

# sreg: Stratified Randomized Experiments



The `sreg` package for R, offers a toolkit for estimating average treatment effects (ATEs) in stratified randomized experiments. The package is designed to accommodate scenarios with multiple treatments and cluster-level treatment assignments, and accomodates optimal linear covariate adjustment based on baseline observable characteristics. The package computes estimators and standard errors based on Bugni, Canay, Shaikh (2018); Bugni, Canay, Shaikh, Tabord-Meehan (2023); and Jiang, Linton, Tang, Zhang (2023).

**Dependencies:** `dplyr`, `tidyr`, `extraDistr`, `rlang`

**Suggests:** `haven`, `knitr`, `rmarkdown`, `testthat` ( $\geq 3.0.0$ )

**R version required:**  $\geq 2.10$

[Latest Build \(v.1.0.1.9000\)](#)

## Authors

- Juri Trifonov [jutrifonov@uchicago.edu](mailto:jutrifonov@uchicago.edu)
- Yuehao Bai [yuehao.bai@usc.edu](mailto:yuehao.bai@usc.edu)
- Azeem Shaikh [amshaikh@uchicago.edu](mailto:amshaikh@uchicago.edu)
- Max Tabord-Meehan [maxtm@uchicago.edu](mailto:maxtm@uchicago.edu)

## Supplementary files

- PDF version of the manual: [Download PDF](#)
- Sketch of the derivation of the ATE variance estimator under cluster-level treatment assignment: [Download PDF](#)
- Expressions for the multiple treatment case (with and without clusters): [Download PDF](#)

## Installation

Install the official CRAN release using:

```
install.packages("sreg")
```

```
trying URL 'https://mirror.las.iastate.edu/CRAN/src/contrib/sreg_1.0.1.tar.gz'
```

```
Content type 'application/x-gzip' length 42789 bytes (41 KB)
```

```
=====
```

```
downloaded 41 KB
```

```
* installing *source* package 'sreg' ...
** package 'sreg' successfully unpacked and MD5 sums checked
** using staged installation
** R
** data
*** moving datasets to lazyload DB
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
```

```

** testing if installed package can be loaded from temporary location
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (sreg)

```

The downloaded source packages are in

```

'/private/var/folders/mp/06gjwr8j56zdp5j2vgdkd4z40000gq/T/RtmpHqU4e9/downloaded_packages'

```

```

library(sreg)

```

```

#> _____ Stratified Randomized
#> / ____| _ \| ____/ ____| Experiments
#> \__ \| |_) | _|| | _
#> ____| | _<| | | | |
#> |___/|_| \_\____\____| version 1.0.1

```

```

#> Type 'citation("sreg")' for citing this R package in publications.

```

The latest development version can be installed using devtools.

```

library(devtools)

```

```

install_github("jutrifonov/sreg")

```

```

Downloading GitHub repo jutrifonov/sreg@HEAD

```

```

— R CMD build —

```

```

✓ checking for file '/private/var/folders/mp/06gjwr8j56zdp5j2vgdkd4z40000gq...
— preparing 'sreg':
✓ checking DESCRIPTION meta-information
— checking for LF line-endings in source and make files and shell scripts
— checking for empty or unneeded directories
— building 'sreg_1.0.1.9000.tar.gz'

```

```

* installing *source* package 'sreg' ...
** using staged installation
** R
** data
*** moving datasets to lazyload DB
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded from temporary location
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (sreg)

```

```

library(sreg)

```

```

#> _____ Stratified Randomized
#> / ____| _ \| ____/ ____| Experiments
#> \__ \| |_) | _|| | _
#> ____| | _<| | | | |
#> |___/|_| \_\____\____| version 1.0.1.9000

```

```

#> Type 'citation("sreg")' for citing this R package in publications.

```

## The function `sreg()`

Estimates the ATE(s) and the corresponding standard error(s) for a (collection of) treatment(s) relative to a control.

### Syntax

```
sreg(Y, S = NULL, D, G.id = NULL, Ng = NULL, X = NULL, HC1 = TRUE)
```

### Arguments

- **Y** - a numeric vector/matrix/data.frame/tibble of the observed outcomes;
- **S** - a numeric vector/matrix/data.frame/tibble of strata indicators 0, 1, 2, ...; if NULL then the estimation is performed assuming no stratification;
- **D** - a numeric vector/matrix/data.frame/tibble of treatments indexed by 0, 1, 2, ..., where D = 0 denotes the control;
- **G.id** - a numeric vector/matrix/data.frame/tibble of cluster indicators; if NULL then estimation is performed assuming treatment is assigned at the individual level;
- **Ng** - a numeric vector/matrix/data.frame/tibble of cluster sizes; if NULL then Ng is assumed to be equal to the number of available observations in every cluster;
- **X** - a matrix/data.frame/tibble with columns representing the covariate values for every observation; if NULL then the estimator without linear adjustments is applied [ $\hat{\cdot}$ ];
- **HC1** - a TRUE/FALSE logical argument indicating whether the small sample correction should be applied to the variance estimator. [ $\hat{\cdot}$ ]: *Note: sreg cannot use individual-level covariates for covariate adjustment in cluster-randomized experiments. Any individual-level covariates will be aggregated to their cluster-level averages.*

### Data Structure

Here we provide an example of a data frame that can be used with `sreg`.

Y	S	D	G.id	Ng	x_1	x_2
-0.57773576	2	0	1	10	1.5597899	0.03023334
1.69495638	2	0	1	10	1.5597899	0.03023334
2.02033740	4	2	2	30	0.8747419	-0.77090031
1.22020493	4	2	2	30	0.8747419	-0.77090031
1.64466086	4	2	2	30	0.8747419	-0.77090031
-0.32365109	4	2	2	30	0.8747419	-0.77090031
2.21008191	4	2	2	30	0.8747419	-0.77090031
-2.25064316	4	2	2	30	0.8747419	-0.77090031
0.37962312	4	2	2	30	0.8747419	-0.77090031

### Summary

`sreg` prints a “Stata-style” table containing the ATE estimates, corresponding standard errors,  $t$ -statistics,  $p$ -values, 95% asymptotic confidence intervals, and significance indicators for different levels  $\alpha$ . The example of the printed output is provided below.

Saturated Model Estimation Results under CAR with clusters and linear adjustments

Observations: 30000

Clusters: 1000

Number of treatments: 2

Number of strata: 4

Covariates used in linear adjustments: x\_1, x\_2

---

Coefficients:

	Tau	As.se	T-stat	P-value	CI.left(95%)	CI.right(95%)	Significance
1	0.01614	0.04513	0.35753	0.72069	-0.07232	0.1046	
2	0.78642	0.04642	16.94263	0.00000	0.69545	0.8774	***

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Return Value

The function returns an object of class `sreg` that is a list containing the following elements:

- **tau.hat** - a  $1 \times |\mathcal{A}|$  vector of ATE estimates, where  $|\mathcal{A}|$  represents the number of treatments;
- **se.rob** - a  $1 \times |\mathcal{A}|$  vector of standard errors estimates, where  $|\mathcal{A}|$  represents the number of treatments;
- **t.stat** - a  $1 \times |\mathcal{A}|$  vector of  $t$ -statistics, where  $|\mathcal{A}|$  represents the number of treatments;
- **p.value** - a  $1 \times |\mathcal{A}|$  vector of corresponding  $p$ -values, where  $|\mathcal{A}|$  represents the number of treatments;
- **CI.left** - a  $1 \times |\mathcal{A}|$  vector of the left bounds of the 95% as. confidence interval;
- **CI.right** - a  $1 \times |\mathcal{A}|$  vector of the right bounds of the 95% as. confidence interval;
- **data** - an original data of the form `data.frame(Y, S, D, G.id, Ng, X)`;
- **lin.adj** - a `data.frame` representing the covariates that were used in implementing linear adjustments.

## Empirical Example

Here, we provide the empirical application example using the data from (Chong et al., 2016), who studied the effect of iron deficiency anemia on school-age children's educational attainment and cognitive ability in Peru. The example replicates the empirical illustration from (Bugni et al., 2019). For replication purposes, the data is included in the package and can be accessed by running `data("AEJapp")`. This example can be accessed directly in R via `help(sreg)`.

```
library(sreg, dplyr, haven)
```

The description of the dataset can be accessed using `help()`:

```
help(AEJapp)
```

We can upload the AEJapp dataset to the R session via `data()`:

```
data("AEJapp")
data <- AEJapp
```

It is pretty straightforward to prepare the data to fit the package syntax using `dplyr`:

```
Y <- data$gradesq34
D <- data$treatment
S <- data$class_level
data.clean <- data.frame(Y, D, S)
data.clean <- data.clean %>%
  mutate(D = ifelse(D == 3, 0, D))
Y <- data.clean$Y
D <- data.clean$D
S <- data.clean$S
head(data.clean)
  Y D S
1 11.2 1 1
2 12.4 0 3
3 11.9 0 5
4 13.1 0 1
```

```
5 13.4 2 2
6 10.7 0 1
```

We can take a look at the frequency table of D and S:

```
table(D = data.clean$D, S = data.clean$S)
  S
D   1  2  3  4  5
0  15 19 16 12 10
1  16 19 15 10 10
2  17 20 15 11 10
```

Now, it is straightforward to replicate the results from (Bugni et al, 2019) using `sreg`:

```
result <- sreg::sreg(Y = Y, S = S, D = D)
print(result)

Saturated Model Estimation Results under CAR
Observations: 215
Number of treatments: 2
Number of strata: 5
---
Coefficients:
      Tau   As.se  T-stat P-value CI.left(95%) CI.right(95%) Significance
1 -0.05113 0.20645 -0.24766 0.80440   -0.45577    0.35351
2  0.40903 0.20651  1.98065 0.04763    0.00427    0.81379          *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Besides that, `sreg` allows adding linear adjustments (covariates) to the estimation procedure:

```
pills <- data$pills_taken
age <- data$age_months
data.clean <- data.frame(Y, D, S, pills, age)
data.clean <- data.clean %>%
  mutate(D = ifelse(D == 3, 0, D))
Y <- data.clean$Y
D <- data.clean$D
S <- data.clean$S
X <- data.frame("pills" = data.clean$pills, "age" = data.clean$age)
result <- sreg::sreg(Y, S, D, G.id = NULL, X = X)
print(result)

Saturated Model Estimation Results under CAR
Observations: 215
Number of treatments: 2
Number of strata: 5
Covariates used in linear adjustments: pills, age
---
Coefficients:
      Tau   As.se  T-stat P-value CI.left(95%) CI.right(95%) Significance
1 -0.02862 0.17964 -0.15929 0.87344   -0.38071    0.32348
2  0.34609 0.18362  1.88477 0.05946   -0.01381    0.70598          .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## The function `sreg.rgen()`

Generates the observed outcomes, treatment assignments, strata indicators, cluster indicators, cluster sizes, and covariates for estimating the treatment effect following the stratified block randomization design under covariate-adaptive randomization (CAR).

### Syntax

```
sreg.rgen(n, Nmax = 50, n.strata,
          tau.vec = c(0), gamma.vec = c(0.4, 0.2, 1),
          cluster = TRUE, is.cov = TRUE)
```

### Arguments

- **n** - a total number of observations in a sample;
- **Nmax** - a maximum size of generated clusters (maximum number of observations in a cluster);
- **n.strata** - an integer specifying the number of strata;
- **tau.vec** - a numeric  $1 \times |\mathcal{A}|$  vector of treatment effects, where  $|\mathcal{A}|$  represents the number of treatments;
- **gamma.vec** - a numeric  $1 \times 3$  vector of parameters corresponding to covariates;
- **cluster** - a TRUE/FALSE argument indicating whether the dgp should use a cluster-level treatment assignment or individual-level;
- **is.cov** - a TRUE/FALSE argument indicating whether the dgp should include covariates or not.

### Return Value

- **Y** - a numeric  $n \times 1$  vector of the observed outcomes;
- **S** - a numeric  $n \times 1$  vector of strata indicators;
- **D** - a numeric  $n \times 1$  vector of treatments indexed by  $0, 1, 2, \dots$ , where  $D = 0$  denotes the control;
- **G.id** - a numeric  $n \times 1$  vector of cluster indicators;
- **Ng** - a numeric vector/matrix/data.frame of cluster sizes; if NULL then Ng is assumed to be equal to the number of available observations in every cluster;
- **X** - a data.frame with columns representing the covariate values for every observation.

### Example

```
library(sreg)
data <- sreg.rgen(n = 1000, tau.vec = c(0), n.strata = 4, cluster = TRUE)
> head(data)
```

	Y	S	D	x_1	x_2
1	1.717293	1	0	4.772092	2.4138491
2	2.553695	2	0	5.413440	2.0551019
3	2.237556	3	2	6.611161	0.9300293
4	1.825809	3	1	2.735503	1.7839981
5	5.536280	2	2	2.469239	2.0495611
6	1.628753	2	0	4.887561	2.1327071

### References

- Bugni, F. A., Canay, I. A., and Shaikh, A. M. (2018). Inference Under Covariate-Adaptive Randomization. *Journal of the American Statistical Association*, 113(524), 1784–1796, doi:10.1080/01621459.2017.1375934.
- Bugni, F., Canay, I., Shaikh, A., and Tabord-Meehan, M. (2024+). Inference for Cluster Randomized Experiments with Non-ignorable Cluster Sizes. *Forthcoming in the Journal of Political Economy: Microeconomics*, doi:10.48550/arXiv.2204.08356.

Jiang, L., Linton, O. B., Tang, H., and Zhang, Y. (2023+). Improving Estimation Efficiency via Regression-Adjustment in Covariate-Adaptive Randomizations with Imperfect Compliance. *Forthcoming in Review of Economics and Statistics*, doi:10.48550/arXiv.2204.08356.