Package: fullROC (via r-universe)

Type Package

Title Plot Full ROC Curves using Eyewitness Lineup Data

Version 0.1.1

Description Enable researchers to adjust identification rates using the 1/(lineup size) method, generate the full receiver operating characteristic (ROC) curves, and statistically compare the area under the curves (AUC). References: Yueran Yang & Andrew Smith. (2022). ``fullROC: An R package for generating and analyzing eyewitness-lineup ROC curves''. Behavior Research Methods. <doi:10.3758/s13428-022-01807-6>, Andrew Smith, Yueran Yang, & Gary Wells. (2020). ``Distinguishing between investigator discriminability and eyewitness discriminability: A method for creating full receiver operating characteristic curves of lineup identification performance''. Perspectives on Psychological Science, 15(3), 589-607. <doi:10.1177/1745691620902426>.

BugReports https://github.com/yuerany/fullROC/issues

Language en-US

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Roxygen list(markdown = TRUE)

Imports stats, graphics, grDevices

Repository https://yuerany.r-universe.dev

RemoteUrl https://github.com/yuerany/fullroc

RemoteRef HEAD

RemoteSha 3d529553e8bb24f4a860a52abe690d0e41fb6668

Contents

---

auc_boot	*Bootstrap AUCs*

Description

A function to simulate bootstrap samples and calculate AUC.

Usage

```
auc_boot(
  data,
  group = NULL,
  nboot = 1000,
  byDR = FALSE,
  ca_adj = FALSE,
  lsize = 6,
  csize = 3
)
```

Arguments

data	A data frame or matrix saving both cp and ca frequencies. cp must precede ca.
group	A vector indicating group membership. Will calculate AUCs by group.
nboot	Number of bootstrap iterations for each group. Defaults to 1,000.
byDR	Whether to order ids by diagnosticity ratios. Defaults to FALSE.
ca_adj	Whether to adjust id rates for ca lineups *after* simulating a sample from the unadjusted rates.
lsize	Size of lineup (used to adjust id rates). Defaults to 6.
csize	Number of confidence levels (used to adjust id rates). Defaults to 3.

Value

A list with simulated AUCs.

Examples

```
cpf <- c(100, 90, 80, 20, 10, 5)
caf <- c(6, 7, 15, 50, 75, 120)
auc_boot(cbind(cpf, caf), nboot = 100)
```

---

auc_ci   *Bootstrap confidence intervals for AUC*

Description

A function to simulate bootstrap samples and calculate CIs for AUC and differences.

Usage

```
auc_ci(cpf, caf, group = NULL, nboot = 1000, alpha = 0.05, ...)
```

Arguments

cpf	A vector of cp frequencies.
caf	A vector of ca frequencies.
group	A vector indicating group membership. Will calculate AUC by group.
nboot	Number of bootstrap iterations. Defaults to 1,000.
alpha	Alpha level for the CIs. Defaults to 0.05.
...	Additional arguments in auc_boot. Will allow users to adjust ca id rates in each simulation.

Value

A data frame of CIs for each group and group differences.

References

Yueran Yang & Andrew Smith. (2020). "fullROC: An R package for generating and analyzing eyewitness-lineup ROC curves" doi:10.13140/RG.2.2.20415.94885/1

Andrew Smith, Yueran Yang, & Gary Wells. (2020). "Distinguishing between investigator discriminability and eyewitness discriminability: A method for creating full receiver operating characteristic curves of lineup identification performance". *Perspectives on Psychological Science, 15*(3), 589-607. doi:10.1177/1745691620902426

Examples

```
cpf1 <- c(100, 90, 80, 20, 10, 5)
caf1 <- c(6, 7, 15, 50, 75, 120)
auc_ci(cpf1, caf1, nboot = 50)

cpf2 <- c(90, 40, 20)
caf2 <- c(10, 70, 80)
auc_ci(cpf2, caf2, nboot = 100)

## compare two groups
cpf <- c(cpf1, cpf2)
caf <- c(caf1, caf2)
group <- rep(letters[1:2], times = c(length(cpf1), length(cpf2) ) )
auc_ci(cpf, caf, group = group)
```

---

data_cum	*A function to generate cumulative id rates for both cp and ca lineups*

Description

A function to generate cumulative id rates for both cp and ca lineups

Usage

```
data_cum(data, byDR = FALSE)
```

Arguments

data	A matrix with both cp and ca id rates.
byDR	Whether to order ids by diagnosticity ratios. Defaults to FALSE.

Value

A data matrix with cumulative cp and ca id rates.

---

id_adj	*Simple adjustment*

Description

A function to adjust the id rates for ca lineups using the 1/(lineup size) method; is applicable to ordered id rates with the same confidence levels for all responses.

Usage

```
id_adj(rate, lsize = 6, csize = 3)
```

Arguments

rate	ID rate vector.
lsize	Lineup size. Defaults to 6.
csize	Number of confidence levels. Defaults to 3.

Value

Adjusted ID vector.

Examples

```
ca_id <- c(rep(0,3), rep(c(0.1, 0.15, 0.25), 2))
id_adj(ca_id)

## change line size to 5
id_adj(ca_id, lsize = 5)

## For multiple groups
ca_id2 <- c(c(rep(0,3), rep(c(0.1, 0.15, 0.25), 2)),
            c(rep(0,3), rep(c(0.1, 0.2, 0.3), 2)) )
group <- rep(c("a", "b"), each = 9)
## Adjust id rates by groups
by(ca_id2, group, id_adj)
```

---

id_adj_name *Match by confidence levels*

Description

A function to adjust the id rates for ca lineups using the 1/(lineup size) method; match and adjust id rates by names of confidence levels for both filler and suspect ids.

Usage

```
id_adj_name(rate, conf = NULL, fid, sid, lsize = 6)
```

Arguments

rate	ID rate vector.
conf	Confidence levels for the id rate vector. Default to be NULL.
fid	Mapping confidence levels from filler id.
sid	To-be-matched confidence levels for suspect id. Must have equal length as fid.
lsize	Lineup size. Defaults to 6.

Value

Adjusted ID vector.

Examples

```
ca_id <- c(rep(0,3), rep(c(0.1, 0.15, 0.25), 2))
names(ca_id) <- paste0(rep(c("IDS", "IDF", "REJ"), each = 3), c("high", "medium", "low"))

fid_conf <- paste0("IDF", c("high", "medium", "low"))
sid_conf <- paste0("IDS", c("high", "medium", "low"))

id_adj_name(ca_id, fid = fid_conf, sid = sid_conf)
```

---

id_adj_pos   *Match by position*

Description

A function to adjust the id rates for ca lineups using the 1/(lineup size) method; match and adjust id rates by positions of filler and suspect ids.

Usage

```
id_adj_pos(rate, fid, sid, lsize = 6)
```

Arguments

rate	ID rate vector.
fid	Mapping positions from filler id.
sid	To-be-matched positions for suspect id. Must have equal length as fid.
lsize	Lineup size. Defaults to 6.

Value

Adjusted ID vector.

Examples

```
ca_id <- c(rep(0,3), rep(c(0.1, 0.15, 0.25), 2))
id_adj_pos(ca_id, fid = 4:6, sid = 1:3, lsize = 5)
```

---

response_calculate   *A function to calculate responses from simulated memory distribution*

Description

A function to calculate responses from simulated memory distribution

Usage

```
response_calculate(memory, criterion, id_criterion = NULL, suspect = TRUE)
```

Arguments

memory	A simulated data matrix with suspect (column 1) and filler memory.
criterion	A vector of judgment criterion. Must have odd number of elements if id_criterion is not specified.
id_criterion	A number to define the criterion for id or rejection. Use the middle element of criterion if not specified.
suspect	Whether there is a designated suspect. Defaults to TRUE.

Value

A data matrix of id responses (IDS, IDF, or REJ) and confidence levels.

---

response_simu   *Simulate witness responses*

Description

A function to simulate both CP and CA responses

Usage

```
response_simu(
  guilt_diff,
  inno_diff = 0,
  n_sim = 1000,
  size = 6,
  inno_suspect = FALSE,
  criterion,
  id_criterion = NULL
)
```

Arguments

`guilt_diff`	Mean difference between guilty suspect and filler distributions.
`inno_diff`	Mean difference between innocent suspect and filler distributions. Defaults to 0.
`n_sim`	Number of simulations per condition. Defaults to 1,000.
`size`	Number of lineup members. Defaults to 6.
`inno_suspect`	Whether there is a designated innocent suspect. Defaults to FALSE.
`criterion`	A vector of response criteria. Must have odd number of elements if id_criterion is not specified.
`id_criterion`	A number to define the criterion for id or rejection. Use the middle element of criterion if not specified.

Value

A data frame including both CP and CA ID responses and confidence levels.

Examples

```
## Set up response criteria
rc1 <- seq(-1, 3, length.out = 5)

## no designated innocent suspect
response_simu(guilt_diff = 2, criterion = rc1)

## with a designated innocent suspect
response_simu(guilt_diff = 2, inno_diff = 0.2, inno_suspect = TRUE, criterion = rc1)

## define a criterion for id/rejection instead of using the middle criterion
response_simu(guilt_diff = 2, criterion = 0:3, id_criterion = 1)
```

---

roc_auc   *A function to calculate AUC using non-cumulative response rates.*

Description

A function to calculate AUC using non-cumulative response rates.

Usage

```
roc_auc(cpr, car, group = NULL, byDR = FALSE)
```

Arguments

`cpr`	A vector of cp id rates.
`car`	A vector of ca id rates.
`group`	A vector indicating group membership. If specified, will calculate AUC by group.
`byDR`	Whether to order ids by diagnosticity ratios. Defaults to FALSE.

Value

Area under the curve.

---

roc_line	*A function to add an ROC curve to an existing ROC plot.*

Description

A function to add an ROC curve to an existing ROC plot.

Usage

```
roc_line(cp, ca, byDR = FALSE, cumdata = FALSE, ...)
```

Arguments

cp	A vector of cp id rates or frequencies.
ca	A vector of ca id rates or frequencies.
byDR	Whether to order ids by diagnosticity ratios. Defaults to FALSE.
cumdata	Whether to output the cumulative data that are used to create the ROC curves. Default to FALSE.
...	Additional plotting parameters. For example, users can change x-axis and y-axis labels using xlab and ylab.

Value

Plot ROC curves and calculate AUCs as side effects.

---

roc_plot	*A function to plot ROC curves. Note that the NA values in the data will be replaced with zero.*

Description

A function to plot ROC curves. Note that the NA values in the data will be replaced with zero.

Usage

```
roc_plot(
  cp,
  ca,
  group = NULL,
  byDR = FALSE,
  cumdata = FALSE,
  grayscale = FALSE,
  ...
)
```

Arguments

`cp`	A vector of cp id rates or frequencies.
`ca`	A vector of ca id rates or frequencies.
`group`	Grouping variable to indicate group membership. Will create an ROC curve and calculate AUC for each group.
`byDR`	Whether to order ids by diagnosticity ratios. Defaults to FALSE.
`cumdata`	Whether to output the cumulative data that are used to create the ROC curves. Default to FALSE.
`grayscale`	Whether to produce the plot in grayscale. Defaults to FALSE.
`...`	Additional plotting parameters. For example, users can change x-axis and y-axis labels using `xlab` and `ylab`.

Value

Plot ROC curves and calculate AUCs as side effects.

References

Yueran Yang & Andrew Smith. (2022). "fullROC: An R package for generating and analyzing eyewitness-lineup ROC curves." *Behavior Research Methods.* doi:10.3758/s13428022018076

Andrew Smith, Yueran Yang, & Gary Wells. (2020). "Distinguishing between investigator discriminability and eyewitness discriminability: A method for creating full receiver operating characteristic curves of lineup identification performance". *Perspectives on Psychological Science, 15*(3), 589-607. doi:10.1177/1745691620902426

Examples

```
cpf1 <- c(100, 90, 80, 20, 10, 5)
caf1 <- c(6, 7, 15, 50, 75, 120)
roc_plot(cpf1, caf1)


cpf2 <- c(90, 40, 20)
caf2 <- c(10, 70, 80)
roc_plot(cpf2, caf2)

## plot two ROC curves
cpf <- c(cpf1, cpf2)
caf <- c(caf1, caf2)
group <- rep(letters[1:2], times = c(length(cpf1), length(cpf2) ) ) )
roc_plot(cpf, caf, group = group)
```

---

zroc_plot	*A function to plot z-ROC curves. Note that the NA values in the data will be replaced with zero.*

Description

A function to plot z-ROC curves. Note that the NA values in the data will be replaced with zero.

Usage

```
zroc_plot(cp, ca, group = NULL, byDR = FALSE, grayscale = FALSE, ...)
```

Arguments

cp	A vector of cp id rates or frequencies.
ca	A vector of ca id rates or frequencies.
group	Grouping variable to indicate group membership. Will create an ROC curve and calculate AUC for each group.
byDR	Whether to order ids by diagnosticity ratios. Defaults to FALSE.
grayscale	Whether to produce the plot in grayscale. Defaults to FALSE.
...	Additional plotting parameters. For example, users can change x-axis and y-axis labels using xlab and ylab.

Value

Plot z-ROC curves.

Examples

```
cpf1 <- c(100, 90, 80, 20, 10, 5)
caf1 <- c(6, 7, 15, 50, 75, 120)
zroc_plot(cpf1, caf1)


cpf2 <- c(90, 40, 20)
caf2 <- c(10, 70, 80)
zroc_plot(cpf2, caf2)

## plot two ROC curves
cpf <- c(cpf1, cpf2)
caf <- c(caf1, caf2)
group <- rep(letters[1:2], times = c(length(cpf1), length(cpf2) ) )
zroc_plot(cpf, caf, group = group)
```

Index