

# Package ‘coRNAi’

October 17, 2017

**Type** Package

**Title** Analysis of co-knock-down RNAi data

**Version** 1.26.0

**Author** Elin Axelsson

**Maintainer** Elin Axelsson <elin.axelsson@imp.ac.at>

**Description** Analysis of combinatorial cell-based RNAi screens

**License** Artistic-2.0

**Depends** R (>= 2.10), cellHTS2, limma, locfit

**Imports** MASS, gplots, lattice, grDevices, graphics, stats

**SystemRequirements** Graphviz

**biocViews** CellBasedAssays

**PackageStatus** Deprecated

**NeedsCompilation** no

## R topics documented:

BoxPlotShorth . . . . .	2
cellHTS2df . . . . .	3
cortestmatrices . . . . .	3
data2graph . . . . .	4
df2array . . . . .	6
df2fitmatrix . . . . .	6
df2lmFit . . . . .	7
estmodel . . . . .	8
faultyscreen . . . . .	9
InteractGraph . . . . .	9
interactiontable . . . . .	10
InteractLevelPlot . . . . .	11
key . . . . .	12
LS main . . . . .	12
MainFitPlot . . . . .	13
PlotHeatmap . . . . .	14
Pplot . . . . .	15
replots . . . . .	16
screen1_raw . . . . .	17
screen2_raw . . . . .	17

signalplots . . . . .	18
tt2matrix . . . . .	19
updateDf . . . . .	19
weightDf . . . . .	20

<b>Index</b>	<b>22</b>
--------------	-----------

---

BoxPlotShorth	<i>Boxplot with horizontal bars at the midpoint of the shorth</i>
---------------	---

---

## Description

Produces a boxplot, but instead of horizontal bars at the median, the bars are at the midpoint of the shorth.

## Usage

```
BoxPlotShorth(formula, data = NULL, ...)
```

## Arguments

formula	formula for how the boxplot should be drawn.
data	the data to be used
...	other arguments to be passed to the plot function

## Value

a boxplot object

## Author(s)

Elin Axelsson

## See Also

See Also [boxplot.formula](#)

## Examples

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral= "Fluc")
BoxPlotShorth(value~replicate,df)
```

---

cellHTS2df	<i>converts cellHTS objects to dataframes</i>
------------	---

---

**Description**

converts a cellHTS2 object into a data.frame object and check that all mandatory meta data is included. Adds Directions, replicates and Pair columns.

**Usage**

```
cellHTS2df(x, neutral)
```

**Arguments**

x	a cellHTS object with correct annotations
neutral	string stating which RNAi is neutral (negative control)

**Value**

data.frame, with the data from the cellHTS object in column "value". Meta data from annotation file and the new columns; Directions, Replicate and Pair

**Author(s)**

Elin Axelsson

**See Also**

[cellHTS](#)

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw, neutral="Fluc")
head(df)
```

---

cortestmatrices	<i>Function to extract correlations and corresponding p-values from interaction matrix.</i>
-----------------	---

---

**Description**

This is a wrapper function for `cor.test`, given a matrix of interaction values, correlations and corresponding p-values for the genewise interaction profiles are calculated.

**Usage**

```
cortestmatrices(mat, method = c("pearson", "kendall", "spearman"))
```

**Arguments**

mat	mat interaction matrix
method	character deciding which correlation method should be used

**Value**

List of two matrices

cor.matrix	matrix with correlations
p.matrix	matrix with p-values

**Author(s)**

Elin Axelsson

**See Also**

[cor.test](#)

**Examples**

```
## simulate data with 2 genes with similar profiles

mat = matrix(rnorm(100*100,0,1),100,100)
pr = sample(2:10,100,replace=TRUE)
mat[1:2,] = mat[1:2,] + matrix(pr,ncol=100,nrow=2,byrow=TRUE)
mat = mat+t(mat)
diag(mat) = NA
dimnames(mat)=list(1:100,1:100)
res = cortestmatrices(mat,method="spearman")
cors= res[[1]]
ps = res[[2]]
print(which(ps==min(ps,na.rm=TRUE),arr.ind=TRUE))
```

---

data2graph

*Function to create .dot files for graph representation of data*

---

**Description**

From a interaction table or list of data matrices a .dot file is created for visualisation of the interaction/correlation network

**Usage**

```
data2graph(indata, sizethres=0, thres, thresBy = "P.Value", cols = c("blue", "white", "red"), gamm
```

**Arguments**

indata	
sizethres	numerical, lower treshold on the absolute effect size for edges
thres	threshold that should be used for interactions/correlations to be included in graph
thresBy	what data should the the threshold by used at. By default the p value from the moderated t test is used but one could also use e.g. the ordinary t or the size
cols	colors to be used in the plot, should be a character vector with the colors for low, neutral and hig values
gamma.col	Factor used to scale the colors
scaleFactor	Scale factor to adjust the distances beteen nodes in the graph
nodecolor	character or character vector, which color(s) should the nodes have. Should either be of lenth 1 (all nodes same color) or same length as the number of nodes.
writedot	logical, should a .dot file be created.
filename	charcter string with name of .dot file
width	width of the nodes
penwidth	width of the lines in the plot
shape	shape of the nodes in the plot
fixedsiz	should all nodes have the same size
fontsize	size of the font in the plot

**Value**

a .dot file is written if writefile argument is TRUE

ninf	Dataframe with information about the nodes in the network
einfo	Dataframe with information about the edges in the network

**Author(s)**

Greg Pau, Elin Axelsson

**See Also**

Also see Graphviz documantation <http://www.graphviz.org/>

**Examples**

#see vignette

---

df2array	<i>function to do go from data frame to an array with values</i>
----------	--

---

**Description**

The function transforms a float column in a data frame (from cHTS2df) to an array.

**Usage**

```
df2array(df, what)
```

**Arguments**

df	Data frame (from cellHTS2df) with additional column weights.
what	which column contains the data, eg. value or residuals

**Value**

An array with the data from the data frame.

**Author(s)**

Elin Axelsson

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw, neutral = "Fluc")
df$weights = as.numeric(df$type=="comb")

aa = df2array(df, what="value")

## see head for first replicate

head(aa[, ,1])
```

---

df2fitmatrix	<i>matrix for lmFit from dataframe</i>
--------------	--

---

**Description**

converts a dataframe into the right format for lmFit function

**Usage**

```
df2fitmatrix(df)
```

**Arguments**

df                      dataframe from cellHTS2df function

**Value**

A matrix with the genepairs as rows and the replicates as columns. This matrix is in the right format for the lmFit.

**Author(s)**

Elin Axelsson

**See Also**

[df2lmFit](#)

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
df = weightDf(df)
lmm = lmmain(df)
df = updateDf(df,lmm)
mfit = df2fitmatrix(df)
head(mfit)
```

---

df2lmFit

*lmFit from dataframe*

---

**Description**

converts a dataframe into the right format for lmFit function, calls the lmFit from limma and returns the result.

**Usage**

```
df2lmFit(df)
```

**Arguments**

df                      dataframe from cellHTS2df function

**Value**

Object of class 'MArrayLM'

**Author(s)**

Elin Axelsson

**See Also**[lmFit](#)**Examples**

```

data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
df = weightDf(df)
lmm = lmmain(df)
df = updateDf(df,lmm)
mfit = df2lmFit(df)
str(mfit)

```

---

estmodel

---

*Function to do estimate the main effects from data using median, mean or shorth.*


---

**Description**

The function estimates the main effect  $i$  from all data with the RNI $i$  against  $i$ . It can be done by median, mean or shorth.

**Usage**

```
estmodel(df, estimate = c("median", "mean", "shorth"), per = NULL)
```

**Arguments**

df	df data frame from cellHTS2df function with extra column weight (see weightDf)
estimate	estimate median, mean or shorth, decides how the main effects will be estimated.
per	per for which factor should the analysis be done separately, eg. batch or replicate.

**Value**

for per =NULL, a list with

coefficient	the estimated main effects
residuals	the residual after the main effects have been subtracted from the observations

for other per, a list of lists like the once described above, one for each level of the factor per.

**Author(s)**

Elin Axelsson



**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
df$weight = as.numeric(df$type=="comb")
main = estmodel(df,estimate="median")
str(main)
```

---

faultyscreen	<i>faulty screen</i>
--------------	----------------------

---

**Description**

A screen with both systematic errors and sporadic contaminations.

**Usage**

```
data(faultyscreen)
```

**Format**

The format is: chr "cellHTS"

**Examples**

```
data(faultyscreen)
## maybe str(faultyscreen) ; plot(faultyscreen) ...
```

---

InteractGraph	<i>functions to visualize interactions as a graph</i>
---------------	---

---

**Description**

visualizes significant interactions as a graph

**Usage**

```
InteractGraph(toptable, thresh, sizecutoff=0, by, key=FALSE, file="interactions", colors = list(neg="b
```

**Arguments**

toptable	toptable table from function topTable
thresh	thresh numeric, threshold for significance
sizecutoff	sizecutoff a minimal absolute size of a interaction for it to be included in the graph as an edge.
by	by column in topTable that thresh should be applied to
key	key optional, data frame with groupings of the genes in the toptable
file	file name of the file the results will be outputed to.
colors	colors list with colors to be used for pos interactions, neg interactions, key (nodes in key) and normal nodes.

**Value**

pdf file with graph

**Author(s)**

Elin Axelsson

**See Also**

[levelplot](#)

**Examples**

```
#see vignette
```

---

interactiontable	<i>Returns a list of interactions with associated statistics.</i>
------------------	---

---

**Description**

This is an extended wrapper around the topTable function from the limma package, as an option the ordinary t statistics can be calculated as well.

**Usage**

```
interactiontable(ebfit, sort = "none", ord.t = FALSE, correction = "BH")
```

**Arguments**

ebfit	ebfit a MArrayLM object produced by the eBayes function
sort	character string specifying which statistic to rank genes by, possible arguments are none, ID,size, t,B,adj.P.val,P.Value, and if ord.t = TRUE: ord.t, ord.p and ord.p.adj.
ord.t	Logical, should ordinary t statistics be calculated? Default is FALSE.
correction	method used to adjust the p-values for multiple testing. Default is BH. See p.adjust for the complete list of options.

**Value**

Returns a dataframe where the rows are the interaction pairs and the columns the statistics:

ID: Interaction pair id

size: the average interaction size

t: the moderated t statistics

P.Value: p-value for the moderated t statistics

adj.P.Val: adjusted p-value

B: the b statistics

if the ord.t=TRUE, the ordinary t statistics (ord.t), with corresponding p-values (ord.p) and adjusted p-values (ord.p.adj)

**Warning**

usage of the ordinary t statistics is not recommended for data sets with few replicates.

**Author(s)**

Elin Axelsson

**See Also**

[p.adjust](#), [topTable](#)

**Examples**

```
## simulated data
y <- matrix(rnorm(50*4,sd=1),50,4)
rownames(y) <- paste("Pair",1:50)

# fit and eBayes
fit <- lmFit(y)
fit <- eBayes(fit)
tt = interactiontable(fit,sort="size")
head(tt)
```

---

InteractLevelPlot      *function to visualize interactions as a levelplot*

---

**Description**

visualizes significant interactions as a levelplot

**Usage**

```
InteractLevelPlot(toptable, thresh = 0.001, by = "P.Value", key = FALSE, col.regions = colorRampPa
```

**Arguments**

toptable	toptable table from function topTable
thresh	thresh numeric, threshold for significance
by	by column in topTable that thresh should be applied to
key	key optional, data frame with groupings of the genes in the toptable
col.regions	col.regions colors to be used
zerolimit	zerolimit threshold below which interactions should be colored as 0.

**Value**

a levelplot, pdf files with graphs

**Author(s)**

Elin Axelsson

**See Also**

[levelplot](#)

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw, neutral="Fluc")
tt = data.frame("size"=runif(length(unique(df$Pair[df$Type=="comb"])), -2, 2), stringsAsFactors=FALSE)
rownames(tt) = unique(df$Pair[df$Type=="comb"])
InteractLevelPlot(tt, thres=0, by="size")
```

---

key	<i>A key to data set screen1, contains (additional) information about the genes in the screen.</i>
-----	--

---

**Description**

Contains information about which of the 16 genes in screen1 are cell cycle related. This is used in interaction graphs/plots.

**Usage**

```
data(key)
```

**Format**

A data frame with 16 observations on the following 2 variables.

GeneID a factor with levels AnnIX CG12785 CG16935 CG3165 CG7889 CG8108 CSN3 CSN4 CSN5 fwd  
pbl Rbf Rho1 sos trb1 zip  
cellCycle a numeric vector

**Examples**

```
data(key)
table(key$cellCycle)
```

---

LS main	<i>main effect estimation</i>
---------	-------------------------------

---

**Description**

for rlmmain the main effects are estimated using rlm function from MASS package, with lmmain the OLS is used.

**Usage**

```
rlmmain(df, per=NULL)
lmmain(df, per=NULL)
```

**Arguments**

df                   df dataframe created by function cellHTS2df  
 per                   string argument for which factor the analysis should be done separately, eg. replicate or batch

**Value**

lm,rlm

**Author(s)**

Elin Axelsson

**See Also**

[rlm,lm](#)

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
df = weightDf(df,exclude=c("controlP1","controlP2","controlN1","controlN2","controlP1N1","double"))
mains = rlmmain(df)
hist(coef(mains))
```

---

MainFitPlot

*Diagnostic plot*

---

**Description**

Plots residuals vs fitted values after fitting of main effects.

**Usage**

```
MainFitPlot(fit, xlab = "Fitted values", ylab = "Residuals", sd.fit = TRUE, main = "Residuals vs Fitted values")
```

**Arguments**

fit                   a fit from lmmmain, rlmmain or similar  
 xlab                  label for x-axis  
 ylab                  label for y-axis  
 sd.fit               logical, should the local estimator of the standard deviation be plotted  
 main                  main title for the plot  
 ...                   arguments to be passed on to the plot function

**Value**

a plot

**Author(s)**

Elin Axelsson

**See Also**[locfit](#)**Examples**

```
## simulated data

fitted.value = rnorm(100,2,1)
residuals = rnorm(100,0,1)
fit = list(fitted.value=fitted.value, residuals = residuals)
class(fit) = "lm"
MainFitPlot(fit)
```

PlotHeatmap

*Plot a heatmap of interactions***Description**

Plots a heatmap of the mean residuals for each interaction pair.

**Usage**

```
PlotHeatmap(toptable, colpal = colorRampPalette(c("blue", "white", "yellow")),
key=FALSE, margins=c(7,7), na.color="grey", breaks=seq(-1,1,by=0.01),...)
```

**Arguments**

toptable	a data frame created by with the interaction estimates as "logFC" and pair id as "ID". Usually created by topTable function in limma
colpal	color palette to be used in the plot
key	logical should a color key be included
margins	margins for plot
na.color	color for NA values
breaks	mapping data to colors in colpal
...	additional arguments to be passed to heatmap.2 call

**Value**

a plot

**Author(s)**

Elin Axelsson

**See Also**[heatmap.2](#)**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
tt = data.frame("size"=runif(length(unique(df$Pair[df$Type=="comb"])),-2,2),stringsAsFactors=FALSE)
rownames(tt) = unique(df$Pair[df$Type=="comb"])
PlotHeatmap(tt)
```

---

**Pplot***Function to plot cumulative p-values*

---

**Description**

Given a vector of p-values a cumulative p-value plot is produced

**Usage**

```
Pplot(x, col = "darkblue", maintitle="", nrpoints = 100, ...)
```

**Arguments**

x	vector with p-values
col	color to be used
maintitle	character, main plot title
nrpoints	numeric, how many points should be plotted
...	additional arguments passed on to the plot

**Value**

a plot

**Author(s)**

Wolfgang Huber

**Examples**

```
x = runif(1000,0,1)
Pplot(x,col="red",maintitle="uniform dist.", nrpoints=200)
```

---

replots	<i>reproducibility plots</i>
---------	------------------------------

---

### Description

plots reproducibility of replicates within/between screens

### Usage

```
BetweenScreenPlot(df, what="value",names,smooth=TRUE)
```

```
WithinScreenPlot(df, what="value",main="within-screen replicates",ylab ="technical replicate 2",x
```

### Arguments

df	df dataframe created by the cellHTS2df function
names	names optional, character vector with names of the different screens.
what	what what should be plotted, eg value or residuals
main	main main title
ylab	ylab label for y-axis
xlab	xlab label for x-axis
smooth	smooth should the smoothScatter function be called. Default is TRUE.
...	... further argument to be passed to the plot function

### Value

pairs plot

### Author(s)

Elin Axelsson

### See Also

[pairs,plot](#)

### Examples

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
BetweenScreenPlot(df)
WithinScreenPlot(df)
```



---

screen1_raw	<i>screen1 raw data</i>
-------------	-------------------------

---

**Description**

cellHTS2 object containing the raw data from screen1 (cellcycle related)

**Usage**

```
data(screen1_raw)
```

**Format**

The format is: chr "cellHTS"

**Examples**

```
data(screen1_raw)  
state(screen1_raw)
```

---

screen2_raw	<i>screen2 raw data</i>
-------------	-------------------------

---

**Description**

cellHTS2 object containing the raw data from screen2 (phospatases)

**Usage**

```
data(screen2_raw)
```

**Format**

The format is: chr "cellHTS"

**Examples**

```
data(screen2_raw)  
state(screen2_raw)
```

---

signalplots

*plot variation vs signal intensity*

---

### Description

plots the variation of replicates vs the mean intensity either by within screen replicate separately or over all screen replicates.

### Usage

```
SDplot(df, xlab="intensity mean",ylab="sd",add=FALSE,main,...)
MAplot(df, main,rank=FALSE)
```

### Arguments

df	df dataframe created by cellHTS2df function
main	main character string to be used as main title
xlab	xlab label for x-axis
ylab	ylab label for y-axis
add	add logical, should result be added to existing plot
rank	rank if TRUE the rank of the average intensities will be used
...	... further arguments to be passed to the plot function.

### Value

plot

### Author(s)

Elin Axelsson

### See Also

[plot](#)

### Examples

```
data(screen1_raw)
df=cellHTS2df(screen1_raw,neutral="Fluc")
MAplot(df,main="raw data")
SDplot(df,main="raw data")
```

---

tt2matrix

*Extracting data from a toptable and format it to matrix*


---

**Description**

Given an dataframe with data, typically from the `interactiontable`, the gene pair data is converted to a symmetric matrix.

**Usage**

```
tt2matrix(toptable, what)
```

**Arguments**

toptable	a dataframe with data for the pairwise interactions. Typically from the <code>interactiontable</code> function.
what	character indicating which of the columns in the dataframe should be used in the matrix.

**Value**

a symmetric matrix with the selected data for gene pair  $i,j$  in `matrix[i,j]` and `matrix[j,i]`

**Author(s)**

Elin Axelsson

**Examples**

```
## simulated data
mytoptable = data.frame("ID" = c("A B", "A C", "B C"), "size" = c(1:3), stringsAsFactors = FALSE)
rownames(mytoptable) = mytoptable$ID
mat = tt2matrix(mytoptable, what = "size")
```

---

updateDf

*updates dataframe after fitting of main estimates*


---

**Description**

adds residuals (value-main effects) to a dataframe

**Usage**

```
updateDf(df, lm, per=NULL)
```

**Arguments**

df	df dataframe created by cellHTS2df function
lm	lm lm objects, residuals from fitting main effects to data
per	per string argument, for which factor the analysis was done separately, eg. replicate.

**Value**

data frame with a new column "residuals"

**Author(s)**

Elin Axelsson

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw, neutral="Fluc")
df = weightDf(df, exclude=c("double", "controlP1", "controlP2", "controlN1", "controlN2", "controlP1N1"))
lmain = lmain(df)
df = updateDf(df, lmain)
hist(df$residuals)
```

---

weightDf	<i>Function to indicate which data points should be involved in down stream analysis.</i>
----------	---

---

**Description**

Function to do add weights to the data points in a data frame. At the time being 0 means excluded and everything >0 means included.

**Usage**

```
weightDf(df, exclude = c("double", "controlN2", "controlP2", "controlP1N1", "controlN1"))
```

**Arguments**

df	data frame from cellHTS2df
exclude	which type of data should be excluded from analysis.

**Details**

See also vignette for information about different "Type" types.

**Value**

A data frame with an added column 'weight'

**Author(s)**

Elin Axelsson

**Examples**

```
data(screen1_raw)
df = cellHTS2df(screen1_raw,neutral="Fluc")
##stupid example
df = weightDf(df,exclude="controlN2")
head(df)
```

# Index

\*Topic **datasets**  
  faultyscreen, 9  
  key, 12

\*Topic **hplot**  
  MainFitPlot, 13  
  replots, 16

\*Topic **models**  
  df2lmFit, 7

BetweenScreenPlot (replots), 16  
boxplot.formula, 2  
BoxPlotShorth, 2

cellHTS, 3  
cellHTS2df, 3  
cor.test, 4  
cortestmatrices, 3

data2graph, 4  
df2array, 6  
df2fitmatrix, 6  
df2lmFit, 7, 7

estmodel, 8

faultyscreen, 9

heatmap.2, 15

InteractGraph, 9  
interactiontable, 10  
InteractLevelPlot, 11

key, 12

levelplot, 10, 12  
lm, 13  
lmFit, 8  
lmmain (LS main), 12  
locfit, 14  
LS main, 12

MainFitPlot, 13  
MAplot (signalplots), 18

p.adjust, 11

pairs, 16  
plot, 16, 18  
PlotHeatmap, 14  
Pplot, 15

replots, 16  
rlm, 13  
rlmmain (LS main), 12

screen1\_raw, 17  
screen2\_raw, 17  
SDplot (signalplots), 18  
signalplots, 18

topTable, 11  
tt2matrix, 19

updateDf, 19

weightDf, 20  
WithinScreenPlot (replots), 16