Quant II

Lab 3: DAG review, bootstrap, clustering

Giacomo Lemoli

February 9, 2023

Today's plan

- DAG Review

Today's plan

- DAG Review
- Bootstrap

Today's plan

- DAG Review
- Bootstrap
- Clustering

Housekeeping

- Homework 1 due February 14 before class
- Email to me, cc Cyrus
- Send code and output (.Rmd or pdf + file)
- Math can be scanned and attached (or from tablet), no need to compile in LaTeX

DAG Review

- Represent causal mechanisms, i.e. causal relationship between different variables

DAG Review

- Represent causal mechanisms, i.e. causal relationship between different variables
- Non-parametric: silent about distribution, functional form, effect magnitude

DAG Review

- Represent causal mechanisms, i.e. causal relationship between different variables
- Non-parametric: silent about distribution, functional form, effect magnitude
- 3 elements: variables, arrows, missing arrows

DAG Review

- Represent causal mechanisms, i.e. causal relationship between different variables
- Non-parametric: silent about distribution, functional form, effect magnitude
- 3 elements: variables, arrows, missing arrows
- Arrow: direct causal effect

DAG Review

- Represent causal mechanisms, i.e. causal relationship between different variables
- Non-parametric: silent about distribution, functional form, effect magnitude
- 3 elements: variables, arrows, missing arrows
- Arrow: direct causal effect
- Missing arrow: no causal effect

DAG Review

- Represent causal mechanisms, i.e. causal relationship between different variables
- Non-parametric: silent about distribution, functional form, effect magnitude
- 3 elements: variables, arrows, missing arrows
- Arrow: direct causal effect
- Missing arrow: no causal effect
  - **Exclusion restrictions**: where assumptions are

DAG Review

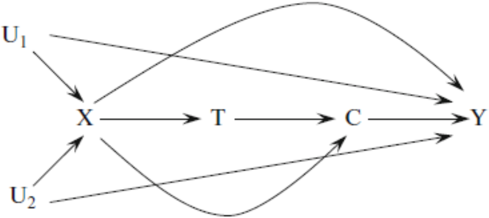
- Represent causal mechanisms, i.e. causal relationship between different variables
- Non-parametric: silent about distribution, functional form, effect magnitude
- 3 elements: variables, arrows, missing arrows
- Arrow: direct causal effect
- Missing arrow: no causal effect
  - **Exclusion restrictions**: where assumptions are
- Acyclic: cannot come back to starting point

DAG Review

Important concepts:

- A $\rightarrow$ B: B is *children* of A, A is *parent* of B
- A $\rightarrow$ C $\leftarrow$ B: C is a *collider*
- Causal path (relative to T): arrows go from T towards Y
- Non-causal path (relative to T): the others
- Backdoor path: non-causal path between T and Y
- Colliders block non-causal paths
- Conditioning on colliders open new non-causal paths

DAG Review



DAG Review

Conditioning on some variable w in a DAG is equivalent to do the following steps:

- If w is a collider, link all pairs of parents of w by drawing an undirected edge between them

DAG Review

Conditioning on some variable w in a DAG is equivalent to do the following steps:

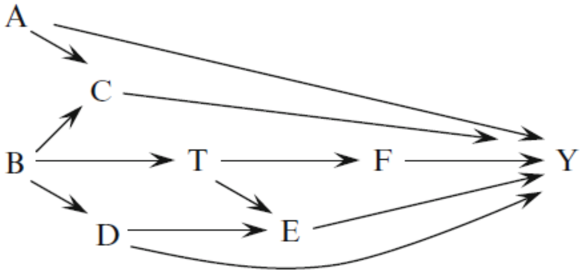
- If w is a collider, link all pairs of parents of w by drawing an undirected edge between them
- For any ancestor of w, if this ancestor is itself a collider, link all pairs of parents of this ancestor with undirected edges to connote induced dependencies

DAG Review

Conditioning on some variable w in a DAG is equivalent to do the following steps:

- If w is a collider, link all pairs of parents of w by drawing an undirected edge between them
- For any ancestor of w, if this ancestor is itself a collider, link all pairs of parents of this ancestor with undirected edges to connote induced dependencies
- Erase w from the graph and all the edges connected with w

DAG Review



Collider bias

Bootstrap

- Simulate sampling distributions (from repeated random sampling) using the available sample

Bootstrap

- Simulate sampling distributions (from repeated random sampling) using the available sample
- Parametric, Semi-parametric, Non-parametric

Bootstrap

- Simulate sampling distributions (from repeated random sampling) using the available sample
- Parametric, Semi-parametric, Non-parametric
- **Non-parametric**: based on re-sampling

Bootstrap

- Simulate sampling distributions (from repeated random sampling) using the available sample
- Parametric, Semi-parametric, Non-parametric
- **Non-parametric**: based on re-sampling
- Random re-sampling from the sample approximates random sampling from the population

Bootstrap

- Simulate sampling distributions (from repeated random sampling) using the available sample
- Parametric, Semi-parametric, Non-parametric
- **Non-parametric**: based on re-sampling
- Random re-sampling from the sample approximates random sampling from the population
- The sampling distribution of statistics from re-samples approximates the true sampling distribution

Non-parametric bootstrap

1. From our sample of size N, draw a random sample of size N with replacement
2. On this sample, compute the estimate
3. Repeat many times (e.g. 1000)
4. Obtain a distribution of estimates from the resamples (a bootstrapped sampling distribution)
5. For inference, compute moments of the bootstrapped distribution: e.g. sd, quantiles

Non-parametric bootstrap

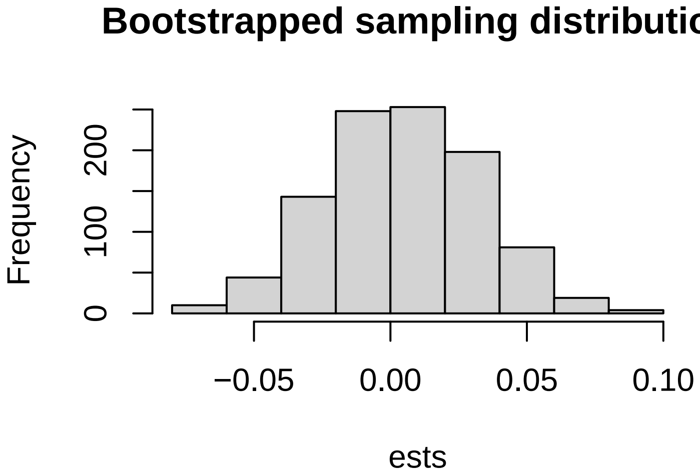
Advantages:

- No distributional assumptions
- Inference on functions of estimators

```r
set.seed(123)
# Population
pop <- rnorm(10000)

# Sample
S <- sample(pop, 500)

# Estimate Mean(x) - Median(x)
delta <- mean(S)-median(S)

# Bootstrap inference
nboot <- 1000
ests <- rep(NA, nboot)
for (i in 1:nboot){
  ids <- sample(1:length(S), length(S), replace=T)
  s <- S[ids]
  ests[i] <- mean(s)-median(s)
}
```

Non-parametric boostrap

```r
# Bootstrapped distribution
hist(ests, main="Bootstrapped sampling distribution")
```



Non-parametric bootstrap

```r
# SE
sd(ests)
```

```
## [1] 0.02840985
```

```r
# CI
delta + quantile(ests, c(0.025, 0.975))
```

```
##        2.5%       97.5%
## -0.03465047  0.07370951
```

Non-parametric bootstrap

We can also use the `boot` function

```
set.seed(123)
library(boot)
# Function for the bootstrap estimate
bootfn <- function(data, id){
  s <- data[id]
  out <- mean(s)-median(s)
  out
}

bs <- boot(S, statistic = bootfn, R=1000)
```

Non-parametric bootstrap

```r
# Output:
bs
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = S, statistic = bootfn, R = 1000)
##
##
## Bootstrap Statistics :
##       original      bias    std. error
## t1* 0.01479699 -0.01066407  0.03066007
```

```r
# Bias:
mean(bs[["t"]])-delta
```

```
## [1] -0.01066407
```

```r
# Std. error
sd(bs[["t"]])
```

```
## [1] 0.03066007
```

Bootstrap in regression

- Code manually
- lm + sandwich

```
library(sandwich);library(lmtest)
set.seed(000)
data("mtcars")
fit <- lm(mpg ~ qsec, mtcars)
coeftest(fit, vcov=vcovBS(fit, cluster=NULL, R=1000))
```

```
##
## t test of coefficients:
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.11404    8.47717 -0.6033 0.550862
## qsec         1.41212    0.48884  2.8887 0.007116 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Bootstrap in regression

- `lm + car`

```
library(car)
set.seed(000)
Boot(fit, f=coef, R=1000)
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot::boot(data = dd, statistic = boot.f, R = R, .fn = f, parallel = p_type,
##     ncpus = ncores, cl = cl2)
##
##
## Bootstrap Statistics :
##      original      bias    std. error
## t1* -5.114038 -1.07603420   9.1139840
## t2*  1.412125  0.05996816   0.5247496
```

```
# You can replace coef with more complex functions
```

Bootstrap in regression

In Stata:

```
** Regression with bootstrap SEs
set seed 000
reg mpg qsec, vce(bootstrap, reps(1000))
```

Output:

```
Linear regression                               Number of obs     =         32
                                                Replications      =      1,000
                                                Wald chi2(1)      =       9.01
                                                Prob > chi2       =     0.0027
                                                R-squared         =     0.1753
                                                Adj R-squared     =     0.1478
                                                Root MSE          =     5.5637
```

mpg	Observed Coef.	Bootstrap Std. Err.	z	P>|z|	Normal-based [95% Conf.	Interval]
qsec	1.412125	.4705516	3.00	0.003	.4898608	2.334389
_cons	-5.11404	8.153858	-0.63	0.531	-21.09531	10.86723

Clustering

- So far: we have adopted the perspective of clustered **sampling**

Clustering

- So far: we have adopted the perspective of clustered **sampling**
  - For instance, we sample villages or electoral precincts: individual behavior (e.g. voting) is correlated within cluster

Clustering

- So far: we have adopted the perspective of clustered **sampling**
  - For instance, we sample villages or electoral precincts: individual behavior (e.g. voting) is correlated within cluster
- Next: perspective of clustered **treatment assignment**

Clustering

- So far: we have adopted the perspective of clustered **sampling**
  - For instance, we sample villages or electoral precincts: individual behavior (e.g. voting) is correlated within cluster
- Next: perspective of clustered **treatment assignment**
  - For instance, we randomize a treatment at the housing block level, assignment is correlated within cluster

Clustering

- So far: we have adopted the perspective of clustered **sampling**
  - For instance, we sample villages or electoral precincts: individual behavior (e.g. voting) is correlated within cluster
- Next: perspective of clustered **treatment assignment**
  - For instance, we randomize a treatment at the housing block level, assignment is correlated within cluster
- In both cases, relative to simple random process, units are contributing less information

Clustering

- So far: we have adopted the perspective of clustered **sampling**
  - For instance, we sample villages or electoral precincts: individual behavior (e.g. voting) is correlated within cluster
- Next: perspective of clustered **treatment assignment**
  - For instance, we randomize a treatment at the housing block level, assignment is correlated within cluster
- In both cases, relative to simple random process, units are contributing less information
- Extreme case: perfect correlation

Clustering

- So far: we have adopted the perspective of clustered **sampling**
  - For instance, we sample villages or electoral precincts: individual behavior (e.g. voting) is correlated within cluster
- Next: perspective of clustered **treatment assignment**
  - For instance, we randomize a treatment at the housing block level, assignment is correlated within cluster
- In both cases, relative to simple random process, units are contributing less information
- Extreme case: perfect correlation
- If we treat units as independent, we overstate the information in the data, and underestimate the variance of the estimator

Guide to empirical practice

Source: MacKinnon, Nielsen, and Webb (JE, 2023)

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- “How do you cluster SEs?”

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
    - When in doubt, cluster at the highest possible level

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
  - When in doubt, cluster at the highest possible level
  - Choose the level which gives the highest SEs

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
  - When in doubt, cluster at the highest possible level
  - Choose the level which gives the highest SEs
- DiD setting: cluster at the unit level

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
  - When in doubt, cluster at the highest possible level
  - Choose the level which gives the highest SEs
- DiD setting: cluster at the unit level
  - Can also cluster in two dimensions if suspects of correlation along other dimensions

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
  - When in doubt, cluster at the highest possible level
  - Choose the level which gives the highest SEs
- DiD setting: cluster at the unit level
  - Can also cluster in two dimensions if suspects of correlation along other dimensions
- Treatment assigned at a higher level: cluster at that level

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
  - When in doubt, cluster at the highest possible level
  - Choose the level which gives the highest SEs
- DiD setting: cluster at the unit level
  - Can also cluster in two dimensions if suspects of correlation along other dimensions
- Treatment assigned at a higher level: cluster at that level
  - E.g. individual-level data, geographic distances at place level

Level of clustering

- In observational work we do not perfectly observe the cluster structure: needs to be established by the researcher
- "How do you cluster SEs?"
- Traditional rules of thumb:
  - When in doubt, cluster at the highest possible level
  - Choose the level which gives the highest SEs
- DiD setting: cluster at the unit level
  - Can also cluster in two dimensions if suspects of correlation along other dimensions
- Treatment assigned at a higher level: cluster at that level
  - E.g. individual-level data, geographic distances at place level
- Tests for cluster level (Ibragimov and Mueller): estimate model for each cluster and compare variation in estimates with the one under finer clusters. Only works if treatment varies within cluster

Leverage and influence

- High heterogeneity between clusters can create problems for inference

Leverage and influence

- High heterogeneity between clusters can create problems for inference
  - Results can be sensitive to the exclusion of few clusters

Leverage and influence

- High heterogeneity between clusters can create problems for inference
  - Results can be sensitive to the exclusion of few clusters
  - Influential cluster (analogous to single observations)

Leverage and influence

- High heterogeneity between clusters can create problems for inference
  - Results can be sensitive to the exclusion of few clusters
  - Influential cluster (analogous to single observations)
- Recommended to plot distribution of estimates from exclusion of each cluster sequentially

Leverage and influence

Main problems:

- Only one or few clusters are treated

Leverage and influence

Main problems:

- Only one or few clusters are treated
- Some clusters are especially large

Leverage and influence

Main problems:

- Only one or few clusters are treated
- Some clusters are especially large
  - Over-rejection increases by a lot

Leverage and influence

Main problems:

- Only one or few clusters are treated
- Some clusters are especially large
  - Over-rejection increases by a lot
  - Example: Delaware contains 50+% of US incorporations

Leverage and influence

Main problems:

- Only one or few clusters are treated
- Some clusters are especially large
  - Over-rejection increases by a lot
  - Example: Delaware contains 50+% of US incorporations
- The opposite (a few small clusters) is generally not problematic

Leverage and influence

Main problems:

- Only one or few clusters are treated
- Some clusters are especially large
  - Over-rejection increases by a lot
  - Example: Delaware contains 50+% of US incorporations
- The opposite (a few small clusters) is generally not problematic
- Only a few clusters have treated observations

Summary guide

1. List all possible clustering dimensions and levels for the data, make informed decision regarding clustering structure
2. For each possible level, report the n. of clusters and the distribution of cluster sizes
3. For key regression specifications, report info on leverage, partial leverage and influence, especially the effective number of clusters for coefficients of interest $\implies$ `summclust` in Stata
   - Is leverage collinear with cluster size?
4. In addition to the CV1 estimator (the most common), use also the jack-knife based CV3 ($\implies$ `summclust`) and WCR ($\implies$ `boottest`)
5. If treatment at cluster level and few treated/control clusters, check with other methods (e.g. randomization inference)

If you have many clusters of similar size and good treatment variation across clusters you are fine

Summclust

```
summclust mw, yvar(hours2) xvar(black female) fevar(educ
age year state) cluster(state) rho(0.5) jack
```

SUMMCLUST - MacKinnon, Nielsen, and Webb

Cluster summary statistics for mw when clustered by statefip.
There are 492827 observations within 51 statefip clusters.

Regression Output

s.e.	Coeff	Sd. Err.	t-stat	P value	CI-lower	CI-upper
CV1	-0.153891	0.062314	-2.4696	0.0170	-0.279051	-0.028730
CV3	-0.153891	0.067127	-2.2925	0.0261	-0.288720	-0.019061
CV3J	-0.153891	0.067127	-2.2925	0.0261	-0.288720	-0.019062

Cluster Variability

Statistic	Ng	Leverage	Partial L.	beta no g
min	258.0	1.015250	0.000837	-0.177645
q1	2495.0	1.136757	0.003932	-0.156898
median	7082.0	1.413468	0.009442	-0.153682
mean	9663.3	1.549020	0.019608	-0.153896
q3	13481.0	1.759903	0.020038	-0.150470
max	35995.0	3.058464	0.160196	-0.122695
coefvar	0.9	0.331260	1.481140	0.062300

Effective Number of Clusters

G*(0) is 16.18657

Summclust

Table 3
Summary statistics for cluster heterogeneity.

Clustering	G	$G^*_\beta(0)$	$\bar{N}_g$	min.	1st quart.	Median	3rd quart.	max.
Hours data: $N = 492{,}827$								
State-year	765	79.4	644	6	176	480	860	3,052
State	51	16.2	9,663	258	2,495	7,082	13,481	35,995
Year	15	6.6	32,855	28,262	28,839	30,733	40,224	40,394
Region	9	7.5	54,759	27,849	37,396	50,489	65,389	96,337
Employment and student data: $N = 1{,}531{,}360$								
State-year	765	66.0	2,002	42	524	1,413	2,426	10,794
State	51	13.1	30,027	927	7,363	22,845	37,020	144,914
Year	15	6.5	102,091	92,701	95,589	102,319	108,858	110,528
Region	9	7.0	170,151	74,172	104,703	181,767	208,099	291,955

Notes: The values of $G^*_\beta(0)$ are calculated using 28 regressors after the state dummies have been partialed out. The β subscript emphasizes the fact that they correspond to the coefficient β in (40). Because there are state fixed effects, values of $G^*_\beta(1)$ are not reported; see MacKinnon et al. (2022b).

Cluster-robust standard errors in regression

```r
library(sandwich)
# Work with MNW example dataset
d <- read.csv("min_wage_teen_hours2.csv")

# lm + sandwich
fit <- lm(hours2 ~ mw + black + female + factor(educ) + factor(age) + factor(year) + factor(statefip), d)

coeftest(fit, vcov=vcovCL(fit, cluster = ~ statefip))["mw",]
```

```
##    Estimate  Std. Error     t value    Pr(>|t|)
## -0.15389068  0.06231373 -2.46961125  0.01352633
```

Cluster-robust standard errors in regression

```r
# lfe
library(lfe)
fit <- felm(hours2 ~ mw + black + female | educ + age + year + statefip | 0 | statefip, d)
summary(fit)
```

```
##
## Call:
##    felm(formula = hours2 ~ mw + black + female | educ + age + year +      statefip | 0 | statefip, data = d)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.152  -6.934  -1.124   5.994  47.645
##
## Coefficients:
##        Estimate Cluster s.e. t value Pr(>|t|)
## mw     -0.15389      0.06231  -2.470    0.017 *
## black   0.98832      0.11714   8.437 3.54e-11 ***
## female -2.61304      0.08368 -31.227  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.887 on 492748 degrees of freedom
## Multiple R-squared(full model): 0.1927   Adjusted R-squared: 0.1926
## Multiple R-squared(proj model): 0.01753   Adjusted R-squared: 0.01738
## F-statistic(full model, *iid*): 1508 on 78 and 492748 DF, p-value: < 2.2e-16
## F-statistic(proj model): 357.1 on 3 and 50 DF, p-value: < 2.2e-16
## *** Standard errors may be too high due to more than 2 groups and exactDOF=FALSE
```

Cluster-robust standard errors in regression

```r
# fixest
library(fixest)
fit <- feols(hours2 ~ mw + black + female | educ + age + year + statefip, cluster=~statefip, data=d)
etable(fit, keep = c("mw"))
```

```
##                                   fit
## Dependent Var.:                hours2
##
## mw                 -0.1539* (0.0623)
## Fixed-Effects:     -----------------
## educ                              Yes
## age                               Yes
## year                              Yes
## statefip                          Yes
## _______________ _________________
## S.E.: Clustered      by: statefip
## Observations               492,827
## R2                         0.19270
## Within R2                  0.01753
```

Cluster-robust standard errors in regression

In Stata:

```
vce(cluster clustervar)
```

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)
- Resample clusters. For each cluster, multiply the residuals by $+1$ or -1 with equal chance

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)
- Resample clusters. For each cluster, multiply the residuals by $+1$ or -1 with equal chance
- Predict new ys with the new residuals and imposing $\beta = 0$

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)
- Resample clusters. For each cluster, multiply the residuals by $+1$ or -1 with equal chance
- Predict new ys with the new residuals and imposing $\beta = 0$
- Re-estimate the model for all parameters and find $\hat{\beta}_b$

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)
- Resample clusters. For each cluster, multiply the residuals by $+1$ or -1 with equal chance
- Predict new ys with the new residuals and imposing $\beta = 0$
- Re-estimate the model for all parameters and find $\hat{\beta}_b$
- Compute t-statistic t_b from the new estimates

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)
- Resample clusters. For each cluster, multiply the residuals by $+1$ or -1 with equal chance
- Predict new ys with the new residuals and imposing $\beta = 0$
- Re-estimate the model for all parameters and find $\hat{\beta}_b$
- Compute t-statistic t_b from the new estimates
- Repeat many times

Cluster bootstrap

- Most commonly used: Wild Cluster Bootstrap (Restricted)
- Run the regression, estimate $\hat{\beta}$ and t
- To test the null hypothesis: run the regression under the null hypothesis (i.e. setting $\beta = 0$)
- Resample clusters. For each cluster, multiply the residuals by $+1$ or -1 with equal chance
- Predict new ys with the new residuals and imposing $\beta = 0$
- Re-estimate the model for all parameters and find $\hat{\beta}_b$
- Compute t-statistic t_b from the new estimates
- Repeat many times
- Compute bootstrap p-values by counting the share of simulated t_b to the left/right of the observed one t

Cluster bootstrap

- In Stata: `boottest`
- Computes bootstrap p-value and confidence interval, but not the standard error

```
. boottest mw, cluster(styear) noci reps(999) seed(123)

Overriding estimator's cluster/robust settings with cluster(styear)

Wild bootstrap-t, null imposed, 999 replications, Wald test, clustering by styear, bootstrap clustering by
>  styear, Rademacher weights:
  mw

                        t(764) =     -3.3823
                    Prob>|t| =      0.0020

.
```

Cluster bootstrap

In R:

- Option `cluster` in `sandwich::vcovBS()`

Cluster bootstrap

In R:

- Option `cluster` in `sandwich::vcovBS()`
- The package `fwildclusterboot` is a translation of Stata's `boottest` (same options)

Cluster bootstrap

In R:

- Option `cluster` in `sandwich::vcovBS()`
- The package `fwildclusterboot` is a translation of Stata's `boottest` (same options)
  - Function `boottest` works with objects of class `lm`, `felm`, `fixest`

Cluster bootstrap

In R:

- Option `cluster` in `sandwich::vcovBS()`
- The package `fwildclusterboot` is a translation of Stata's `boottest` (same options)
  - Function `boottest` works with objects of class `lm`, `felm`, `fixest`
- Another option is the package `multiwaycov` and the function `cluster.boot` which can be used for post-estimation SE calculation (e.g. in `coeftest` or `stargazer`)