Brief Tutorial on `art` Stata Command

In this note we briefly describe how to use the `art` Stata Command for doing inference with few clusters. The command implements the approximate randomization test (ART) with random sign changes proposed in Canay et al. (2017). For additional examples on how to use this command in two other empirical applications, see Cai et al. (2021) and its supplemental material.

Syntax

The syntax for the command is as follows:

`art` *depvar indepvar* [*if*] [*in*] [,*options*]

It is important to state that in *depvar* only pre-generated stata variables are allowed. To be specific, it does not allow for stata factor and time series operators. The command allows for the following options some of which are necessary.

- **model**(*string*)
  Predefined **necessary** stata parametric model: `regress`, `probit` and `logit`.
- **noconstant**
  Implements model without constant. If this option is omitted then a constant is included by default.
- **margins**
  Implements stata `margins` package for marginal effects. If this option is omitted then marginal effects are not used by default.
- **cluster**(*string*)
  Predefined **necessary** variable determining clusters. This options must be specified.
- **level**(#)
  Predefined level of confidence. Default is 95%.
- **signs**(#)
  Predefined number of random sign changes. Default is 999.
- **report**(*varlist*)
  Predefined list of variables amongst the covariates whose results are reported. Note that the joint test is performed only on reported variables. If this option is omitted then all variables are reported by default.
- **tolerance**(#)
  Predefined tolerance level for the termination of the test inversion algorithm. To be specific, it determines at what difference between the p-value and the chosen level of significance can we terminate the algorithm. Default is 0.025.

Example

Here we illustrate the use of the command to reproduce some of the results in Canay et al. (2017) on the Angrist and Lavy (2009) application. The *depvar* is `zakaibag`, and the *indepvar* are `treated`, `semarab`, and `semrel`.

As noted earlier, the use of command requires the pre-specification of some compulsory options, i.e. **model**(*string*) and **cluster**(*string*). In this example, we use a linear model and hence specify **model**(`regress`). The next element that the researcher must specify is a variable that decides which cluster every observation is in. It is important to note that the procedure only uses clusters where the parameter is identified. Hence, there must be enough variation to identify the parameter of interest individually in each cluster. In turn, some applications require grouping smaller clusters together ensure that this is satisfied. In this example, we generate a variable `group` that follows the grouping of smaller clusters used in Canay et al. (2017) - see accompanying do.file for details.

Furthermore, we are interest primarily in the variable `treated` and in a 90% confidence interval. We hence specify the options **report**(`treated`) and **level**(90). Note that reporting results only for some variables also reduces the computing time. Below is the command we input into stata:

```
. art zakaibag treated semarab semrel, m(regress) clust(group) l(90) r(treated)
```

The output that stata then produces is the following:

CRS Approximate Randomization Test.
Model used is regress.

Cluster var	group
Number	11
Max obs	613
Min obs	139

Number of obs	=	3821
F statistic	=	.00427653
Prob > F_stat	=	.51751752
Test statistic	=	Wald
Number of sign changes	=	999

zakaibag	Coef.	Mean	P.value	[90% Conf.	Interval]
treated	.05609727	.04931498	.51751752	-.07306608	.1624887
_cons	.17300559	.13979711	.02102102	.04227096	.24419543

The value reported under **Coef.** is the OLS estimate using the whole sample, whereas the value reported under **Mean** is the mean of the OLS estimates using only the sample under each specified cluster.

To understand which clusters are used in the procedure i.e. clusters where the parameter is identified, we use the following command to return the matrix of estimates from each cluster,

```
. matlist r(b_cluster)
```

which gives the following output below.

Here the rows denote the different clusters and the columns the different variables. The variables are in the order they were specified initially and the constant is in the final column unless it is not

	c1	c2	c3	c4
r1	-.0611255	.0350905	.0047875	.056338
r2	.028153	.2283648	.1014817	.0821918
r3	.1364965	0	.3164628	.0553871
r4	.1077586	0	.3282871	.0467129
r5	.3091821	0	.544056	.0335829
r6	-.0720738	0	0	.1641791
r7	.1872191	.0651261	.127907	.1848739
r8	.177812	.0424996	0	.2818177
r9	.2347982	.3410945	.1281316	-.1281316
r10	.0499631	.1777653	0	.1244527
r11	-.5557185	0	0	.6363636

included. As we can observe from the first column, the parameter on our variable of interest `treated` is identified in all the clusters since the estimates are non-zero everywhere. However, looking at the multiple zeros present in the other columns, we can infer that some of the other variables might not be. Note that the command produces a warning if the parameter of the reported variables are not identified in some clusters.

References

Angrist, J. and Lavy, V. (2009). The effects of high stakes high school achievement awards: Evidence from a randomized trial. *The American Economic Review*, **99** 1384–1414.

Cai, Y., Canay, I. A., Kim, D. and Shaikh, A. M. (2021). A user's guide to approximate randomization tests with a small number of clusters. *Working Paper*.

Canay, I. A., Romano, J. P. and Shaikh, A. M. (2017). Randomization tests under an approximate symmetry assumption. *Econometrica*, **85** 1013–1030.