ordered.data` data frame of the inputed data ordered by predictions
- `reference.groups` matrix of the groups of reference set elements where each column is a group.
- `predicted.groups` matrix of the groups of predicted set elements where each column is a group.
- `residual.groups` matrix of the groups of residuals where each column is a group.
- `group.size` the number of data elements in each group.

Author(s)

Brian Walters <walte137@msu.edu>

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

See Also

Functions that use "nnDgrps" objects: scedast, outInflu, bias

Examples

```
data(LuceVolume)
data(LuceVolume_indx)

##Using the defaults
x <- grouper(LuceVolume$ref.volume, LuceVolume$pred.vol_k18, LuceVolume_indx)
x

##Not using the defaults
x <- grouper(LuceVolume$ref.volume, LuceVolume$pred.vol_k18, LuceVolume_indx,
group.size = 37, best = FALSE)
x
```

---

knnvarfun *Variance Estimators for kNN*

Description

The purpose of this function is to calculate variance estimates for kNN predictions. The function performs a kNN prediction on a data set split into reference and target sets and outputs variance estimates that correspond to equations (9a) and (11a) in McRoberts, et al. (2007).

Usage

```
knnvarfun(k = 2, ref, y.ref, target, R, ...)
```

Arguments

`k`	the number of nearest neighbors, default is 2
`ref`	matrix of reference set x variables
`y.ref`	vector of reference set observations
`target`	matrix of target set x variables
`R`	spatial correlation matrix of all points
`...`	currently not used

Details

Estimates are undefined for k = 1.

Value

Returns a list containing the following elements:

`yhat`	vector of estimated values at target points
`var1`	variance of mu hat, pertaining to equation (9a) in McRoberts, et al. (2007)
`var2`	variance of y hat, pertaining to equation (11a)
`varterm2`	the second term in equation (11a): $-2k\Sigma\rho$

Author(s)

Zhen Zhang <zhangz19@stt.msu.edu>, Brian Walters <walte137@msu.edu>

References

McRoberts, R.E., Tomppo, E.O., Finley, A.O., Heikkinen, J. (2007) Estimating areal means and variances of forest attributes using the k-Nearest Neighbors technique and satellite imagery, *Remote Sensing of Environment*. **111**, 466–480.

See Also

Functions that are used in the examples below: `as.geodata`, `variog`, `variofit`, `iDist`, `mba.surf`, `image.plot`.

Examples

```
##########################
## Using synthetic data ##
##########################

coords.tar <- as.matrix(expand.grid(seq(0,2000,100), seq(0,2000,100)))
coords.ref <- as.matrix(expand.grid(seq(10,990,length.out=2), seq(10,990,length.out=2)))
coords <- rbind(coords.ref, coords.tar)

n <- nrow(coords)
sigma.sq <- 1
phi <- 3/500
R <- exp(-phi*as.matrix(dist(coords)))

y <- mvrnorm(1, rep(0,n), sigma.sq*R)
refs <- 1:nrow(coords.ref)
y.ref <- y[refs]
y.tar <- y[-refs]

x <- knnvarfun(k=2, coords.ref, y.ref, coords.tar, R)

## Create and graph surfaces from \code{x}
par(mfrow=c(2,3))
surf <- mba.surf(cbind(coords, y), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="observed", ylim=c(0,2000), xlim=c(0,2000))

surf <- mba.surf(cbind(coords.tar, x$yhat), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="y.hat", ylim=c(0,2000), xlim=c(0,2000))
points(coords.ref, pch=19, cex=0.5)

surf <- mba.surf(cbind(coords.tar, x$var1), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="var_mu.hat", ylim=c(0,2000), xlim=c(0,2000))
points(coords.ref, pch=19, cex=2)
points(coords.tar, cex=2)

surf <- mba.surf(cbind(coords.tar, x$var2), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="var_y.hat", ylim=c(0,2000), xlim=c(0,2000))
points(coords.ref, pch=19, cex=2)
points(coords.tar, cex=2)

surf <- mba.surf(cbind(coords.tar, x$varterm2), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="var_y.hat_term2", ylim=c(0,2000), xlim=c(0,2000))
points(coords.ref, pch=19, cex=2)
points(coords.tar, cex=2)


########################
## Using package data ##
########################
data(LuceVolume)

## Obtain the \code{R} correlation matrix from a variogram
```

```
xy.coords <- LuceVolume[,10:11]
max.dist <- max(iDist(xy.coords))

gd <- as.geodata(cbind(xy.coords, LuceVolume[,"ref.volume"]))

vario.1 <- variog(gd, uvec=(seq(0, 0.5*max.dist, length=25)))

vario.fit.1 <-variofit(vario.1, ini.cov.pars=c(750000, 1000),
                       cov.model="exponential",
                       minimisation.function="nls",
                       weights="equal")

phi <-  1/vario.fit.1$cov.pars[2]

R.matrix <- exp(-phi*as.matrix(dist(xy.coords)))

## Split the data into reference and target sets
ref <- sample(1:nrow(LuceVolume), nrow(LuceVolume)*0.5)

ref.coord <- as.matrix(LuceVolume[ref, 10:11])
tar.coord <- as.matrix(LuceVolume[-ref, 10:11])
ref.y <- LuceVolume[ref, "ref.volume"]
tar.y <- LuceVolume[-ref, "ref.volume"]
ref.spect <- as.matrix(LuceVolume[ref, 12:14])
tar.spect <- as.matrix(LuceVolume[-ref, 12:14])
coord <- rbind(ref.coord,tar.coord)
y <- c(ref.y, tar.y)


kvf <- knnvarfun(k=5, ref.spect, ref.y, tar.spect, R.matrix)

## Create and graph surfaces from \code{kvf}
par(mfrow=c(2,3))
surf <- mba.surf(cbind(coord, y), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="observed")

surf <- mba.surf(cbind(tar.coord, kvf$yhat), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="y.hat")
points(ref.coord, pch=20)

surf <- mba.surf(cbind(tar.coord, kvf$var1), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="var_mu.hat")
points(ref.coord, pch=20)
points(tar.coord, cex=2)

surf <- mba.surf(cbind(tar.coord, kvf$var2), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="var_y.hat")
points(ref.coord, pch=20)
points(tar.coord, cex=2)

surf <- mba.surf(cbind(tar.coord, kvf$varterm2), no.X=100, no.Y=100, extend=TRUE)$xyz.est
image.plot(surf, main="var_y.hat_term2")
points(ref.coord, pch=20)
```

```
points(tar.coord, cex=2)
```

LuceForgrp *Forest Inventory Forest Group Type Data on Subplots in Luce County, Michigan*

Description

This data was extracted from the freely available Forest Inventory and Analysis National Program of the USDA Forest Service Database. The coordinates of each subplot are not exact, they have been fuzzed and swapped to meet privacy requirements established by Congress. See pages 10 - 12 in The Forest Inventory and Analysis Database: Database Description and Users Manual Version 4.0 for Phase 2, Revision for more information on the fuzzing and swapping of plot coordinates.

Usage

```
data(LuceForgrp)
```

Format

A data frame containing 734 rows the following 4 columns:

`plt_cn` the plot identifier

`subp` the subplot identifier

`ref.forgrp` forest group on the subplot

`pred.forgrp` predicted forest group on the subplot for k = 12

Source

USDA Forest Service Forest Inventory and Analysis National Program http://fiatools.fs.fed.us/fiadb-downloads/datamart.html

References

Bechtold, W.A. and Patterson, P.L. (2005) *The Enhanced Forest Inventory and Analysis Program - National Sampling Design and Estimation Procedures*.

Forest Inventory and Analysis Program (2009) *The Forest Inventory and Analysis Database: Database Description and Users Manual Version 4.0 for Phase 2, Revision 1*. http://fia.fs.fed.us/library/database-documentation/draftFIADB_userguide_v4-0_p2_05_27_09_pdf.pdf.

LuceVolume *Forest Inventory Volume Data for Subplots in Luce County, Michigan*

Description

This data was extracted from the freely available Forest Inventory and Analysis National Program of the USDA Forest Service Database. The coordinates of each subplot are not exact, they have been fuzzed and swapped to meet privacy requirements established by Congress. See pages 10 - 12 in The Forest Inventory and Analysis Database: Database Description and Users Manual Version 4.0 for Phase 2, Revision for more information on the fuzzing and swapping of plot coordinates.

Usage

```
data(LuceVolume)
```

Format

A data frame containing 671 rows and the following 14 columns:

- `plt_cn` the plot identifier
- `subp` the subplot identifier
- `ref.volume` volume on the subplot in cu.ft./acre
- `pred.vol_k1` predicted volume on subplot for k = 1
- `pred.vol_k5` predicted volume on subplot for k = 5
- `pred.vol_k10` predicted volume on subplot for k = 10
- `pred.vol_k15` predicted volume on subplot for k = 15
- `pred.vol_k18` predicted volume on subplot for k = 18
- `pred.vol_k22` predicted volume on subplot for k = 22
- `xAlb` x location of the subplot projected in Albers Equal Area
- `yAlb` y location of the subplot projected in Albers Equal Area
- `tcap_B` spectral value of Tasseled Cap Transformation brightness
- `tcap_G` spectral value of Tasseled Cap Transformation greenness
- `tcap_W` spectral value of Tasseled Cap Transformation wetness

Source

USDA Forest Service Forest Inventory and Analysis National Program http://fiatools.fs.fed.us/fiadb-downloads/datamart.html

References

Bechtold, W.A. and Patterson, P.L. (2005) *The Enhanced Forest Inventory and Analysis Program - National Sampling Design and Estimation Procedures*.

Forest Inventory and Analysis Program (2009) *The Forest Inventory and Analysis Database: Database Description and Users Manual Version 4.0 for Phase 2, Revision 1*. http://fia.fs.fed.us/library/database-documentation/draftFIADB_userguide_v4-0_p2_05_27_09_pdf.pdf.

LuceVolume_indx — *Index of Neighbors for kNN Classification of Volume, k = 18*

Description

An index of each time a data element was used as a nearest neighbor in a kNN classification where k = 18.

Usage

```
data(LuceVolume_indx)
```

Format

A data frame with 671 rows and 20 columns.

Details

Column 1 represents the first neighbor chosen in the classification, column 2 the second chosen up to column 18 being the 18th neighbor chosen.

LuceVolume_subset — *Subset of Forest Inventory Volume Data for Subplots in Luce County, Michigan*

Description

This data was extracted from the freely available Forest Inventory and Analysis National Program of the USDA Forest Service Database. It is a subset of the larger `LuceVolume` dataset. The purpose of this subset of data is to make the examples that utilize images run faster. The coordinates of each subplot are not exact, they have been fuzzed and swapped to meet privacy requirements established by Congress. See pages 10 - 12 in The Forest Inventory and Analysis Database: Database Description and Users Manual Version 4.0 for Phase 2, Revision for more information on the fuzzing and swapping of plot coordinates.

Usage

```
data(LuceVolume_subset)
```

Format

A data frame containing 12 rows and the following 14 columns:

`plt_cn` the plot identifier

`subp` the subplot identifier

`ref.volume` volume on the subplot in cu.ft./acre

`pred.vol_k1` predicted volume on subplot for k = 1

`pred.vol_k5` predicted volume on subplot for k = 5

`pred.vol_k10` predicted volume on subplot for k = 10

`pred.vol_k15` predicted volume on subplot for k = 15

`pred.vol_k18` predicted volume on subplot for k = 18

`pred.vol_k22` predicted volume on subplot for k = 22

`xAlb` x location of the subplot projected in Albers Equal Area

`yAlb` y location of the subplot projected in Albers Equal Area

`tcap_B` spectral value of Tasseled Cap Transformation brightness

`tcap_G` spectral value of Tasseled Cap Transformation greenness

`tcap_W` spectral value of Tasseled Cap Transformation wetness

Source

USDA Forest Service Forest Inventory and Analysis National Program http://fiatools.fs.fed.us/fiadb-downloads/datamart.html

References

Bechtold, W.A. and Patterson, P.L. (2005) *The Enhanced Forest Inventory and Analysis Program - National Sampling Design and Estimation Procedures*.

Forest Inventory and Analysis Program (2009) *The Forest Inventory and Analysis Database: Database Description and Users Manual Version 4.0 for Phase 2, Revision 1*. http://fia.fs.fed.us/library/database-documentation/draftFIADB_userguide_v4-0_p2_05_27_09_pdf.pdf.

---

outInflu — *Find Potential Outliers and Influential Observations*

Description

Assesses the relationship between standardized residuals and the sum of times used as a neighbor. Outliers/influential observations will have a high standardized residual absolute value and/or have been used as a neighbor numerous times.

Usage

```
outInflu(object)
```

Arguments

`object`	object of class "`nnDgrps`"

Value

An object of `class` "`nnDoi`", which is a list containing the following components:

`neighbor.count`	number of times used as a neighbor in kNN classification
`standardized.residuals`	standardized residuals (the ratios of residuals and their standard deviations)

Author(s)

Brian Walters `<walte137@msu.edu>`

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

See Also

`grouper`

Examples

```
data(LuceVolume)
data(LuceVolume_indx)

##First use grouper to make the \code{"nnDgrps"} object
grps <- grouper(LuceVolume$ref.volume, LuceVolume$pred.vol_k18, LuceVolume_indx)

x <- outInflu(object = grps)
## Not run: plot(x)
```

`rmse` *Root Mean Square Error*

Description

Given a reference set of continuous variable observations and a set of corresponding predictions, this function will calculate the root mean square error of the classification.

Usage

```
rmse(reference.set, predicted.set)
```

Arguments

- `reference.set` vector of observed values
- `predicted.set` vector or matrix of predicted values

Details

The `predicted.set` may be a vector if there is only one classification to find RMSE for, or a matrix if there is multiple classifications. For example, if kNN classifications were run using the same reference set with multiple values of k.

Value

A vector of root mean square error value(s).

Author(s)

Brian Walters `<walte137@msu.edu>`

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

Examples

```
data(LuceVolume)

##Using one predicted set
x <- rmse(LuceVolume$ref.volume, LuceVolume$pred.vol_k18)
x

##Using multiple predicted sets
x <- rmse(LuceVolume$ref.volume, as.matrix(LuceVolume[,4:9]))
x
```

`scedast` *Test of Scedasticity*

Description

Assesses the relationship between standard deviations of residuals with respect to response variable predictions for a test of scedasticity.

Usage

```
scedast(object)
```

Arguments

`object` object of class "`nnDgrps`"

Value

An object of `class` "`nnDsced`", which is a list containing the following components:

`mean.prediction`
: a vector of means of the groups of predictions

`stdev.residuals`
: a vector of the standard deviations of the residual groups

Author(s)

Brian Walters <walte137@msu.edu>

References

McRoberts, R.E. (2009) Diagnostic tools for nearest neighbors techniques when used with satellite imagery, *Remote Sensing of Environment*. **113**, 489–499.

Examples

```
data(LuceVolume)
data(LuceVolume_indx)

##First use grouper to make the \code{"nnDgrps"} object
grps <- grouper(LuceVolume$ref.volume, LuceVolume$pred.vol_k18, LuceVolume_indx)

x <- scedast(object = grps)
## Not run:
plot(x)
plot(x, ylab = "Standard Deviation of Volume Residuals (ft^3/acre)",
xlab = "Mean Volume Predictions(ft^3/acre)", ylim = c(500,2500), xlim =
c(500,2500), pch = 20)

## End(Not run)
```

Index