Beyond the basic of OLS

Mauricio Romero

Beyond the basic of OLS

A few things that don't get enough attention

Error structure

Statistical power

Beyond the basic of OLS

A few things that don't get enough attention

Error structure

Statistical power

Beyond the basic of OLS

A few things that don't get enough attention

- How to interpret coefficients/regression table
- Leverage
- The perils of p-hacking
- What if your outcome is a dummy?
- Ordinal/Categorical data

Error structure

- Heteroskedasticity
- Cluster standard errors

Statistical power

- Randomizing at the Unit of Analysis
- Cluster Randomized Experiments

Interpreting a regression output

- Great, you ran a regression
- Let's assume it has a causal interpretation (big if)
- How do you interpret the results?

Warning!

- Be careful not to confuse percent with percentage point
- A change from 10% to 13% is a rise of 3 (13-10) percentage points
- This is not equal to a 3% change; rather, it's a $30\% = 100\frac{13-10}{10}$ increase

Level-level Regression

- If you have a level-level regression

$$y_i = \beta_0 + \beta_1 x_i + u_i$$

- If you increase x by one, we expect y to change by β_1

An example

- A regression of wages on: Age (in years), race (black=1) and IQ percentile (0-100)
- For every year, we expect *wages* to change by $\widehat{\beta_{age}}$ USD
- On average, we expect *wages* to higher/lower for blacks by $\widehat{\beta_{female}}$ USD than for non-blacks
- For every **percentage point** increase in IQ, we expect *wages* to change by $\widehat{\beta_{IQ}}$ USD

Simulations!

```
library(wooldridge)
library(stargazer)
data("wage2")
wage2$IQ_Percentile=quantile(wage2$IQ, seq(0, 1, 0.1))

levlev=lm(wage ~ IQ_Percentile + age + black, data = wage2)
summary(levlev)
stargazer(levlev, title="Level-Level", align=TRUE,
          type="latex", omit.table.layout="=!a",
          out="Lectures/tables/levlev.tex",
          covariate.labels=c("IQ (percentile)","Age","Black(=1)"),
          digits=2,digits.extra=1,no.space=T,colnames=F,
          dep.var.caption="",dep.var.labels="Wage",
          column.sep.width="0pt",header=F,
          omit.stat=c("adj.rsq","rsq","f","ser"))
```

```
Call:
lm(formula = wage ~ IQ_Percentile + age + black, data = wage2)

Residuals:
    Min      1Q  Median      3Q     Max 
-803.60 -271.87  -62.62  212.27 2174.38 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)    332.4888   150.2711   2.213   0.0272 *  
IQ_Percentile    0.1320     0.5615   0.235   0.8141    
age             19.4666     4.1241   4.720 2.72e-06 ***
black         -248.0806    38.2995  -6.477 1.51e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 391.2 on 931 degrees of freedom
Multiple R-squared:  0.06681,	Adjusted R-squared:  0.0638 
F-statistic: 22.22 on 3 and 931 DF,  p-value: 6.71e-14
```

	Wage
IQ (percentile)	0.13
	(0.56)
Age	19.47***
	(4.12)
Black(=1)	−248.08***
	(38.30)
Constant	332.49**
	(150.27)
Observations	935
Note:	*p<0.1; **p<0.05; ***p<0.01

Log-level Regression

- If you have a log-level regression

$$\ln(y_i) = \beta_0 + \beta_1 x_i + u_i$$

- If you increase x by one, we expect y to change by $100\beta_1$ percent
    - Technically, $\%\Delta y = 100(e^{\beta_1} - 1)$
    - But $\%\Delta y = 100(e^{\beta_1} - 1) \approx 100\beta_1$ for values $-0.1 < \beta_1 < 0.1$
- You can only include observations for which $y_i > 0$
- Only do it if this doesn't introduce bias into your sample
    - In general, only do it if $y_i > 0$ for almost all i
    - **Adding 1 or 0.1, or 100 is not a valid fix**

An example

- A regression of ln(*wages*) on: Age (in years), race (black=1) and IQ percentile (0-100)

- For every year, we expect *wages* to change by $100\widehat{\beta_{age}}$ percent

- On average, we expect *wages* to be higher/lower for blacks by $100\widehat{\beta_{female}}$ percent than for non-blacks

- For every **percentage point** increase in IQ, we expect *wages* to change by $100\widehat{\beta_{IQ}}$ percent

Simulations!

```
loglev=lm(log(wage) ~ IQ_Percentile + age + black, data = wage2)

summary(loglev)
stargazer(loglev, title="Log-Level", align=TRUE,
          type="latex", omit.table.layout="=!a",
          out="Lectures/tables/loglev.tex",
          covariate.labels=c("IQ (percentile)","Age","Black(=1)"),
          digits=2,digits.extra=1,no.space=T,colnames=F,
          dep.var.caption="",dep.var.labels="ln(Wage)",
          column.sep.width="0pt",header=F,
          omit.stat=c("adj.rsq","rsq","f","ser"))
```

```
Call:
lm(formula = lwage ~ IQ_Percentile + age + black, data = wage2)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.98581 -0.25765  0.01094  0.27996  1.30084 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    6.128e+00  1.556e-01  39.378  < 2e-16 ***
IQ_Percentile -1.153e-05  5.814e-04  -0.020    0.984    
age            2.083e-02  4.271e-03   4.878 1.26e-06 ***
black         -2.852e-01  3.966e-02  -7.191 1.33e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4052 on 931 degrees of freedom
Multiple R-squared:  0.07746,	Adjusted R-squared:  0.07449 
F-statistic: 26.06 on 3 and 931 DF,  p-value: 3.438e-16
```

Log-Level

	ln(Wage)
IQ (percentile)	-0.000
	(0.001)
Age	0.02^{***}
	(0.004)
Black(=1)	-0.29^{***}
	(0.04)
Constant	6.13^{***}
	(0.16)
Observations	935
Note:	*p<0.1; **p<0.05; ***p<0.01

Level-log Regression

- If you have a log-level regression

$$y_i = \beta_0 + \beta_1 \ln(x_i) + u_i$$

- If you increase x by one percent (**NOT BY ONE PERCENTAGE POINT!**), we expect y to change by $\frac{\beta_1}{100}$ units of y
- You can only include observations for which $x_i > 0$
- Only do it if this doesn't introduce bias into your sample
  - In general, only do it if $x_i > 0$ for almost all i
  - **Adding 1 or 0.1, or 100 is not a valid fix**

An example

- A regression of *wages* on: ln(Age), race (black=1) and ln(IQ) (IQ is the percentile)

- For an increase in 1 percent in age, we expect *wages* to change by $\frac{\widehat{\beta_{age}}}{100}$ USD

- On average, we expect *wages* to be higher/lower for blacks by $\frac{\widehat{\beta_{female}}}{100}$ USD than for non-blacks

- For an increase in 1 percent in the IQ percentile (that is, a percent change in percentage points), we expect *wages* to change by $\frac{\widehat{\beta_{IQ}}}{100}$ USD

Simulations!

```
levlog=lm(wage ~ log(IQ_Percentile) + log(age) + black, data = wage2)

summary(levlog)
stargazer(levlog, title="Level-Log", align=TRUE,
          type="latex", omit.table.layout="=!a",
          out="Lectures/tables/levlog.tex",
          covariate.labels=c("ln(IQ (percentile))","ln(Age)","Black(=1)"),
          digits=2,digits.extra=1,no.space=T,colnames=F,
          dep.var.caption="",dep.var.labels="Wage",
          column.sep.width="0pt",header=F,
          omit.stat=c("adj.rsq","rsq","f","ser"))
```

```
Call:
lm(formula = wage ~ log(IQ_Percentile) + log(age) + black, data = wage2)

Residuals:
    Min      1Q  Median      3Q     Max 
-803.07 -271.50  -60.65  210.88 2180.13 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)        -1340.20     536.08  -2.500   0.0126 *  
log(IQ_Percentile)    13.98      49.60   0.282   0.7782    
log(age)             648.41     136.38   4.754 2.30e-06 ***
black               -247.97      38.29  -6.477 1.51e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 391.2 on 931 degrees of freedom
Multiple R-squared:  0.06713,	Adjusted R-squared:  0.06412 
F-statistic: 22.33 on 3 and 931 DF,  p-value: 5.731e-14
```

Level-Log

	Wage
ln(IQ (percentile))	13.98
	(49.60)
ln(Age)	648.41***
	(136.38)
Black(=1)	−247.97***
	(38.29)
Constant	−1,340.20**
	(536.08)
Observations	935
Note:	*p<0.1; **p<0.05; ***p<0.01

log-log Regression

- If you have a log-level regression

$$\ln(y_i) = \beta_0 + \beta_1 \ln(x_i) + u_i$$

- If you increase x by one percent (**NOT BY ONE PERCENTAGE POINT!**), we expect y to change by β_1 percent
- You can only include observations for which $x_i > 0$ and $y_i > 0$
- Only do it if this doesn't introduce bias into your sample
    - In general, only do it if $x_i > 0$ and $y_i > 0$ for almost all i
    - **Adding 1 or 0.1, or 100 is not a valid fix**

An example

- A regression of $ln(wages)$ on: $\ln(Age)$, race (black=1) and $\ln(IQ)$ (IQ is the percentile)

- For an increase in one percent in age, we expect *wages* to change by $\widehat{\beta_{age}}$ percent

- On average, we expect *wages* to be higher/lower for blacks by $\widehat{\beta_{female}}$ percent than for non-blacks

- For an increase in one percent in the IQ percentile (that is, a percent change in percentage points), we expect *wages* to change by $\widehat{\beta_{IQ}}$ percent

Simulations!

```r
loglog=lm(log(wage) ~ log(IQ_Percentile) + log(age) + black, data = wage2)

summary(loglog)
stargazer(loglog, title="Log-Level", align=TRUE,
          type="latex", omit.table.layout="=!a",
          out="Lectures/tables/loglog.tex",
          covariate.labels=c("ln(IQ (percentile))","ln(Age)","Black(=1)"),
          digits=2,digits.extra=1,no.space=T,colnames=F,
          dep.var.caption="",dep.var.labels="ln(Wage)",
          column.sep.width="0pt",header=F,
          omit.stat=c("adj.rsq","rsq","f","ser"))
```

```
Call:
lm(formula = log(wage) ~ log(IQ_Percentile) + log(age) + black,
    data = wage2)

Residuals:
     Min       1Q   Median       3Q      Max
-1.98259 -0.25865  0.01121  0.28098  1.30397

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         4.3985406  0.5551443   7.923 6.58e-15 ***
log(IQ_Percentile) -0.0009437  0.0513599  -0.018    0.985
log(age)            0.6929047  0.1412305   4.906 1.10e-06 ***
black              -0.2850449  0.0396476  -7.189 1.33e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4051 on 931 degrees of freedom
Multiple R-squared:  0.07774,   Adjusted R-squared:  0.07477
F-statistic: 26.16 on 3 and 931 DF,  p-value: 2.994e-16
```

Log-Level

	ln(Wage)
ln(IQ (percentile))	−0.001
	(0.05)
ln(Age)	0.69***
	(0.14)
Black(=1)	−0.29***
	(0.04)
Constant	4.40***
	(0.56)
Observations	935
Note:	*p<0.1; **p<0.05; ***p<0.01

Beyond the basic of OLS

Leverage

- Remember that

$$\widehat{\beta} = \frac{cov(x, y}{v(x)} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

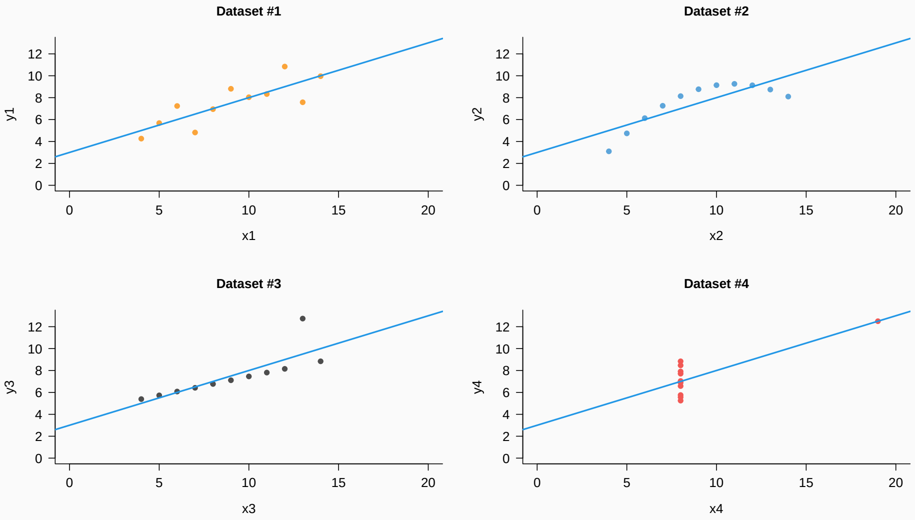
- We can rewrite as:

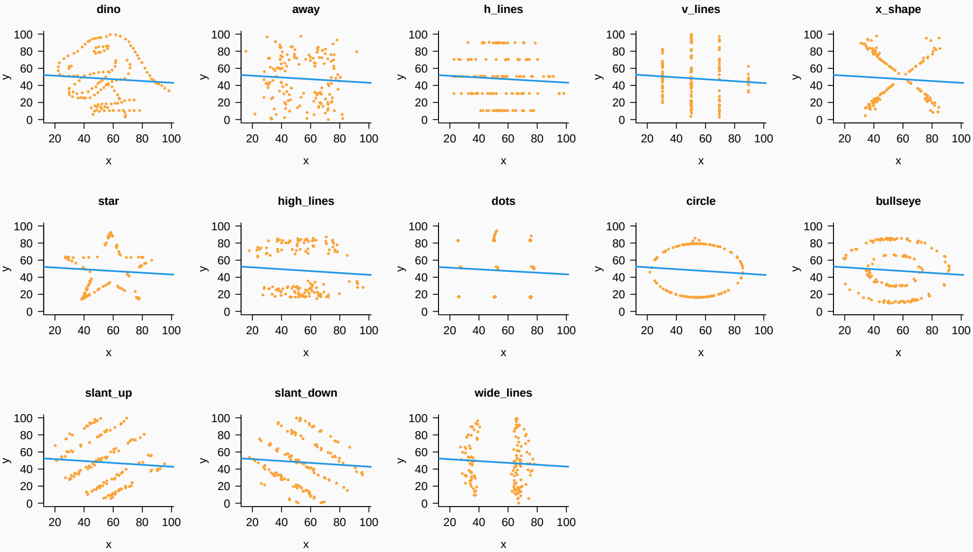
$$\widehat{\beta} = \frac{(x_1 - \overline{x})(y_1 - \overline{y}) + \sum_{i=2}^{n}(x_i - \overline{x})(y_i - \overline{y})}{(x_1 - \overline{x})^2 + \sum_{i=2}^{n}(x_i - \overline{x})^2}$$

- If $x_i = \overline{x}$, then $\widehat{\beta} = \frac{\sum_{i=2}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=2}^{n}(x_i - \overline{x})^2}$
- The first observation doesn't affect the outcome

Leverage: Big Picture

- That was an extreme case ($x_i = \overline{x}$) but generally speaking:
- The farther an observation is from $\overline{x}$, the more it affects the OLS estimator
- This is called "leverage"
- See a recent discussion on Twitter of economist arguing about this `https://twitter.com/arindube/status/1279919438419165184?s=20`


Dataset #1
Dataset #2
Dataset #3
Dataset #4
y1
x1
y2
x2
y3
x3
y4
x4
0
2
4
6
8
10
12
0
5
10
15
20


dino
away
h_lines
v_lines
x_shape
star
high_lines
dots
circle
bullseye
slant_up
slant_down
wide_lines
x
y
0
20
40
60
80
100

The perils of p-hacking

https://xkcd.com/882/

What if your outcome is a dummy?

- All we have talked about still holds
- Logit/Probit have very strong assumptions (the shape of the error term)
- Regression is more robust in general

An example

- A regression of *employment* (=1 for employed, =0 for unemployed)) on: *Age*, gender (female=1) and *IQ* (percentile)
- For an increase in 1 year of age, we expect the probability of employment to change by $100\widehat{\beta_{age}}$ **percentage points**
- On average, we expect the probability of employment to be higher for females by $100\widehat{\beta_{female}}$ **percentage points** than for males
- For an increase in 1 percentage point in IQ , we expect the probability of employment to change by $100\widehat{\beta_{IQ}}$ **percentage points**

A few things that don't get enough attention

- How to interpret coefficients/regression table
- Leverage
- The perils of p-hacking
- What if your outcome is a dummy?
- Ordinal/Categorical data

Error structure

- Heteroskedasticity
- Cluster standard errors

Statistical power

- Randomizing at the Unit of Analysis
- Cluster Randomized Experiments

What if your outcome is Ordinal/Categorical?

- Then you cannot do OLS
- OLS assumes a metric
  - Distance between $Y = 1$ and $Y = 2$ is the same as $Y = 2 and Y = 3$
- Unclear in what units β is

What can you do?

- Transform your data to binary

- Do order probit/logit

Beyond the basic of OLS

A few things that don't get enough attention

Error structure

Statistical power

Beyond the basic of OLS

A few things that don't get enough attention

Error structure

Statistical power

Variance of OLS estimators

The correct variance estimation procedure is given by the structure of the data

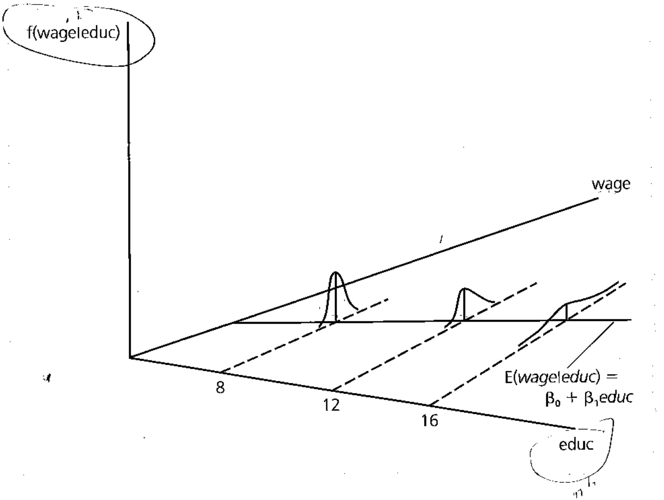
- It is very unlikely that all observations in a dataset are unrelated, but drawn from identical distributions (**homoskedasticity**)
- For instance, the variance of income is often greater in families belonging to top deciles than among poorer families (**heteroskedasticity**)
- Some phenomena do not affect observations individually, but they do affect groups of observations uniformly within each group (**clustered data**)

Beyond the basic of OLS

OLS inference is generally faulty in the presence of heteroskedasticity

Figure 2.9

Var (*wage*|*educ*) increasing with *educ*.



Heteroskedasticity

- Assume

$$Var(u_i|x_i) = \sigma_i^2$$

- Fortunately, OLS is still useful ($\widehat{\beta}$ still consistent/unbiased)
- Note that errors are still independent from each other
- The variance of our estimator, $\widehat{\beta}_1$ equals:

$$Var(\widehat{\beta}_1) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2\sigma_i^2}{\sum_{i=1}^{n}(x_i - \overline{x})^2} = (X'X)^{-1}X'V(u_i|X)X(X'X)^{-1}$$

- When $\sigma_i^2 = \sigma^2$ for all i, this formula reduces to the usual form, $\frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \overline{x})^2} = \sigma^2(X'X)^{-1}$

Robust standard errors

- A valid estimator of Var($\widehat{\beta}_1$) for heteroskedasticity of any form (including homoskedasticity) is

$$Var(\widehat{\beta}_1) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2 \widehat{u}_i^{\,2}}{\sum_{i=1}^{n}(x_i - \overline{x})^2} = (X'X)^{-1}X'(\sum_{i=1}^{n} x_i x_i' \widehat{u}_i^{\,2})X(X'X)^{-1}$$

which is easily computed from the data after the OLS regression

- As a rule, you should always use "robust standard errors"

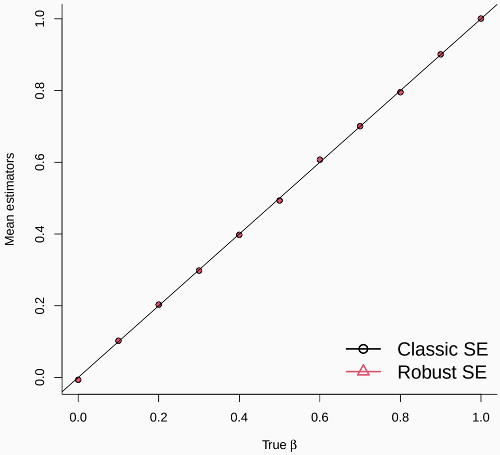
Simulations!

```
library(sandwich)
alpha=0 #intercept
Reps=1000 #how many simulations?
Nobs=100 #number of obs
SequenceBetas=seq(0,1,0.1) #lets do different betas
FractionSignificant=NULL #fraction significant 5\% level
FractionSignificant_robust=NULL #fraction significant 5\% level when using robust
betaVector=NULL #mean estimator
betaVector_robust=NULL #mean estimator robust
```

Simulations!

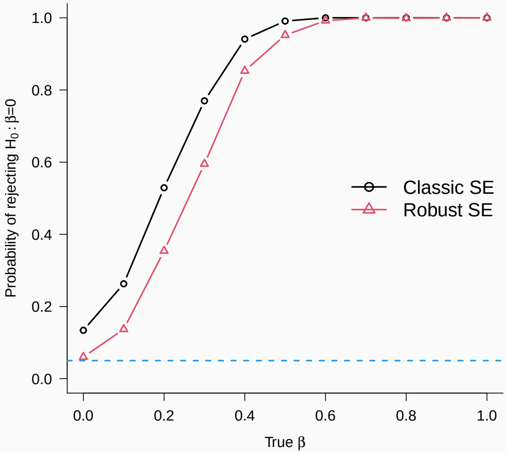
```
for(beta in SequenceBetas){
  #save the outcomes from the simulations
  beta_estimate=rep(NA,Reps)
  beta_pvalue=rep(NA,Reps)
  beta_estimate_robust=rep(NA,Reps)
  beta_pvalue_robust=rep(NA,Reps)
  X=as.matrix(runif(Nobs,-5,5)) #generate some x data
  for(r in 1:Reps){
  #use the DGP to generate outcome data with heteroskedasticity
    Y=alpha+beta*X+rnorm(Nobs,sd=1)*X
    OLS=lm(Y~X) #estimate OLS
    ResultsOLS=summary(OLS)$coef #save results from OLS table
    beta_estimate[r]=ResultsOLS[2,1]
    beta_pvalue[r]=ResultsOLS[2,4]
    #Results from robust OLS: HC1 yields same results as stata
    ResultsRobust=coeftest(OLS, vcov = vcovHC(OLS, type = "HC1"))
    beta_estimate_robust[r]=ResultsRobust[2,1]
    beta_pvalue_robust[r]=ResultsRobust[2,4]
  }
  #Save the results for the given value of beta
  FractionSignificant=c(FractionSignificant,mean(beta_pvalue<0.05))
  FractionSignificant_robust=c(FractionSignificant_robust,mean(beta_pvalue_robust<0.05))
  betaVector=c(betaVector,mean(beta_estimate))
  betaVector_robust=c(betaVector_robust,mean(beta_estimate_robust))
}
```

No bias)



Power Curve – Incorrect type-I error from classic OLS, correct from robust SE)

Proportion of times we reject the null at $\alpha = 0.05$



Clustered data

- But what if errors are not independent?
- Maybe observations between units in a group are related to each other
    - Imagine you randomly assing a treatment at the school level (e.g., extra resources)
    - The **unobservables** of kids belonging to the same school are correlated (e.g., teacher quality, recess routines)
    - The **unobservables** of kids in different school are unlikely to be correlated
- Then independence of errors across observations is violated
- But maybe independence holds across schools, just not within schools

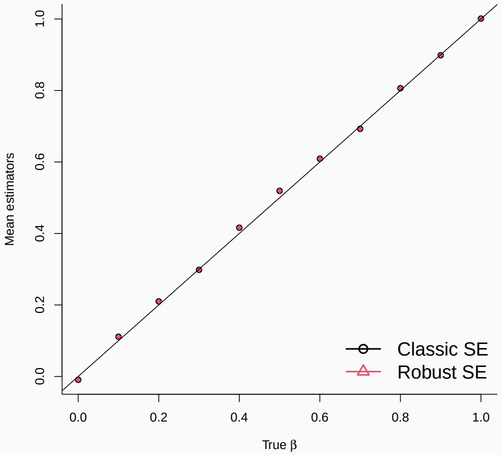
Simulations!

```
Classes=50 #number of classes or schools
StudentsPerClass=10 #number of obs per schools
Reps=1000 #repetitions
SequenceBetas=seq(0,1,0.1) #try different betas (treatment effects)
alpha=0 #intercept
FractionSignificant=NULL #fraction significant 5\% level
FractionSignificant_robust=NULL #fraction significant 5\% level when using robust
betaVector=NULL #mean estimator
betaVector_robust=NULL #mean estimator robust
```

Simulations!

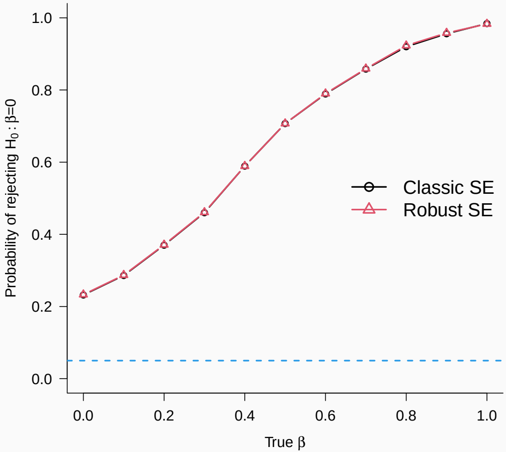
```r
for(beta in SequenceBetas){
  #save the outcomes from the simulations
  beta_estimate=rep(NA, Reps)
  beta_pvalue=rep(NA, Reps)
  beta_estimate_robust=rep(NA, Reps)
  beta_pvalue_robust=rep(NA, Reps)
  X=as.matrix(runif(StudentsPerClass*Classes,-5,5)) #generate some x data
  for(r in 1:Reps){
    Schoks_Cluster=rep(rnorm(Classes),each=StudentsPerClass)
    Schoks_Individual=rnorm(StudentsPerClass*Classes,sd=1)
    Y=alpha+beta*X+Schoks_Cluster+Schoks_Individual
    OLS=lm(Y~X) #estimate OLS
    ResultsOLS=summary(OLS)$coef
    beta_estimate[r]=ResultsOLS[2,1]
    beta_pvalue[r]=ResultsOLS[2,4]
    #Results from robust OLS: HC1 yields same results as stata
    ResultsRobust=coeftest(OLS, vcov = vcovHC(OLS, type = "HC1"))
    beta_estimate_robust[r]=ResultsRobust[2,1]
    beta_pvalue_robust[r]=ResultsRobust[2,4]
  }
  #Save the results for the given value of beta
  FractionSignificant=c(FractionSignificant,mean(beta_pvalue<0.05))
  FractionSignificant_robust=c(FractionSignificant_robust,mean(beta_pvalue_robust<0.05))
  betaVector=c(betaVector,mean(beta_estimate))
  betaVector_robust=c(betaVector_robust,mean(beta_estimate_robust))
}
```

No bias



Power Curve – Incorrect type-I error from classic OLS and from robust SE)

Proportion of times we reject the null at $\alpha = 0.05$



Cluster robust standard errors

- Both classic OLS and robust SE overreject (i.e., they reject the null when its true more times than we thought at a given level)
- We kneed to allow for arbitrary correlation within group
- Instead of summing over each individual, we first sum over groups
- I'll use matrix notation as it's easier for me to explain by stacking the data

Clustered data

- Let's stack the observations by cluster

$$y_g = x_g \beta + u_g$$

- The OLS estimator of β is:

$$\widehat{\beta} = [X'X]^{-1}X'y$$

- The variance is given by:

$$Var(\beta) = E[[X'X]^{-1}X'\Omega X[X'X]^{-1}]$$

Clustered data

With this in mind, we can now write the variance-covariance matrix for clustered data

$$Var(\widehat{\beta}) = [X'X]^{-1}[\sum_{i=1}^{G} x_g' \widehat{u}_g \widehat{u}_g' x_g][X'X]^{-1}$$

where $\widehat{u}_g$ are residuals from the stacked regression

- In STATA: `vce(cluster clustervar)`
- In R use lfe package

Simulations!

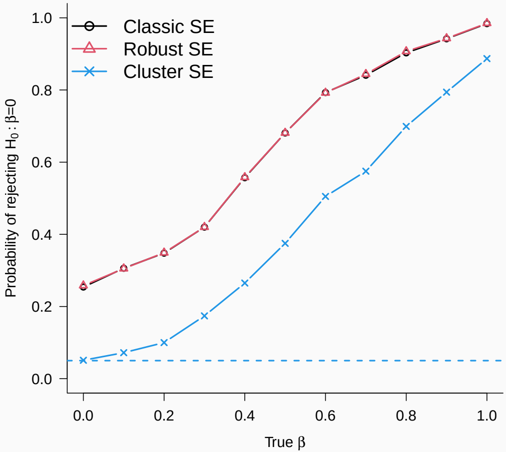
```
library(lfe)
Classes=50 #number of classes or schools
StudentsPerClass=5 #number of obs per schools
Reps=1000 #repetitions
SequenceBetas=seq(0,1,0.1) #try different betas (treatment effects)
alpha=0 #intercept
FractionSignificant=NULL #fraction significant 5\% level
FractionSignificant_robust=NULL #fraction significant 5\% level when using robust
FractionSignificant_cluster=NULL #fraction significant 5\% level when using cluster
```

Simulations!

```
for(beta in SequenceBetas){
  #save the outcomes from the simulations
  beta_pvalue=rep(NA,Reps)
  beta_pvalue_robust=rep(NA,Reps)
  beta_pvalue_cluster=rep(NA,Reps)
  ClusterIndicator=rep(1:Classes,each=StudentsPerClass)
  TreatmentClassLevel=sample(c(0,1),Classes,replace=T)
  TreatmentIndividual=rep(TreatmentClassLevel,each=StudentsPerClass)
  for(r in 1:Reps){
    Schoks_Cluster=rep(rnorm(Classes),each=StudentsPerClass)
    Schoks_Individual=rnorm(StudentsPerClass*Classes,sd=1)
    Y=alpha+beta*TreatmentIndividual+Schoks_Cluster+Schoks_Individual
    OLS=felm(Y~TreatmentIndividual | 0 | 0 | ClusterIndicator) #estimate OLS
    beta_pvalue[r]=OLS$pval[2]
    #Results from robust SE
    beta_pvalue_robust[r]=OLS$rpval[2]
    #Results from cluster SE
    beta_pvalue_cluster[r]=OLS$cpval[2]
  }
  #Save the results for the given value of beta
  FractionSignificant=c(FractionSignificant,mean(beta_pvalue<0.05))
  FractionSignificant_robust=c(FractionSignificant_robust,mean(beta_pvalue_robust<0.05))
  FractionSignificant_cluster=c(FractionSignificant_cluster,mean(beta_pvalue_cluster<0.05))
}
```

Power Curve)

Proportion of times we reject the null at $\alpha = 0.05$



The importance of knowing your data

- In real world you should never go with the “independent and identically distributed” (i.e., homoskedasticity) case. Life is not that simple.
- You need to know your data in order to choose the correct error structure and then infer the required SE calculation
- At a minimum, use robust standard errors
- If you have aggregate variables, like class size, you need to consinder clustering at that level

When to cluster?

- Case 1: If sampling follows a two stage process where in the first stage, a subset of clusters were sampled randomly from a population of clusters, and in the second stage, units were sampled randomly from the sampled clusters

- Case 2: When clusters of units, rather than units, are assigned to a treatment

When to cluster?

- The results on cluster SE

$$Var(\widehat{\beta}) = [X'X]^{-1}[\sum_{i=1}^{G} x_g' \widehat{u}_g \widehat{u}_g' x_g][X'X]^{-1}$$

  relies on "asymptotic results" based on the number of clusters (G) — not on the total sample size N
- Can only use cluster SE if number of clusters is "large" (usually over $\sim 40 - 50$)
- If number of clusters is small consider:
  - Collapsing the data at the "cluster" level
  - Wild bootstrap
  - Randomization inference (if you have an experiment)

When to cluster?

- Two good reads on clustering:
  - Cameron, A.C. and Miller, D.L., 2015. A practitioner's guide to cluster-robust inference. Journal of human resources. `http://jhr.uwpress.org/content/50/2/317.refs`
  - Abadie, A., Athey, S., Imbens, G.W. and Wooldridge, J., 2017. When should you adjust standard errors for clustering? (No. w24003). National Bureau of Economic Research. `https://www.nber.org/papers/w24003`

Beyond the basic of OLS

A few things that don't get enough attention

Error structure

Statistical power

A few things that don't get enough attention

Error structure

Statistical power

Introduction

- In a simple experiment the average treatment effect is the difference in sample means between the treatment and the control group
- This is the OLS coefficient of β in the regression

$$Y_i = \alpha + \beta T_i + \varepsilon_i$$

Regression analysis of OLS

$$X'X = pN \begin{pmatrix} \frac{1}{p} & 1 \\ 1 & 1 \end{pmatrix}$$

$$\left(X'X\right)^{-1} = \frac{1}{N(1-p)} \begin{pmatrix} 1 & -1 \\ -1 & \frac{1}{p} \end{pmatrix}$$

And

$$V \begin{pmatrix} \widehat{\alpha} \\ \widehat{\beta} \end{pmatrix} = \sigma^2 (X'X)^{-1}$$

Statistical power

How many observations are enough?

Statistical power

How many observations are enough?

Definition
The power of the design is the probability that, for a given effect size and a given statistical significance level, we will be able to reject the hypothesis of zero effect

Statistical power

- **Is the unit of treatment the same as the unit of analysis?** Or, is the treatment to be administered to a 'cluster' of units?

Statistical power

- **Is the unit of treatment the same as the unit of analysis?** Or, is the treatment to be administered to a 'cluster' of units?
- Examples of individual randomizations:
  - Individuals who are given mobile phones to induce them to use an m-banking platform
  - Farmers individually provided with improved agricultural inputs
  - Students admitted to an elite school by a lottery process

Beyond the basic of OLS

Randomizing at the Unit of Analysis

- The estimate of treatment effect is $\widehat{\beta}$ in the regression

$$Y_i = \alpha + \beta T_i + \varepsilon_i$$

- The mean of $\widehat{\beta}$ is β (the true effect)
- The variance of $\widehat{\beta}$ is $V(\widehat{\beta}) = \frac{\sigma^2}{p(1-p)N}$
- σ^2 is the variance of the outcome (Y_i)
- p is the proportion of treated units
- N is the number of observations

Randomizing at the Unit of Analysis

- We are generally interested in testing the null hypothesis (H_0) that the effect of the program is equal to zero against the alternative that it is not
- The **significance level**, or size, of a test represents the probability of a type I error, i.e., the probability we reject the hypothesis when it is in fact true
- The **power of the test** the probability that we reject H_0 when it is in fact false

Randomizing at the Unit of Analysis

- We are generally interested in testing the null hypothesis (H_0) that the effect of the program is equal to zero against the alternative that it is not
- The **significance level**, or size, of a test represents the probability of a type I error, i.e., the probability we reject the hypothesis when it is in fact true
- The **power of the test** the probability that we reject H_0 when it is in fact false

We will constantly use the fact that:

$$\widehat{\beta} \sim N\left(\beta, \frac{\sigma^2}{p(1-p)N}\right)$$

Randomizing at the Unit of Analysis

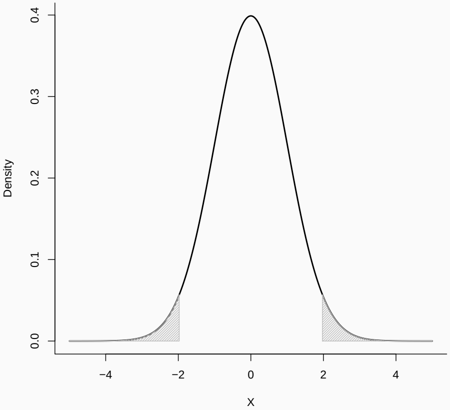
- We are generally interested in testing the null hypothesis (H_0) that the effect of the program is equal to zero against the alternative that it is not
- The **significance level**, or size, of a test represents the probability of a type I error, i.e., the probability we reject the hypothesis when it is in fact true
- The **power of the test** the probability that we reject H_0 when it is in fact false

We will constantly use the fact that:

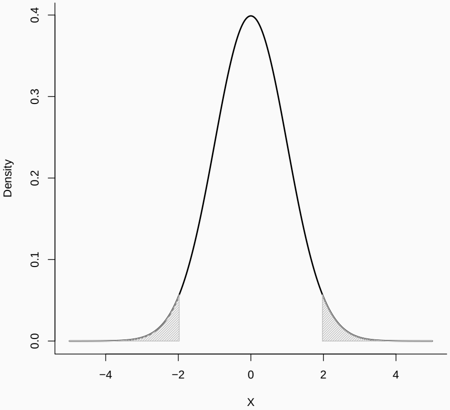
$$\widehat{\beta} \sim N\left(\beta, \frac{\sigma^2}{p(1-p)N}\right)$$

We often normalize the outcome and present results in terms of SD (so $\sigma^2 = 1$).

Significance level - Assume null is true (no effect)



Significance level - Assume null is true (no effect)

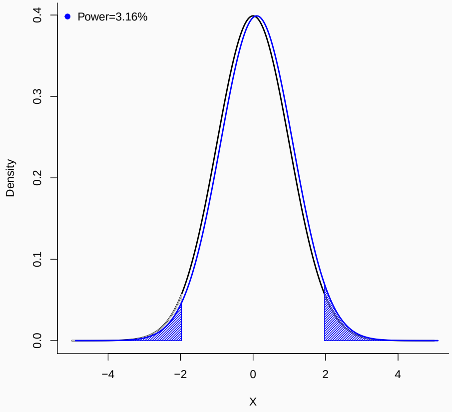


Gray area is the probability we reject the null when it is true

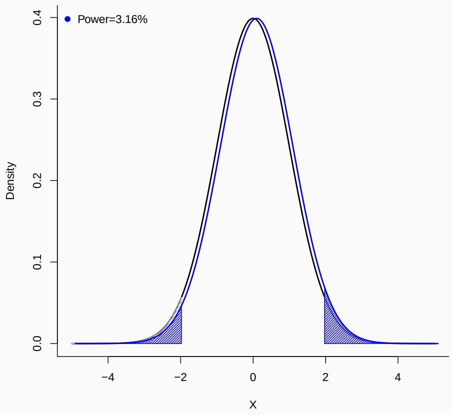
Power when the effect is β_1

For a true effect size β this is the fraction of the area under this curve that falls to the right of the critical value $t_{\frac{\alpha}{2}}$

Power when the effect is $\beta = 0.1$

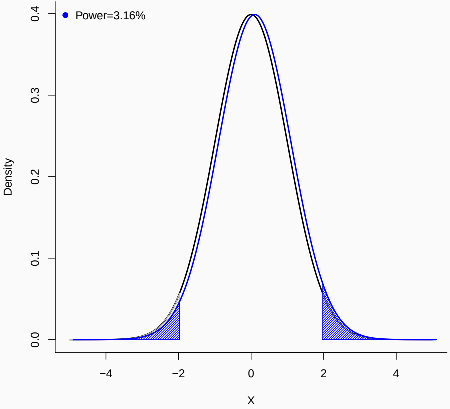


Power when the effect is $\beta = 0.1$

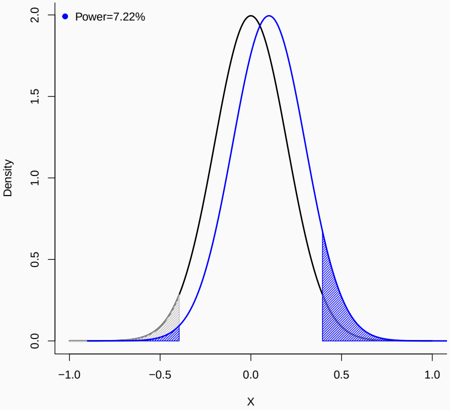


Blue area is the probability we reject the null when β is 0.1

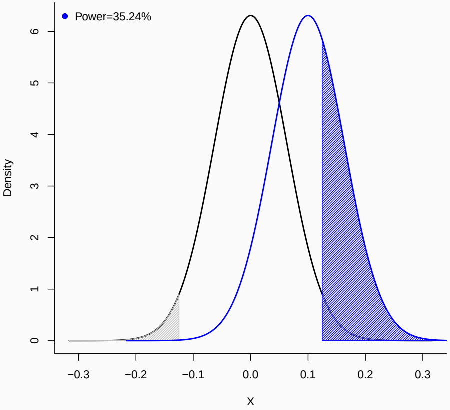
Power when $\beta = 0.1$, $N = 4$, and $p = 0.5$



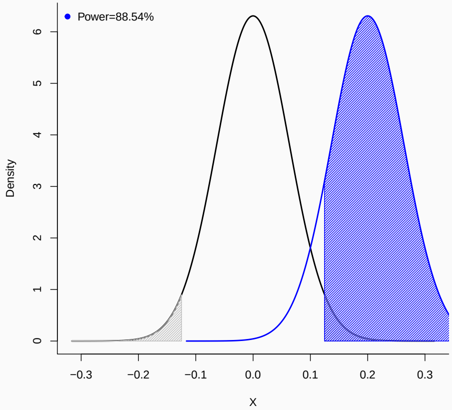
Power when $\beta = 0.1$, $N = 100$, and $p = 0.5$



Power when $\beta = 0.1$, $N = 1,000$, and $p = 0.5$

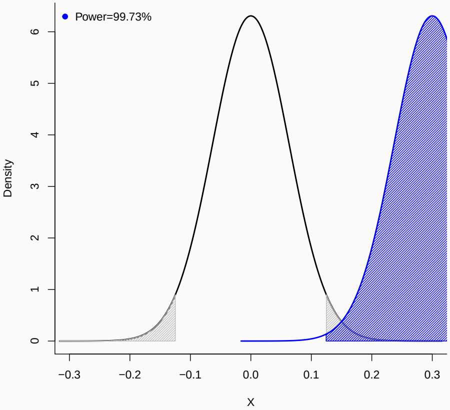


Power when $\beta = 0.2$, $N = 1,000$, and $p = 0.5$



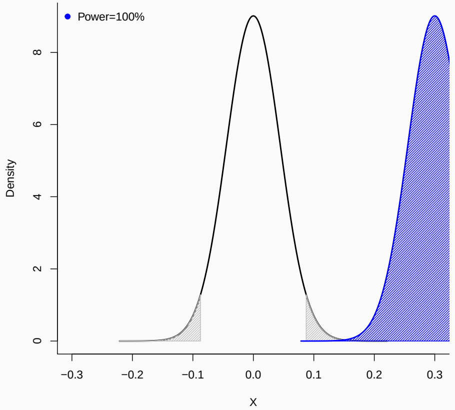
Blue area is the probability we reject the null when β is 0.2

Power when the effect is $\beta = 0.3$, $N = 1,000$, and $p = 0.5$



Blue area is the probability we reject the null when β is 0.3

Power when the effect is $\beta = 0.3$, $N = 1,000$, $p = 0.5$, and $\sigma = 0.7$



Blue area is the probability we reject the null when β is 0.3

Statistical power and clusters

- All these quantities we just looked at are related

- To achieve a power κ, it must therefore be that

$$\beta > (t_{\frac{\alpha}{2}} + t_{1-\kappa})\sigma_{\widehat{\beta}}$$

Minimum detectable effect

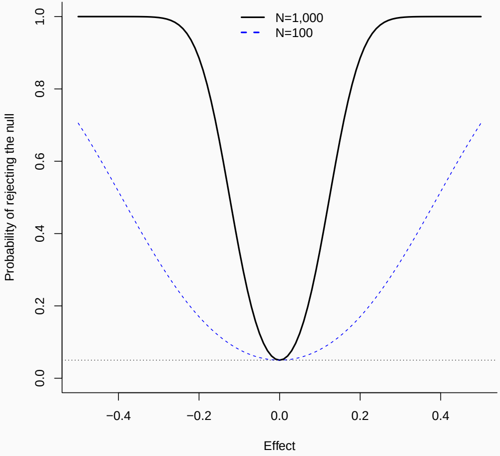
- The **minimum detectable effect** size for a given power (κ), significance level (α), sample size (N), and portion of subjects allocated to treatment group (p) is given by

$$MDE = (t_{\frac{\alpha}{2}} + t_{1-\kappa})\sqrt{\frac{\sigma^2}{p(1-p)N}}$$

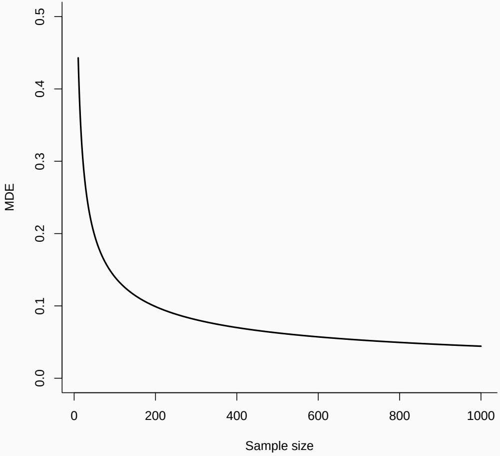
Randomizing at the Unit of Analysis

- The standard is to set $\kappa = 0.8$ or $\kappa = 0.9$
- The standard is to set $\alpha = 0.05$ or $\alpha = 0.1$
- The variance of outcomes σ^2 is typically the raw variance of the dependent variable you intend to use
- The sample size N is the number of observations in the study (you can change this)
- The fraction of the sample treated is p (you can change this)

Effect vs Power



Sample size vs MDE



How should you think about the MDE?

- What is the treatment effect below which it is pointless to implement the program and/or study its effect?
- If sample size is too small, you're likely to end up with an insignificant result for something that actually matters

Beyond the basic of OLS

Cluster Randomized Experiments

- Is the unit of treatment the same as the unit of analysis? Or, **is the treatment to be administered to a 'cluster' of units?**

Cluster Randomized Experiments

- Is the unit of treatment the same as the unit of analysis? Or, **is the treatment to be administered to a 'cluster' of units?**
- Examples of clustered randomizations:
  - Changing the business practices at a firm level and studying the impact on individual employees
  - Providing schools with new textbooks and studying the effect on individual student performance
  - Offering a new financial service to all residents in a village and studying the impact on micro enterprise outcomes
- In a clustered randomization the power of the study is coming partly from the number of individuals in the study, and partly from the number of clusters in the study

Cluster Randomized Experiments

- The estimate of treatment effect is $\widehat{\beta}$ in the regression

$$Y_{ij} = \alpha + \beta T_j + \omega_j + \varepsilon_{ij}$$

- σ^2 is the variance of the outcome (ε_{ij})
- τ^2 is the variance of the outcome (ω_j)
- p is the proportion of treated units
- n is the number of observations in each cluster
- J is the number of clusters
- The variance of $\widehat{\beta}$ is $\sigma_{\widehat{\beta}} = \frac{n\tau^2+\sigma^2}{p(1-p)nJ}$

Cluster Randomized Experiments

- Often, expressed using the intra-cluster correlation (ICC) $\equiv \frac{\tau^2}{\tau^2+\sigma^2}$
- The variance of $\widehat{\beta}$ is $V(\widehat{\beta}) = \sigma^2 \frac{\rho + \frac{(1-\rho)}{n}}{p(1-p)J}$ (comes from the cluster SE formula we saw)
- The ICC can be obtained using *loneway* in stata

Minimum detectable effect

- The **minimum detectable effect** is given by

$$MDE = (t_{\frac{\alpha}{2}} + t_{1-\kappa})\sigma\sqrt{\frac{\rho + \frac{(1-\rho)}{n}}{p(1-p)J}}$$

Power Calculations Rules of Thumb

- For an individual-level experiment, 200-300 observations will typically be sufficient to detect a reasonable effect size
- For a clustered experiment, a low ICC (0.1) would need 50-100 clusters and > 5 observations per cluster to detect a moderate effect. As the ICC gets larger, the number of **clusters** has to go up
- For **very** complicated research designs, you can always use simulations to get the power of the design