

Package ‘braQCA’

January 29, 2022

Title Bootstrapped Robustness Assessment for Qualitative Comparative Analysis

Version 1.2.1.29

Description Test the robustness of a user's Qualitative Comparative Analysis solutions to randomness, using the bootstrapped assessment: `baQCA()`. This package also includes a function that provides recommendations for improving solutions to reach typical significance levels: `brQCA()`. Data included come from McVeigh et al. (2014) <[doi:10.1177/0003122414534065](https://doi.org/10.1177/0003122414534065)>.

Depends R (>= 3.2.3)

Imports QCA, dplyr, bootstrap

License GPL-3

LazyData true

NeedsCompilation no

RoxygenNote 7.1.1

Author C. Ben Gibson [aut] (<<https://orcid.org/0000-0003-2561-9574>>),
Burrell Vann Jr [aut, cre] (<<https://orcid.org/0000-0003-3066-5815>>)

Maintainer Burrell Vann Jr <bvannjr@sdsu.edu>

Repository CRAN

Date/Publication 2022-01-29 16:30:02 UTC

R topics documented:

baQCA	2
brQCA	3
conf.table	4
rallies	5
sim.brQCA	5
Index	7

baQCA

*Bootstrapped Assessment***Description**

This function performs the the Bootstrapped Assessment for QCA (baQCA) on a given QCA model object.

Usage

```
baQCA(
  mod,
  sim = 2000,
  include = c(""),
  row.dom = FALSE,
  omit = c(),
  dir.exp = c()
)
```

Arguments

<code>mod</code>	name of the QCA eqmcc/minimize model object.
<code>sim</code>	the number of simulations the baQCA function should run. Default set to <code>sim=2000</code> .
<code>include</code>	[from QCA package] "A vector of additional output function values to be included in the minimization." Default set to <code>include=c("")</code> .
<code>row.dom</code>	[from QCA package] "Logical, impose row dominance as constraint on solution to eliminate dominated inessential prime implicants." Default set to <code>FALSE</code> .
<code>omit</code>	[from QCA package] "A vector of configuration index values or matrix of configurations to be omitted from minimization." Default set to <code>omit=c()</code> .
<code>dir.exp</code>	[from QCA package] "A vector of directional expectations for deriving intermediate solutions." Default set to <code>dir.exp=c()</code> .

Value

After some time, this function returns the probability that the data will return a random result given the parameters set by the researcher in the model (configurational n threshold, consistency score threshold, etc), as well a confidence interval around this value. This value is interpreted similarly to a p-value, i.e. a .05 value coincides with a 95% "confidence level."

Examples

```
qca.data <- rallies[,8:13]
rownames(qca.data) <- rownames(rallies)
truth <- QCA::truthTable(qca.data, outcome="P", sort.by="incl", incl.cut1=0.85, n.cut=1, show.cases=TRUE)
mod1 <- QCA::minimize(truth, details=TRUE, show.cases=TRUE)
```

```
baQCA(mod1, sim=1)
```

brQCA

Bootstrapped Recommendation

Description

Provides recommendations for consistency score and configurational n thresholds to attain a desired level of confidence in a QCA algorithm application.

Usage

```
brQCA(
  qca.data,
  outcome = "OUT",
  type = "crisp",
  inclcut = "",
  ncut = 2,
  neg.out = FALSE,
  sim = 10,
  verbose = TRUE
)
```

Arguments

qca.data	the QCA data frame.
outcome	the outcome variable in the QCA data frame of causal conditions; "OUT" is the outcome variable for an application of QCA.
type	of QCA application, "crisp" or "fuzzy" sets. Default set to type = "crisp".
inclcut	range of consistency scores for inclusion. If not specified, this defaults to seq(from = 0.5, to = 1, by = 0.01).
ncut	configurational n levels to simulate. Can be altered to give options for the range of minimum to maximum ncut value that the truth table yields, by naming the truth table object (e.g. truth) and calling the minimum and maximum number of cases, using ncut=min(truth\$tt\$n):max(truth\$tt\$n) identified by the truth table. Default set to ncut=2.
neg.out	[from QCA package] "Logical, use negation of outcome (ignored if data is a truth table object)." Default set to neg.out=FALSE.
sim	number of simulations to run for each combination of parameters. The final number of simulations is length(inclcut)*length(ncut)*sim*2. Default set to sim=10.
verbose	prints the system time used to run the simulation and the percent complete. Default set to verbose=TRUE.

Value

Significance levels reached (.10,.05, .01, .001) when specifying a combination of inclcut, ncut, and neg.out in a QCA model.

Examples

```
qca.data <- rallies[,8:13]

## Not run:
brQCA(qca.data, outcome="P", ncut=5, sim=1)

## End(Not run)
```

 conf.table

Configuration Table

Description

Internal function; calculates via logistic regression the output of the Bootstrapped Robustness Recommendation

Usage

```
conf.table(data, ncut = ncut)
```

Arguments

data	name of the model object; the table of solutions for an application of QCA. Default set to data.
ncut	configurational n levels for inclusion. Default set to ncut=4

Value

The output of the Bootstrapped Recommendation #' @export

rallies	<i>McVeigh et al. (2014) Tea Party Data</i>
---------	---

Description

This data set is an abbreviated version of the data set used by McVeigh et al. (2014). These data cover all 67 counties in Florida, and come from the American Community Survey (2005-2009).

Usage

rallies

Format

A data frame with 67 observations and 13 variables.

tprallies	number of Tea Party rallies in county, 2009-2010
reppct2008	percent of county vote for the Republican Presidential candidate (John McCain) in 2008
dempct2008	percent of county vote for the Democratic Presidential candidate (Barack Obama) in 2008
pctBA25	percent of county, aged 25 or older, with a bachelor's degree
pctunemp	percent of county that is unemployed
pctevang	percent of county that belongs to an Evangelical denomination
pctblack	percent of county that identifies as Black
P	binary. 0 if county had no Tea Party rallies, 1 if county had <i>at least</i> on Tea Party rally
R	binary. 0 if the majority of votes in the county were for the Democratic Presidential candidate (Barack Obama)
C	binary. 0 if percent of county with a bachelor's degree was below-average for Florida, 1 if percent of county was above-average
U	binary. 0 if percent unemployed in county was below-average for Florida, 1 if percent unemployed in county was above-average
E	binary. 0 if percent Evangelical in county was below-average for Florida, 1 if percent Evangelical in county was above-average
B	binary. 0 if percent Black in county was below-average for Florida, 1 if percent Black in county was at or above average

sim.brQCA	<i>Simulation Application</i>
-----------	-------------------------------

Description

Internal function to calculate the Bootstrapped Recommendation.

Usage

```
sim.brQCA(
  qca.data,
  outcome = "OUT",
  conditions = c(""),
```

```

sim = 10,
ncut = 2,
type = "crisp",
inclcut = "",
neg.out = FALSE,
verbose = TRUE
)

```

Arguments

qca.data	the data frame of causal conditions.
outcome	the outcome variable (object name) in the QCA data frame of causal conditions; "OUT" is the outcome variable for an application of QCA. Default set to outcome="OUT".
conditions	a set of causal conditions. Default set to conditions=c("")
sim	number of simulations to run. Default set to sim=10.
ncut	configurational n levels for inclusion. Default set to ncut=2.
type	type of QCA application, "crisp" or "fuzzy" sets. Default set to type = "crisp".
inclcut	minimum sufficiency score for inclusion. Default set to inclcut="".
neg.out	[from QCA package] "Logical, use negation of outcome (ignored if data is a truth table object)." Default set to neg.out=FALSE.
verbose	prints the system time used to run the simulation and the percent complete. Default set to verbose=TRUE.

Value

Simulation information later passed on to conf.table.

Index

- * **datasets**
 - rallies, [5](#)
- baQCA, [2](#)
- brQCA, [3](#)
- conf.table, [4](#)
- rallies, [5](#)
- sim.brQCA, [5](#)