R documentation

of '/Users/Niladri/Documents/Research/Extending' etc.

November 4, 2013

R topics documented:

`bin_dist` *Binned Distance*

Description

euclidean distance is calculated by binning the data and counting the number of points in each bin

Usage

```
bin_dist(X, PX, X.bin = 5, Y.bin = 5)
```

Arguments

X	a data.frame with two variables, the first two columns are used
PX	another data.frame with two variables, the first two columns are used
X.bin	number of bins on the x-direction, by default nbin.X = 5
Y.bin	number of bins on the y-direction, by default nbin.Y = 5

Value

distance between X and PX

---

box_dist *Distance based on side by side Boxplots for two levels*

Description

distance is calculated by looking at the difference between first quartile, median and third quartile

Usage

```
box_dist(X, PX)
```

Arguments

X	a data.frame with one factor variable and one continuous variable
PX	a data.frame with one factor variable and one continuous variable

Value

distance between X and PX

---

calc_diff *Uses binned distance to calculate the mean distance between the true plot and the null plots in a lineup. also calculates the mean distance of the null plots among themselves and finds the difference between the mean distance of the true plot and the maximum mean distance of the null plots*

Description

Uses binned distance to calculate the mean distance between the true plot and the null plots in a lineup. also calculates the mean distance of the null plots among themselves and finds the difference between the mean distance of the true plot and the maximum mean distance of the null plots

Usage

```
calc_diff(lineup.dat, X.bin, Y.bin, pos, m = 20)
```

Arguments

lineup.dat	lineup data to get the lineup
X.bin	number of bins on the x-direction
Y.bin	number of bins on the y-direction
pos	position of the true plot in the lineup
m	number of plots in the lineup, by default m = 20

Value

difference between the mean distance of the true plot and the maximum mean distance of the null plots

---

decrypt *Use decrypt to reveal the position of the real data.*

Description

The real data position is encrypted by the lineup function, and writes this out as a text string. Decrypt, decrypts this text string to reveal which where the real data is.

Usage

```
decrypt(...)
```

Arguments

...	character vector to decrypt

Examples

```
decrypt("0uXR2p rut L2O2")
```

---

distmet *Calculates the distance measures*

Description

Calculates the distance measures

Usage

```
distmet(lineup.dat, met, method, pos, m = 20,
  dist.arg = NULL, plot = TRUE)
```

Arguments

lineup.dat	lineup data
met	distance metric needed to calculate the distance
method	method for generating null data sets
pos	position of the observed data in the lineup
m	the number of plots in the lineup; m = 20 by default
dist.arg	a vector of inputs for the distance metric met; NULL by default
plot	LOGICAL; if TRUE, returns density plot showing the distn of the measures; TRUE by default

Author(s)

Niladri Roy Chowdhury

---

lal *Los Angeles Lakers play-by-play data.*

Description

Play by play data from all games played by the Los Angeles lakers in the 2008/2009 season.

---

lineup *The line-up protocol.*

Description

In this protocol the plot of the real data is embedded amongst a field of plots of data generated to be consistent with some null hypothesis. If the observe can pick the real data as different from the others, this lends weight to the statistical significance of the structure in the plot. The protocol is described in Buja, Cook, Hofmann, Lawrence, Lee, Swayne, Wickham (2009) Statistical inference for exploratory data analysis and model diagnostics, Phil. Trans. R. Soc. A, 367, 4361-4383.

Usage

```
lineup(method, true = NULL, n = 20, pos = sample(n, 1),
  samples = NULL)
```

Arguments

method	method for generating null data sets
true	true data set. If NULL, find_plot_data will attempt to extract it from the current ggplot2 plot.
n	total number of samples to generate (including true data)
pos	position of true data. Leave missing to pick position at random. Encryped position will be printed on the command line, decrypt to understand.
samples	samples generated under the null hypothesis. Only specify this if you don't want lineup to generate the data for you.

Details

Generate n - 1 null datasets and randomly position the true data. If you pick the real data as being noticeably different, then you have formally established that it is different to with p-value 1/n.

Examples

```
if (require("ggplot2")) {
qplot(mpg, wt, data = mtcars) %+%
  lineup(null_permute("mpg"), mtcars) +
  facet_wrap(~ .sample)
qplot(mpg, .sample, data = lineup(null_permute("cyl"), mtcars),
  colour = factor(cyl))
}
```

---

null_dist *Generate null data with a specific distribution.*

Description

Null hypothesis: variable has specified distribution

Usage

```
null_dist(var, dist, params = NULL)
```

Arguments

var	variable name
dist	distribution name. One of: beta, cauchy, chi-squared, exponential, f, gamma, geometric, log-normal, lognormal, logistic, negative binomial, normal, poisson, t, weibull
params	list of parameters of distribution. If NULL, will use fitdistr to estimate them.

Value

a function that given data generates a null data set. For use with lineup or rorschach

---

null_lm *Generate null data with null residuals from a model.*

Description

Null hypothesis: variable is linear combination of predictors

Usage

```
null_lm(f, method = "rotate", ...)
```

Arguments

f	model specification formula, as defined by lm
method	method for generating null residuals. Built in methods "rotate", "pboot" and "boot" are defined by resid_rotate, resid_pboot and resid_boot respectively
...	other arguments passedd onto method.

Value

a function that given data generates a null data set. For use with lineup or rorschach

Examples

```
if (require("ggplot2") && require("reshape2")) {

x <- lm(tip ~ total_bill, data = tips)
tips.reg <- data.frame(tips, .resid = residuals(x), .fitted = fitted(x))
qplot(total_bill, .resid, data = tips.reg) %+%
  lineup(null_lm(tip ~ total_bill, method = "rotate"), tips.reg) +
  facet_wrap(~ .sample)
}
```

---

null_permute *Generate null data by permuting a variable.*

Description

Null hypothesis: variable is independent of others

Usage

```
null_permute(var)
```

Arguments

var	name of variable to permute

Value

a function that given data generates a null data set. For use with lineup or rorschach

opt_diff *finds the difference using* calc_diff *for all combinations of number of bins in x and y direction*

Description

finds the difference using calc_diff for all combinations of number of bins in x and y direction

Usage

```
opt_diff(lineup.dat, xlow, xhigh, ylow, yhigh, pos,
  plot = FALSE, m = 20)
```

Arguments

lineup.dat	lineup data to get the lineup
xlow	the lowest value of number of bins on the x-direction
xhigh	the highest value of number of bins on the x-direction
ylow	the lowest value of number of bins on the y-direction
yhigh	the highest value of number of bins on the y-direction
pos	position of the true plot in the lineup
plot	LOGICAL; if true, returns a tile plot for the combinations of number of bins with the differences as weights
m	number of plots in the lineup, by default m = 20

Value

a dataframe with the number of bins and differences the maximum mean distance of the null plots

reg_dist *Distance based on the regression parameters*

Description

Distance based on the regression parameters

Usage

```
reg_dist(X, PX, X.bin = 1, Y.bin = X.bin)
```

Arguments

X	a data.frame with two variables, the first column giving the explanatory variable and the second column giving the response variable
PX	another data.frame with two variables, the first column giving the explanatory variable and the second column giving the response variable

Value

distance between X and PX

resid_boot *Bootstrap residuals.*

Description

For use with null_lm

Usage

```
resid_boot(model, data)
```

Arguments

model	to extract residuals from
data	used to fit model

resid_pboot *Parametric bootstrap residuals.*

Description

For use with null_lm

Usage

```
resid_pboot(model, data)
```

Arguments

model	to extract residuals from
data	used to fit model

resid_rotate *Rotation residuals.*

Description

For use with null_lm

Usage

```
resid_rotate(model, data)
```

Arguments

model	to extract residuals from
data	used to fit model

resid_sigma *Residuals simulated by a normal model, with specified sigma*

Description

For use with null_lm

Usage

```
resid_sigma(model, data, sigma = 1)
```

Arguments

model	to extract residuals from
data	used to fit model
sigma,	a specific sigma to model

rorschach *The Rorschach protocol.*

Description

This protocol is used to calibrate the eyes for variation due to sampling. All plots are typically null data sets, data that is consistent with a null hypothesis. The protocol is described in Buja, Cook, Hofmann, Lawrence, Lee, Swayne, Wickham (2009) Statistical inference for exploratory data analysis and model diagnostics, Phil. Trans. R. Soc. A, 367, 4361-4383.

Usage

```
rorschach(method, true = NULL, n = 20, p = 0)
```

Arguments

method	method for generating null data sets
true	true data set. If NULL, find_plot_data will attempt to extract it from the current ggplot2 plot.
n	total number of samples to generate (including true data)
p	probability of including true data with null data.

sep_dist *Distance based on separation of clusters*

Description

distance based on the separation between clusters separation is the minimum distances of a point in the cluster to a a point of another cluster

Usage

```
sep_dist(X, PX, clustering = FALSE, nclust = 3)
```

Arguments

X	a data.frame with two or three columns, the first two columns providing the dataset
PX	a data.frame with two or three columns, the first two columns providing the dataset
clustering	LOGICAL; if TRUE, the third column is used as the clustering variable, by default FALSE
nclust	the number of clusters to be obtained by hierarchial clustering, by default nclust = 3

Value

distance between X and PX export

uni_dist *Distance for univariate data*

Description

distance is calculated based on the first four moments

Usage

```
uni_dist(X, PX)
```

Arguments

X	a data.frame where the first column is only used
PX	another data.frame where the first column is only used

Value

distance between X and PX

Index