Package: rdpower (via r-universe)

May 18, 2026

Type Package

Title Power, Sample Size, and Minimum Detectable Effect Calculations for RD Designs

Version 3.0

Date 2026-05-15

Description The regression discontinuity (RD) design is a popular quasi-experimental design for causal inference and policy evaluation. The 'rdpower' package provides tools to perform power, sample size, and minimum detectable effect calculations in RD designs: rdpower() calculates the power of an RD design, rdsampsi() calculates the required sample size to achieve a desired power, and rdmde() calculates minimum detectable effects. See Cattaneo, Titiunik and Vazquez-Bare (2019) <https://rdpackages.github.io/references/Cattaneo-Titiunik-VazquezBare_2019_Stata.pdf> for further methodological details.

Encoding UTF-8

Depends R (>= 3.6.0)

License GPL-3

URL https://github.com/rdpackages/rdpower

BugReports https://github.com/rdpackages/rdpower/issues

Imports rdrobust (>= 4.0.0)

RoxygenNote 7.3.2

NeedsCompilation no

Author Matias D. Cattaneo [aut, cre], Rocio Titiunik [aut], Gonzalo Vazquez-Bare [aut]

Maintainer Matias D. Cattaneo <matias.d.cattaneo@gmail.com>

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

Date/Publication 2026-05-18 00:15:32 UTC

RemoteUrl https://github.com/cran/rdpower

RemoteRef HEAD

RemoteSha 2745c824694ad86628b0aee1ba7f9c679c344f9a

Contents

rdpower-package 2
rdmde 3
rdpower 6
rdsampsi 10
summary.rdpower 14

Index **16**

`rdpower-package` *rdpower: Power and Sample Size Calculations for RD Designs*

Description

The regression discontinuity (RD) design is a popular quasi-experimental design for causal inference and policy evaluation. The `'rdpower'` package provides tools to perform power, sample size and MDE calculations in RD designs: `rdpower()` calculates the power of an RD design, `rdsampsi()` calculates the required sample size to achieve a desired power and `rdmde()` calculates minimum detectable effects. This package relies on the `rdrobust` package. See Calonico, Cattaneo and Titiunik (2014, 2015) and Calonico, Cattaneo, Farrell and Titiunik (2017). For more details, and related `Stata` and `R` packages useful for analysis of RD designs, visit `https://rdpackages.github.io/`.

Author(s)

Matias D. Cattaneo, Princeton University. <matias.d.cattaneo@gmail.com>

Rocio Titiunik, Princeton University. <rocio.titiunik@gmail.com>

Gonzalo Vazquez-Bare, UC Santa Barbara. <gvazquezbare@gmail.com>

References

Calonico, S., M. D. Cattaneo, M. Farrell and R. Titiunik. (2017).`rdrobust`: Software for Regression Discontinuity Designs. *Stata Journal* 17(2): 372-404.

Calonico, S., M. D. Cattaneo, and R. Titiunik. (2014). Robust Data-Driven Inference in the Regression-Discontinuity Design. *Stata Journal* 14(4): 909-946.

Calonico, S., M. D. Cattaneo, and R. Titiunik. (2015).`rdrobust`: An R Package for Robust Nonparametric Inference in Regression-Discontinuity Designs. *R Journal* 7(1): 38-51.

Cattaneo, M. D., R. Titiunik and G. Vazquez-Bare. (2019). Power Calculations for Regression Discontinuity Designs. *Stata Journal*, 19(1): 210-245.

rdmde *MDE Calculations for RD Designs*

Description

rdmde() performs MDE calculations for RD designs.

Usage

```
rdmde(
  data = NULL,
  cutoff = 0,
  alpha = 0.05,
  beta = 0.8,
  nsamples = NULL,
  sampsi = NULL,
  samph = NULL,
  bias = NULL,
  variance = NULL,
  init.cond = NULL,
  covs = NULL,
  covs_drop = TRUE,
  deriv = 0,
  p = 1,
  q = NULL,
  h = NULL,
  b = NULL,
  rho = NULL,
  kernel = "triangular",
  bwselect = "mserd",
  vce = "nn",
  cluster = NULL,
  nnmatch = 3,
  scalepar = 1,
  scaleregul = 1,
  sharpbw = FALSE,
  fuzzy = NULL,
  level = 95,
  weights = NULL,
  masspoints = "adjust",
  bwcheck = NULL,
  bwrestrict = TRUE,
  stdvars = FALSE,
  subset = NULL,
  ginv.tol = 1e-20
)
```

Arguments

`data`	a matrix (Y,R) containing the outcome variable and the running variable (as column vectors).
`cutoff`	the RD cutoff (default is 0).
`alpha`	specifies the significance level for the power function. Default is 0.05.
`beta`	specifies the desired power. Default is 0.8.
`nsamples`	sets the total sample size to the left, sample size to the left inside the bandwidth, total sample size to the right and sample size to the right of the cutoff inside the bandwidth to calculate the variance when the running variable is not specified. When not specified, the values are calculated using the running variable.
`sampsi`	sets the sample size at each side of the cutoff for power calculation. The first number is the sample size to the left of the cutoff and the second number is the sample size to the right. Default values are the sample sizes inside the chosen bandwidth.
`samph`	sets the bandwidths at each side of the cutoff for power calculation. The first number is the bandwidth to the left of the cutoff and the second number is the bandwidth to the right. Default values are the bandwidths used by `rdrobust`.
`bias`	set bias to the left and right of the cutoff. If not specified, the biases are estimated using `rdrobust`.
`variance`	set variance to the left and right of the cutoff. If not specified, the variances are estimated using `rdrobust`.
`init.cond`	sets the initial condition for the Newton-Raphson algorithm that finds the MDE. Default is 0.2 times the standard deviation of the outcome below the cutoff.
`covs`	option for `rdrobust()`: specifies additional covariates to be used for estimation and inference.
`covs_drop`	option for `rdrobust()`: if TRUE, it checks for collinear additional covariates and drops them. Default is TRUE.
`deriv`	option for `rdrobust()`: specifies the order of the derivative of the regression functions to be estimated.
`p`	option for `rdrobust()`: specifies the order of the local-polynomial used to construct the point-estimator.
`q`	option for `rdrobust()`: specifies the order of the local-polynomial used to construct the bias-correction.
`h`	option for `rdrobust()`: specifies the values of the main bandwidth to be used on the left and on the right of the cutoff, respectively.
`b`	option for `rdrobust()`: specifies the values of the bias bandwidth b to be used on the left and on the right of the cutoff, respectively.
`rho`	option for `rdrobust()`: specifies the value of `rho` so that the bias bandwidth `b` equals `b=h/rho`.
`kernel`	option for `rdrobust()`: kernel function used to construct the local-polynomial estimators.
`bwselect`	option for `rdrobust()`: specifies the bandwidth selection procedure to be used.

vce	option for rdrobust(): specifies the variance-covariance estimator. Current options are nn, hc0, hc1, hc2, hc3, and, with cluster specified, cr1, cr2, and cr3.
cluster	option for rdrobust(): indicates the cluster ID variable used for cluster-robust variance estimation.
nnmatch	option for rdrobust(): minimum number of neighbors for vce=nn. Default is nnmatch=3.
scalepar	option for rdrobust(): specifies scaling factor for RD parameter of interest.
scaleregul	option for rdrobust(): specifies scaling factor for the regularization terms of bandwidth selectors.
sharpbw	option for rdrobust(): if TRUE, fuzzy RD estimation uses bandwidth selection for the sharp RD model.
fuzzy	option for rdrobust(): specifies the treatment status variable used to implement fuzzy RD estimation.
level	option for rdrobust(): sets the confidence level for confidence intervals.
weights	option for rdrobust(): is the variable used for optional weighting of the estimation procedure. The unit-specific weights multiply the kernel function.
masspoints	option for rdrobust(): checks and controls for repeated observations in tue running variable.
bwcheck	option for rdrobust(): if a positive integer is provided, the preliminary bandwidth used in the calculations is enlarged so that at least bwcheck unique observations are used.
bwrestrict	option for rdrobust(): if TRUE, computed bandwidths are restricted to lie withing the range of x. Default is bwrestrict=TRUE.
stdvars	option for rdrobust(): if TRUE, x and y are standardized before computing the bandwidths. Default is stdvars=FALSE.
subset	option for rdrobust(): optional vector specifying a subset of observations to use.
ginv.tol	option for rdrobust(): tolerance used to invert matrices involving covariates when covs_drop=TRUE.

Value

mde	MDE using robust bias corrected standard error
se.rbc	robust bias corrected standard error
sampsi.r	number of observations inside the window to the right of the cutoff
sampsi.l	number of observations inside the window to the left of the cutoff
samph.r	bandwidth to the right of the cutoff
samph.l	bandwidth to the left of the cutoff
alpha	significance level used in power function
bias.r	bias to the right of the cutoff
bias.l	bias to the left of the cutoff

`Vr.rb`	Robust bias corrected variance to the right of the cutoff
`Vl.rb`	Robust bias corrected variance to the left of the cutoff
`N.r`	Total sample size to the right of the cutoff
`N.l`	Total sample size to the left of the cutoff
`mde.conv`	MDE using conventional inference
`se.conv`	conventional standard error

Author(s)

Matias D. Cattaneo, Princeton University. <matias.d.cattaneo@gmail.com>

Rocio Titiunik, Princeton University. <rocio.titiunik@gmail.com>

Gonzalo Vazquez-Bare, UC Santa Barbara. <gvazquezbare@gmail.com>

References

Cattaneo, M. D., R. Titiunik and G. Vazquez-Bare. (2019). Power Calculations for Regression Discontinuity Designs. *Stata Journal*, 19(1): 210-245.

Examples

```
# Toy dataset
X <- array(rnorm(2000),dim=c(1000,2))
R <- X[,1] + X[,2] + rnorm(1000)
Y <- 1 + R -.5*R^2 + .3*R^3 + (R>=0) + rnorm(1000)
# MDE calculation
tmp <- rdmde(data=cbind(Y,R),init.cond=0.5)
```

`rdpower` *Power Calculations for RD Designs*

Description

`rdpower()` performs power calculations for RD designs.

Usage

```
rdpower(
  data = NULL,
  cutoff = 0,
  tau = NULL,
  alpha = 0.05,
  nsamples = NULL,
  sampsi = NULL,
  samph = NULL,
```

```
  bias = NULL,
  variance = NULL,
  plot = FALSE,
  graph.range = NULL,
  covs = NULL,
  covs_drop = TRUE,
  deriv = 0,
  p = 1,
  q = NULL,
  h = NULL,
  b = NULL,
  rho = NULL,
  kernel = "triangular",
  bwselect = "mserd",
  vce = "nn",
  cluster = NULL,
  nnmatch = 3,
  scalepar = 1,
  scaleregul = 1,
  sharpbw = FALSE,
  fuzzy = NULL,
  level = 95,
  weights = NULL,
  masspoints = "adjust",
  bwcheck = NULL,
  bwrestrict = TRUE,
  stdvars = FALSE,
  subset = NULL,
  ginv.tol = 1e-20
)
```

Arguments

`data`	a matrix (Y,R) containing the outcome variable and the running variable (as column vectors).
`cutoff`	the RD cutoff (default is 0).
`tau`	specifies the treatment effect under the alternative at which the power function is evaluated. The default is half the standard deviation of the outcome for the untreated group.
`alpha`	specifies the significance level for the power function. Default is 0.05.
`nsamples`	sets the total sample size to the left, sample size to the left inside the bandwidth, total sample size to the right and sample size to the right of the cutoff inside the bandwidth to calculate the variance when the running variable is not specified. When not specified, the values are calculated using the running variable.
`sampsi`	sets the sample size at each side of the cutoff for power calculation. The first number is the sample size to the left of the cutoff and the second number is the sample size to the right. Default values are the sample sizes inside the chosen bandwidth.

samph	sets the bandwidths at each side of the cutoff for power calculation. The first number is the bandwidth to the left of the cutoff and the second number is the bandwidth to the right. Default values are the bandwidths used by rdrobust.
bias	set bias to the left and right of the cutoff. If not specified, the biases are estimated using rdrobust.
variance	set variance to the left and right of the cutoff. If not specified, the variances are estimated using rdrobust.
plot	plots the power function using the conventional and robust bias corrected standard errors from rdrobust.
graph.range	range of the plot.
covs	option for rdrobust(): specifies additional covariates to be used for estimation and inference.
covs_drop	option for rdrobust(): if TRUE, it checks for collinear additional covariates and drops them. Default is TRUE.
deriv	option for rdrobust(): specifies the order of the derivative of the regression functions to be estimated.
p	option for rdrobust(): specifies the order of the local-polynomial used to construct the point-estimator.
q	option for rdrobust(): specifies the order of the local-polynomial used to construct the bias-correction.
h	option for rdrobust(): specifies the values of the main bandwidth to be used on the left and on the right of the cutoff, respectively.
b	option for rdrobust(): specifies the values of the bias bandwidth b to be used on the left and on the right of the cutoff, respectively.
rho	option for rdrobust(): specifies the value of rho so that the bias bandwidth b equals b=h/rho.
kernel	option for rdrobust(): kernel function used to construct the local-polynomial estimators.
bwselect	option for rdrobust(): specifies the bandwidth selection procedure to be used.
vce	option for rdrobust(): specifies the variance-covariance estimator. Current options are nn, hc0, hc1, hc2, hc3, and, with cluster specified, cr1, cr2, and cr3.
cluster	option for rdrobust(): indicates the cluster ID variable used for cluster-robust variance estimation.
nnmatch	option for rdrobust(): minimum number of neighbors for vce=nn. Default is nnmatch=3.
scalepar	option for rdrobust(): specifies scaling factor for RD parameter of interest.
scaleregul	option for rdrobust(): specifies scaling factor for the regularization terms of bandwidth selectors.
sharpbw	option for rdrobust(): if TRUE, fuzzy RD estimation uses bandwidth selection for the sharp RD model.
fuzzy	option for rdrobust(): specifies the treatment status variable used to implement fuzzy RD estimation.

`level`	option for `rdrobust()`: sets the confidence level for confidence intervals.
`weights`	option for `rdrobust()`: is the variable used for optional weighting of the estimation procedure. The unit-specific weights multiply the kernel function.
`masspoints`	option for `rdrobust()`: checks and controls for repeated observations in tue running variable.
`bwcheck`	option for `rdrobust()`: if a positive integer is provided, the preliminary bandwidth used in the calculations is enlarged so that at least `bwcheck` unique observations are used.
`bwrestrict`	option for `rdrobust()`: if TRUE, computed bandwidths are restricted to lie withing the range of `x`. Default is `bwrestrict=TRUE`.
`stdvars`	option for `rdrobust()`: if `TRUE`, `x` and `y` are standardized before computing the bandwidths. Default is `stdvars=FALSE`.
`subset`	option for `rdrobust()`: optional vector specifying a subset of observations to use.
`ginv.tol`	option for `rdrobust()`: tolerance used to invert matrices involving covariates when `covs_drop=TRUE`.

Value

`power.rbc`	power against tau using robust bias corrected standard error
`se.rbc`	robust bias corrected standard error
`sampsi.r`	number of observations inside the window to the right of the cutoff
`sampsi.l`	number of observations inside the window to the left of the cutoff
`samph.r`	bandwidth to the right of the cutoff
`samph.l`	bandwidth to the left of the cutoff
`alpha`	significance level used in power function
`tau`	treatment effect under alternative hypothesis
`bias.r`	bias to the right of the cutoff
`bias.l`	bias to the left of the cutoff
`Vr.rb`	Robust bias corrected variance to the right of the cutoff
`Vl.rb`	Robust bias corrected variance to the left of the cutoff
`N.r`	Total sample size to the right of the cutoff
`N.l`	Total sample size to the left of the cutoff
`power.conv`	power against tau using conventional inference
`se.conv`	conventional standard error

Author(s)

Matias D. Cattaneo, Princeton University. <matias.d.cattaneo@gmail.com>

Rocio Titiunik, Princeton University. <rocio.titiunik@gmail.com>

Gonzalo Vazquez-Bare, UC Santa Barbara. <gvazquezbare@gmail.com>

References

Cattaneo, M. D., R. Titiunik and G. Vazquez-Bare. (2019). Power Calculations for Regression Discontinuity Designs. *Stata Journal*, 19(1): 210-245.

Examples

```
# Toy dataset
X <- array(rnorm(2000),dim=c(1000,2))
R <- X[,1] + X[,2] + rnorm(1000)
Y <- 1 + R -.5*R^2 + .3*R^3 + (R>=0) + rnorm(1000)
# Power against tau = 1
tmp <- rdpower(data=cbind(Y,R),tau=1)
# Power against tau = 1 including covariates
tmp <- rdpower(data=cbind(Y,R),tau=1,covs=X)
```

`rdsampsi` *Sample Size Calculations for RD Designs*

Description

`rdsampsi()` performs sample size calculations for RD designs.

Usage

```
rdsampsi(
  data = NULL,
  cutoff = 0,
  tau = NULL,
  alpha = 0.05,
  beta = 0.8,
  samph = NULL,
  nsamples = NULL,
  bias = NULL,
  variance = NULL,
  nratio = NULL,
  init.cond = NULL,
  plot = FALSE,
  graph.range = NULL,
  covs = NULL,
  covs_drop = TRUE,
  deriv = 0,
  p = 1,
  q = NULL,
  h = NULL,
  b = NULL,
```

```
  rho = NULL,
  kernel = "triangular",
  bwselect = "mserd",
  vce = "nn",
  cluster = NULL,
  nnmatch = 3,
  scalepar = 1,
  scaleregul = 1,
  sharpbw = FALSE,
  fuzzy = NULL,
  level = 95,
  weights = NULL,
  masspoints = "adjust",
  bwcheck = NULL,
  bwrestrict = TRUE,
  stdvars = FALSE,
  subset = NULL,
  ginv.tol = 1e-20
)
```

Arguments

`data`	a matrix (Y,R) containing the outcome variable and the running variable (as column vectors).
`cutoff`	the RD cutoff (default is 0).
`tau`	specifies the treatment effect under the alternative at which the power function is evaluated. The default is half the standard deviation of the outcome for the untreated group.
`alpha`	specifies the significance level for the power function. Default is 0.05.
`beta`	specifies the desired power. Default is 0.8.
`samph`	sets the bandwidths at each side of the cutoff for power calculation. The first number is the bandwidth to the left of the cutoff and the second number is the bandwidth to the right. Default values are the bandwidths used by `rdrobust`.
`nsamples`	sets the total sample size to the left, sample size to the left inside the bandwidth, total sample size to the right and sample size to the right of the cutoff inside the bandwidth to calculate the variance when the running variable is not specified. When not specified, the values are calculated using the running variable.
`bias`	set bias to the left and right of the cutoff. If not specified, the biases are estimated using `rdrobust`.
`variance`	set variance to the left and right of the cutoff. If not specified, the variances are estimated using `rdrobust`.
`nratio`	specifies the proportion of treated units in the window. Default is the ratio of the standard deviation of the treated to the sum of the standard deviations for treated and controls.
`init.cond`	sets the initial condition for the Newton-Raphson algorithm that finds the sample size. Default is the number of observations in the sample with non-missing values of the outcome and running variable.

plot	plots the power function using the conventional and robust bias corrected standard errors from rdrobust.
graph.range	range of the plot.
covs	option for rdrobust(): specifies additional covariates to be used for estimation and inference.
covs_drop	option for rdrobust(): if TRUE, it checks for collinear additional covariates and drops them. Default is TRUE.
deriv	option for rdrobust(): specifies the order of the derivative of the regression functions to be estimated.
p	option for rdrobust(): specifies the order of the local-polynomial used to construct the point-estimator.
q	option for rdrobust(): specifies the order of the local-polynomial used to construct the bias-correction.
h	option for rdrobust(): specifies the values of the main bandwidth to be used on the left and on the right of the cutoff, respectively.
b	option for rdrobust(): specifies the values of the bias bandwidth b to be used on the left and on the right of the cutoff, respectively.
rho	option for rdrobust(): specifies the value of rho so that the bias bandwidth b equals b=h/rho.
kernel	option for rdrobust(): kernel function used to construct the local-polynomial estimators.
bwselect	option for rdrobust(): specifies the bandwidth selection procedure to be used.
vce	option for rdrobust(): specifies the variance-covariance estimator. Current options are nn, hc0, hc1, hc2, hc3, and, with cluster specified, cr1, cr2, and cr3.
cluster	option for rdrobust(): indicates the cluster ID variable used for cluster-robust variance estimation.
nnmatch	option for rdrobust(): minimum number of neighbors for vce=nn. Default is nnmatch=3.
scalepar	option for rdrobust(): specifies scaling factor for RD parameter of interest.
scaleregul	option for rdrobust(): specifies scaling factor for the regularization terms of bandwidth selectors.
sharpbw	option for rdrobust(): if TRUE, fuzzy RD estimation uses bandwidth selection for the sharp RD model.
fuzzy	option for rdrobust(): specifies the treatment status variable used to implement fuzzy RD estimation.
level	option for rdrobust(): sets the confidence level for confidence intervals.
weights	option for rdrobust(): is the variable used for optional weighting of the estimation procedure. The unit-specific weights multiply the kernel function.
masspoints	option for rdrobust(): checks and controls for repeated observations in tue running variable.

bwcheck	option for rdrobust(): if a positive integer is provided, the preliminary bandwidth used in the calculations is enlarged so that at least bwcheck unique observations are used.
bwrestrict	option for rdrobust(): if TRUE, computed bandwidths are restricted to lie withing the range of x. Default is bwrestrict=TRUE.
stdvars	option for rdrobust(): if TRUE, x and y are standardized before computing the bandwidths. Default is stdvars=FALSE.
subset	option for rdrobust(): optional vector specifying a subset of observations to use.
ginv.tol	option for rdrobust(): tolerance used to invert matrices involving covariates when covs_drop=TRUE.

Value

alpha	significance level
beta	desired power
tau	treatment effect under alternative hypothesis
sampsi.h.tot	total number of observations inside the window
sampsi.h.r	number of observations inside the window to the right of the cutoff
sampsi.h.l	number of observations inside the window to the left of the cutoff
N.r	Total sample size to the right of the cutoff
N.l	Total sample size to the left of the cutoff
samph.r	bandwidth to the right of the cutoff
samph.l	bandwidth to the left of the cutoff
var.r	Robust bias corrected variance to the right of the cutoff
Var.l	Robust bias corrected variance to the left of the cutoff
sampsi.h.tot.cl	implied total number of observations inside the window using conventional s.e.
sampsi.h.r.cl	number of observations inside the window to the right of the cutoff using conventional s.e.
sampsi.h.l.cl	number of observations inside the window to the left of the cutoff using conventional s.e.
no.iter	number of iterations until convergence of the Newton-Raphson algorithm
init.cond	initial condition of the Newton-Raphson algorithm

Author(s)

Matias D. Cattaneo, Princeton University. <matias.d.cattaneo@gmail.com>

Rocio Titiunik, Princeton University. <rocio.titiunik@gmail.com>

Gonzalo Vazquez-Bare, UC Santa Barbara. <gvazquezbare@gmail.com>

References

Cattaneo, M. D., R. Titiunik and G. Vazquez-Bare. (2019). Power Calculations for Regression Discontinuity Designs. *Stata Journal*, 19(1): 210-245.

Examples

```
# Toy dataset
X <- array(rnorm(2000),dim=c(1000,2))
R <- X[,1] + X[,2] + rnorm(1000)
Y <- 1 + R -.5*R^2 + .3*R^3 + (R>=0) + rnorm(1000)
# Sample size to achieve power of 0.8 against tau = 1
tmp <- rdsampsi(data=cbind(Y,R),tau=1)
# Sample size against tau = 1 including covariates
tmp <- rdsampsi(data=cbind(Y,R),tau=1,covs=X)
```

`summary.rdpower` *Summarize RDPOWER objects*

Description

`summary()` prints the default robust bias-corrected output. Use `all = TRUE` to also display the conventional output.

Usage

```
## S3 method for class 'rdpower'
summary(object, ..., all = FALSE)

## S3 method for class 'rdpower'
print(x, ...)

## S3 method for class 'summary.rdpower'
print(x, ...)

## S3 method for class 'rdsampsi'
summary(object, ..., all = FALSE)

## S3 method for class 'rdsampsi'
print(x, ...)

## S3 method for class 'summary.rdsampsi'
print(x, ...)

## S3 method for class 'rdmde'
summary(object, ..., all = FALSE)
```

```
## S3 method for class 'rdmde'
print(x, ...)

## S3 method for class 'summary.rdmde'
print(x, ...)
```

Arguments

object	object returned by rdpower(), rdsampsi(), or rdmde().
...	additional arguments.
all	displays the conventional output in addition to the robust bias-corrected output.
x	object returned by rdpower(), rdsampsi(), or rdmde().

Value

A summary object containing the original return values, returned invisibly by the print method.

Index

print.rdmde *(*summary.rdpower*)*, 14
print.rdpower *(*summary.rdpower*)*, 14
print.rdsampsi *(*summary.rdpower*)*, 14
print.summary.rdmde *(*summary.rdpower*)*, 14
print.summary.rdpower *(*summary.rdpower*)*, 14
print.summary.rdsampsi *(*summary.rdpower*)*, 14

rdmde, *2*, 3, *15*
rdpower, *2*, 6, *15*
rdpower-package, 2
rdpower_package *(*rdpower-package*)*, 2
rdsampsi, *2*, 10, *15*

summary.rdmde *(*summary.rdpower*)*, 14
summary.rdpower, 14
summary.rdsampsi *(*summary.rdpower*)*, 14