Stata Statistics/Data Analysis

**Title**

**rdsampsi** -- Sample size selection based on power calculations for Regression Discontinuity designs using robust bias-corrected local polynomial inference.

**Syntax**

**rdsampsi** *depvar runvar* [*if*] [*in*] [, **c(**#**)** **tau(**#**)** **alpha(**#**)** **beta(**#**)** **nsamples(**# # # #**)** **samph(**# #**)** **all** **plot** **graph_range(**# #**)** **graph_step(**#**)** **graph_options(***graph_opt***)** **bias(**# #**)** **variance(**# #**)** **nratio(**#**)** **init_cond(**#**)** **covs(***covars***)** **deriv(**#**)** **p(**#**)** **q(**#**)** **h(**# #**)** **b(**# #**)** **rho(**#**)** **fuzzy(***fuzzyvar [sharpbw]***)** **kernel(***kernelfn***)** **bwselect(***bwmethod***)** **vce(***vcetype [vceopt1 vceopt2]***)** **scalepar(**#**)** **scaleregul(**#**)** ]

**Description**

**rdsampsi** provides sample size selection based on power calculations in Regression Discontinuity designs using conventional and robust bias-corrected local polynomial methods. Companion command is: rdpower for power calculations.

A detailed introduction to this command is given in Cattaneo, Titiunik and Gonzalo Vazquez-Bare (2018).

Companion R functions are also available here.

This command employs the Stata (and R) package rdrobust for underlying calculations. See Calonico, Cattaneo and Titiunik (2014) and Calonico, Cattaneo, Farrell and Titiunik (2017) for more details.

Related Stata and R packages useful for inference in RD designs are described in the following website:

https://sites.google.com/site/rdpackages/

**Options**

**c(**#**)** specifies the RD cutoff for *indepvar*. Default is **c(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 **alpha(.05)**.

**beta(**#**)** specifies the desired power. Default is **beta(.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 this option is not specified, the values are calculated using the running variable.

**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**.

**all** displays the power using the conventional variance estimator, in addition to the robust bias corrected one.

**plot** plots the power function using the robust bias corrected standard error from **rdrobust**. If **all** is specified, the conventional power function is also plotted.

**graph_range(**# #**)** specifies the range of the plot when **plot** option is used. Default range is [-1.5*tau ; 1.5*tau].

**graph_step(**#**)** specifies the step increment of the plot when **plot** option is used. Default range is 0.2*range.

**graph_options(**#**)** specifies the graph options (title, axes titles, etc) to be passed to the plot when **plot** option is used.

**bias(**# #**)** allows the user to set bias to the left and right of the cutoff. If not specified, the biases are estimated using **rdrobust**.

**variance(**# #**)** allows the user to 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.

The following options are passed directly to **rdrobust**:

**covs(***covars***)** specifies additional covariates to be used for estimation and inference.

**deriv(**#**)** specifies the order of the derivative of the regression functions to be estimated. Default is **deriv(0)**. Setting **deriv(1)** results in estimation of a Kink RD design (up to scale).

**p(**#**)** specifies the order of the local polynomial used to construct the point estimator. Default is **p(1)** (local linear regression).

**q(**#**)** specifies the order of the local polynomial used to construct the bias correction. Default is **q(2)** (local quadratic regression).

**h(**# #**)** specifies the main bandwidth (*h*) used to construct the RD point estimator. If not specified, bandwidth *h* is computed by the companion command rdbwselect. If two bandwidths are specified, the first bandwidth is used for the data below the cutoff and the second bandwidth is used for the data above the cutoff.

**b(**# #**)** specifies the bias bandwidth (*b*) used to construct the bias-correction estimator. If not specified, bandwidth *b* is computed by the companion command rdbwselect. If two bandwidths are specified, the first bandwidth is used for the data below the cutoff and the second bandwidth is used for the data above the cutoff.

**rho(**#**)** specifies the value of *rho*, so that the bias bandwidth *b* equals *b*=*h*/*rho*. Default is **rho(1)** if *h* is specified but *b* is not.

**fuzzy(***fuzzyvar [sharpbw]***)** specifies the treatment status variable used to implement fuzzy RD estimation (or Fuzzy Kink RD if **deriv(1)** is also specified). Default is Sharp RD design and hence this option is not used. If the option *sharpbw* is set, the fuzzy RD estimation is performed using a bandwidth selection procedure for the sharp RD model. This option is automatically selected if there is perfect compliance at either side of the threshold.

**kernel(***kernelfn***)** specifies the kernel function used to construct the local-polynomial estimator(s). Options are: **triangular**, **epanechnikov**, and **uniform**. Default is **kernel(triangular)**.

**bwselect(***bwmethod***)** specifies the bandwidth selection procedure to be used. By default it computes both *h* and *b*, unless *rho* is specified, in which case it only computes *h* and sets *b*=*h*/*rho*. Options are:

**mserd** one common MSE-optimal bandwidth selector for the RD treatment effect estimator.
**msetwo** two different MSE-optimal bandwidth selectors (below and above the cutoff) for the RD treatment effect estimator.
**msesum** one common MSE-optimal bandwidth selector for the sum of regression estimates (as opposed to difference thereof).
**msecomb1** for min(**mserd**,**msesum**).
**msecomb2** for median(**msetwo**,**mserd**,**msesum**), for each side of the cutoff separately.
**cerrd** one common CER-optimal bandwidth selector for the RD treatment effect estimator.
**certwo** two different CER-optimal bandwidth selectors (below and above the cutoff) for the RD treatment effect estimator.
**cersum** one common CER-optimal bandwidth selector for the sum of regression estimates (as opposed to difference thereof).
**cercomb1** for min(**cerrd**,**cersum**).
**cercomb2** for median(**certwo**,**cerrd**,**cersum**), for each side of the cutoff separately.
Note: MSE = Mean Square Error; CER = Coverage Error Rate.
Default is **bwselect(mserd)**. For details on implementation see Calonico, Cattaneo and Titiunik (2014a), Calonico, Cattaneo and Farrell (2016a), and Calonico, Cattaneo, Farrell and Titiunik (2016), and the companion software articles.

**vce(***vcetype [vceopt1 vceopt2]***)** specifies the procedure used to compute the variance-covariance matrix estimator. Options are:

**vce(nn** *[nnmatch]***)** for heteroskedasticity-robust nearest neighbor variance estimator with *nnmatch* indicating the minimum number of neighbors to be used.
**vce(hc0)** for heteroskedasticity-robust plug-in residuals variance estimator without weights.
**vce(hc1)** for heteroskedasticity-robust plug-in residuals variance estimator with *hc1* weights.
**vce(hc2)** for heteroskedasticity-robust plug-in residuals variance estimator with *hc2* weights.
**vce(hc3)** for heteroskedasticity-robust plug-in residuals variance estimator with *hc3* weights.
**vce(nncluster** *clustervar [nnmatch]***)** for cluster-robust nearest neighbor variance estimation using with *clustervar* indicating the cluster ID variable and *nnmatch* matches indicating the minimum number of neighbors to be used.
**vce(cluster** *clustervar***)** for cluster-robust plug-in residuals variance estimation with degrees-of-freedom weights and *clustervar* indicating the cluster ID variable.
Default is **vce(nn 3)**.

**scalepar(**#**)** specifies scaling factor for RD parameter of interest. This option is useful when the estimator of interest requires a known multiplicative factor rescaling (e.g., Sharp Kink RD). Default is **scalepar(1)** (no rescaling).

**scaleregul(**#**)** specifies scaling factor for the regularization term added to the denominator of the bandwidth selectors. Setting **scaleregul(0)** removes the regularization term from the bandwidth selectors. Default is **scaleregul(1)**.

---

**Example: Cattaneo, Frandsen and Titiunik (2015) Incumbency Data**

Setup
**. use rdpower_senate.dta**

Sample size calculation against an alternative hypothesis of tau = 5
**. rdsampsi demvoteshfor2 demmv, tau(5)**

Sample size calculation with covariates
    **. rdsampsi demvoteshfor2 demmv, tau(5) covs(population dopen dmidterm)**

Sample size calculation with user-specified bandwidths
    **. rdsampsi demvoteshfor2 demmv, tau(5) h(16 18) b(18 20)**

Sample size calculation with user-specified options
    **. rdsampsi demvoteshfor2 demmv, tau(5) beta(.9) all samph(18 19) nratio(.5)**

Power function plot with default options
    **. rdsampsi demvoteshfor2 demmv, tau(5) plot**

Power function plot with user-specified range and step
    **. rdsampsi demvoteshfor2 demmv, tau(5) plot graph_range(0 800) graph_step(200)**

Power function plot with user-specified options
    **. rdsampsi demvoteshfor2 demmv, tau(5) plot graph_range(0 800) graph_step(200) graph_options(title(Power function) xtitle(sample size) ytitle(power) graphregion(fcolor(white)))**

## Saved results

**rdsampsi** saves the following in **r()**:

Scalars

| | |
|---|---|
| **r(alpha)** | significance level |
| **r(beta)** | desired power |
| **r(tau)** | desired effect |
| **r(samph_l)** | bandwidth to the left of the cutoff |
| **r(samph_h)** | bandwidth to the right of the cutoff |
| **r(var_l)** | robust bias corrected variance to the left of the cutoff |
| **r(var_r)** | robust bias corrected variance to the right of the cutoff |
| **r(bias_l)** | bias to the left of the cutoff |
| **r(bias_r)** | bias to the right of the cutoff |
| **r(N_h_l)** | sample size in bandwidth to the left of the cutoff for variance calculation |
| **r(N_h_r)** | sample size in bandwidth to the right of the cutoff for variance calculation |
| **r(N_l)** | sample size to the left of the cutoff for variance calculation |
| **r(N_r)** | sample size to the right of the cutoff for variance calculation |
| **r(sampsi_tot)** | implied total sample size using robust bias corrected s.e. |
| **r(sampsi_h_l)** | sample size to the left of the cutoff using robust bias corrected s.e. |
| **r(sampsi_h_r)** | sample size to the right of the cutoff using robust bias corrected s.e. |
| **r(sampsi_h_tot)** | sample size inside the window using robust bias corrected s.e. |
| **r(var_l_cl)** | conventional variance to the left of the cutoff |
| **r(var_r_cl)** | conventional variance to the right of the cutoff |
| **r(sampsi_tot_cl)** | implied total sample size using conventional s.e. |
| **r(sampsi_h_l_cl)** | sample size to the left of the cutoff using conventional s.e. |
| **r(sampsi_h_r_cl)** | sample size to the right of the cutoff using conventional s.e. |
| **r(sampsi_h_tot_cl)** | sample size inside the window using conventional s.e. |
| **r(no_iter)** | number of iterations until convergence of the Newton-Raphson algorithm |
| **r(init_cond)** | initial condition of the Newton-Raphson algorithm |

## References

Calonico, S., M. D. Cattaneo, M. H. 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.

Cattaneo, M. D., Frandsen, B., and R. Titiunik. 2015. Randomization Inference in the Regression Discontinuity Design: An Application to Party Advantages in the U.S. Senate.
*Journal of Causal Inference* 3(1): 1-24.

Cattaneo, M. D., R. Titiunik, and G. Vazquez-Bare. 2018. Power Calculations for Regression Discontinuity Designs.
Forthcoming, *Stata Journal*.

**Authors**

Matias D. Cattaneo, University of Michigan, Ann Arbor, MI. cattaneo@umich.edu.

Rocio Titiunik, University of Michigan, Ann Arbor, MI. titiunik@umich.edu.

Gonzalo Vazquez-Bare, University of California, Santa Barbara. gvazquez@econ.ucsb.edu.