

Package ‘sreg’

Juri Trifonov

Latest version is available at <https://github.com/yurytrifonov/sreg>

April 8, 2024

General Description

The ‘sreg’ package 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).

The function sreg()

sreg	<i>Estimates ATEs, standard errors, p-values & confidence intervals for stratified experiments</i>
-------------	--

Description

This function 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 = FALSE)
```

Arguments

Y	a numeric $n \times 1$ vector of the <i>observed outcomes</i> .
S	a numeric $n \times 1$ vector of <i>strata indicators</i> ; if NULL then the estimation is performed assuming no stratification.
D	a numeric $n \times 1$ vector of <i>treatments</i> indexed by $\{0, 1, 2, \dots\}$, where $D = 0$ denotes the control.
G.id	a numeric $n \times 1$ vector of <i>cluster indicators</i> ; if NULL then estimation is performed assuming treatment is assigned at the individual level.
Ng	a numeric $n \times 1$ vector of <i>cluster sizes</i> ; 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 cluster-level <i>covariate values</i> for every observation; if NULL then the estimator without linear adjustments is applied. (<i>Note: if individual-level covariates with varying values within clusters are provided, then the cluster-level averaging is implemented to estimate the model</i>).
HC1	a TRUE/FALSE logical argument indicating whether the small sample correction should be applied to the variance estimator.

Data Example

Here we provide an example of a data frame that can be used with `sreg`. It is possible to generate a pseudo-random sample using the built-in function `sreg.rgen()`.

```
data <- sreg.rgen(n = 10, tau.vec = c(0.2, 0.8),
                 n.strata = 4, cluster = T, Nmax = 50)
head(data, n = 20)
```

	Y	S	D	G.id	Ng	x_1	x_2
1	-0.57773576	2	0	1	10	1.5597899	0.03023334
2	1.69495638	2	0	1	10	1.5597899	0.03023334
3	-0.96773507	2	0	1	10	1.5597899	0.03023334
4	0.21314929	2	0	1	10	1.5597899	0.03023334
5	-0.03443068	2	0	1	10	1.5597899	0.03023334
6	0.16122821	2	0	1	10	1.5597899	0.03023334
7	-1.17397819	2	0	1	10	1.5597899	0.03023334
8	1.14804237	2	0	1	10	1.5597899	0.03023334
9	0.08311056	2	0	1	10	1.5597899	0.03023334
10	0.36998709	2	0	1	10	1.5597899	0.03023334
11	2.02033740	4	2	2	30	0.8747419	-0.77090031
12	1.22020493	4	2	2	30	0.8747419	-0.77090031
13	1.64466086	4	2	2	30	0.8747419	-0.77090031
14	-0.32365109	4	2	2	30	0.8747419	-0.77090031
15	0.83957681	4	2	2	30	0.8747419	-0.77090031
16	0.59969969	4	2	2	30	0.8747419	-0.77090031
17	2.01177519	4	2	2	30	0.8747419	-0.77090031
18	2.21008191	4	2	2	30	0.8747419	-0.77090031
19	-2.25064316	4	2	2	30	0.8747419	-0.77090031
20	0.37962312	4	2	2	30	0.8747419	-0.77090031

Value

Printed output

The function 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 α . The example of the printed output is provided below.

Saturated Model Estimation Results under CAR
with clusters and linear adjustments

```
Observations:      2910
Clusters:          100
Number of treatments:  2
Number of strata:     2
Covariates used in linear adjustments:  Ng, x_1, x_2
---
```

Coefficients:

Tau	As.se	T-stat	P-value	CI.left(95%)	CI.right(95%)	Significance
0.74881	0.13408	5.58497	0.00000	0.48602	1.01159	***
0.56782	0.14034	4.04610	0.00005	0.29276	0.84288	***

```
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")`.

```
library(devtools)
install_github("yurytrifonov/sreg")
library(sreg)
library(dplyr)
library(haven)

# Data description
?AEJapp
# Upload the data from the package
data("AEJapp")
data <- AEJapp
# Prepare the data
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))
# Look at the input data
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

# Look at the frequency table
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
```

```

Y <- data.clean$Y
D <- data.clean$D
S <- data.clean$S

# Replicate the results from (Bugni et al, 2019)
result <- sreg::sreg(Y, S, D, HC1 = TRUE)

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, it is possible to add linear adjustments (covariates)
x_1 <- data$pillars_taken
x_2 <- data$age_months
data.clean <- data.frame(Y, D, S, x_1, x_2)
data.clean <- data.clean %>%
  mutate(D = ifelse(D == 3, 0, D))
# Look at the input data
head(data.clean)
      Y D S x_1      x_2
1 11.2 1 1  0 156.8460
2 12.4 0 3 16 186.4148
3 11.9 0 5  5 209.0513
4 13.1 0 1 21 146.2012
5 13.4 2 2  9 168.7721
6 10.7 0 1  5 146.0370
Y <- data.clean$Y
D <- data.clean$D
S <- data.clean$S
X <- data.frame(data.clean$x_1, data.clean$x_2)

# Results with linear adjustments
result <- sreg::sreg(Y, S, D, X, HC1 = TRUE)

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 -0.02862 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

```