Monte Carlo Simulation

Taro Mieno

AECN 896-003: Applied Econometrics

Monte Carlo Simulation

What is it?

test econometric theories (prediction) via simulation

Monte Carlo Simulation

How is it used in econometrics?

- confirm ecoometric theory numerically
  1. OLS estimators are unbiased if $E[u|x] = 0$ along with other conditions (theory)
  2. I know the above theory is right, but let's check if it is true numerically
- You kind of sense that something in your data may cause problems, but there is no proven econometric theory about what's gonna happen (I used MC simulation for this purpose a lot)
- assist students in understanding econometric theories by providing actual numbers instead of a series of Greek letters

Monte Carlo Simulation

Question

Suppose you are interested in checking what happens to OLS estimators if $E[u|x] = 0$ (the error term and x are not correlated) is violated.

Can you use the real data to do this?

Monte Carlo Simulation

Key part of MC simulation

You generate data (you have control over how data are generated)

- You know the true parameter unlike the real data generating process
- You can change only the part that you want to change about data generating process and econometric methods with everything else fixed

Generating data

Pseudo random number generator

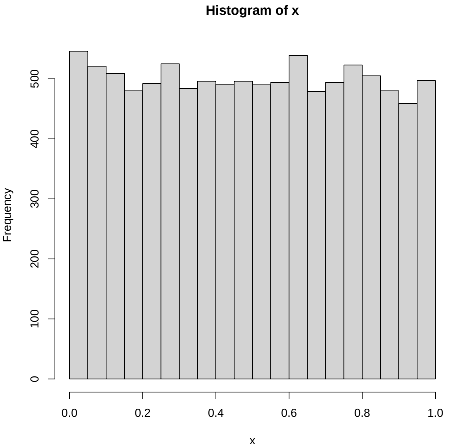
algorithm for generating a sequence of numbers whose properties approximate the properties of sequences of random numbers

Examples in R: Uniform Distribution

```r
runif(5) # default is min=0 and max=1
[1] 0.11011017 0.19312151 0.20463320 0.05153304 0.75362836
```

Examples in R: Uniform Distribution

```
x <- runif(10000)
hist(x)
```



Pseudo random number generator

- Pseudo random number generators are not really random number generators
- What numbers you will get are pre-determined
- What numbers you will get can be determined by setting a seed
- An example:

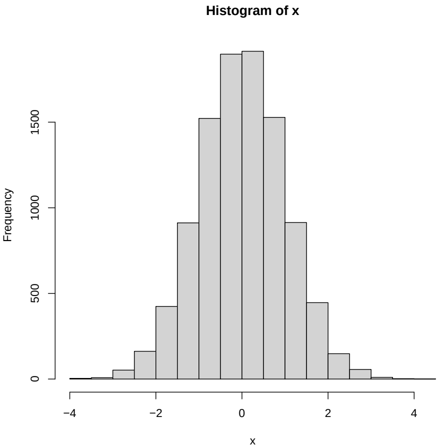
```r
set.seed(2387438)
runif(5)
```

```
[1] 0.0474233 0.7116970 0.4066674 0.2422949 0.3567480
```

- What benefits does setting a seed have?

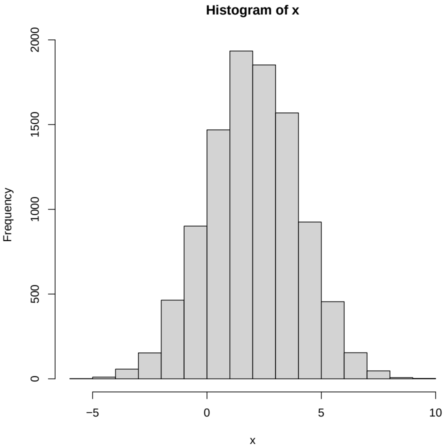
Examples in R: Normal Distribution

```r
x <- rnorm(10000) # default is mean=0,sd=1

hist(x)
```



Examples in R: Normal Distribution

```r
x <- rnorm(10000,mean=2,sd=2) # mean=2,sd=2
hist(x)
```



Other distributions

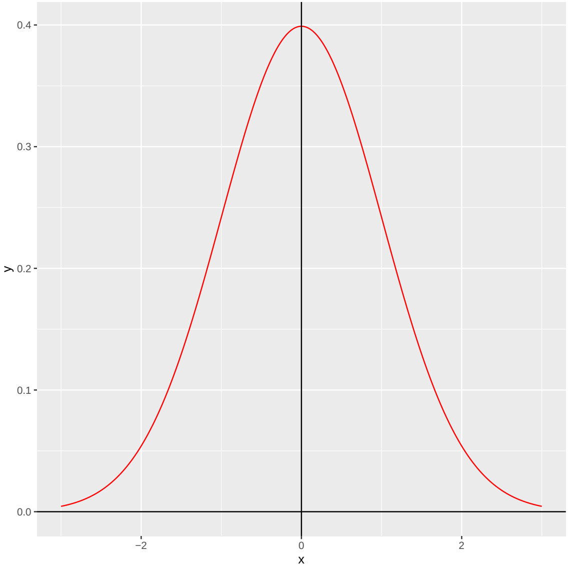
- Beta
- Chi-square
- F
- Logistic
- Log-normal
- many others

r,p,r,d

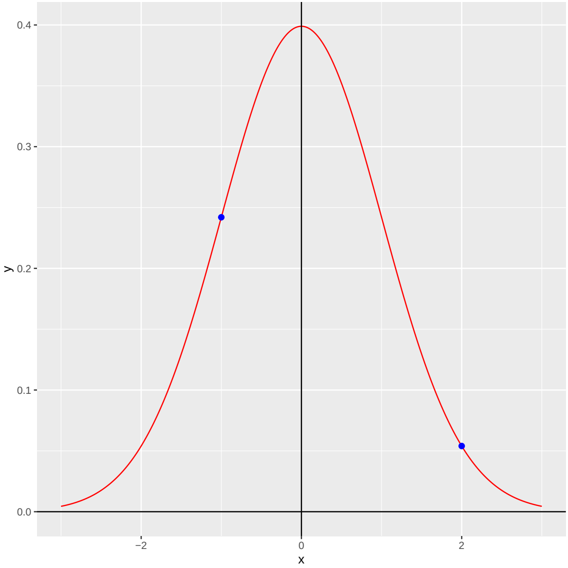
For each distribution, you have four different kinds of functions:

- $pnorm$: distribution function
- $qnorm$: quantile function
- $dnorm$: density function
- $rnorm$: random draw

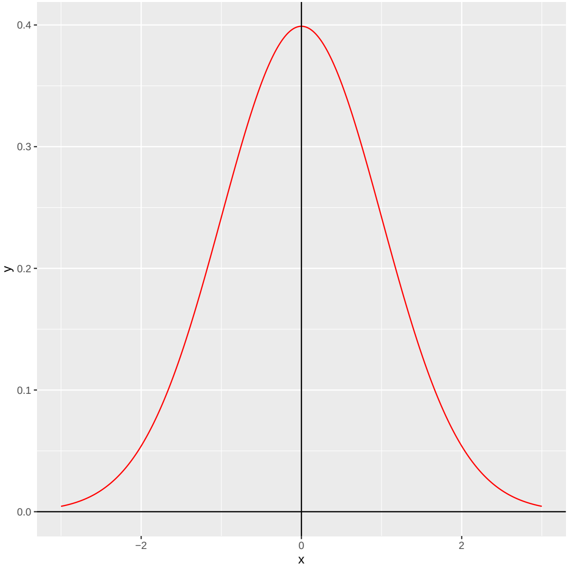
pnorm



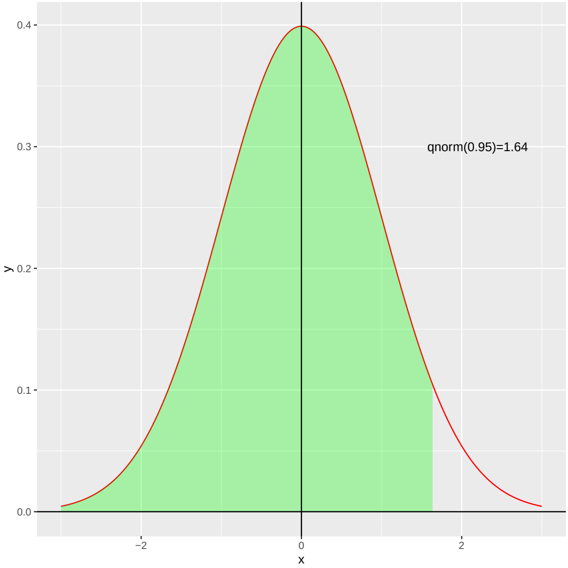
pnorm



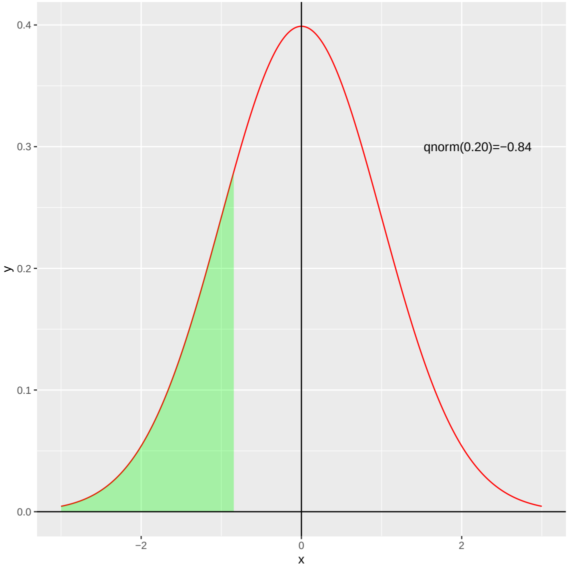
$qnorm$



qnorm



qnorm



Monte Carlo Simulation: Steps

1. specify the data generating process
2. generate data based on the data generating process
3. get an estimate based on the generated data (e.g. OLS, mean)
4. repeat the above steps many many times
5. compare your estimates with the true parameter

Question

Why do the steps $1 - 3$ many many times?

Monte Carlo Simulation: Example 1

Question

Is sample mean really an unbiased estimator of the expected value? ($\frac{1}{n}\sum_{i=1}^{n} x_i = E[x]$ where x_i is an independent random draw from the same distribution)

Sample Mean: Steps 1-3

R code: Steps 1-3

```r
#--- steps 1 and 2:  ---#
# specify the data generating process and generate data
x <- runif(100) # Here, E[x]=0.5

#--- step 3 ---#
# calculate sample mean
mean_x <- mean(x)
mean_x
```

```
[1] 0.507078
```

Sample Mean: Step 4

- Step 4: repeat the above steps many times (why?)
- We use a loop to do the same (similar) thing over and over again

R code: for loop

```
#--- the number of iterations ---#
B <- 1000

#--- repeat steps 1-3 B times ---#
for (i in 1:B){
  print(i) # print i
}
```

Sample Mean: Step 4

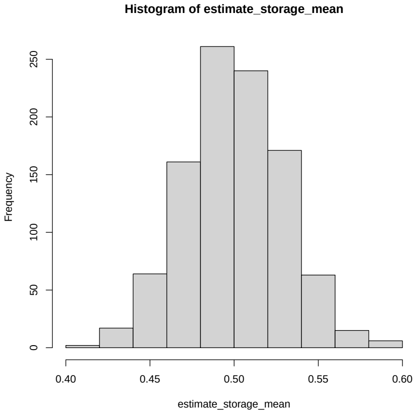
R code: Step 4

```r
#--- the number of iterations ---#
B <- 1000

#--- create a storage that stores estimates ---#
estimate_storage_mean <- rep(0,B)

#--- repeat steps 1-3 B times ---#
for (i in 1:B){
  #--- steps 1 and 2:  ---#
  # specify the data generating process and generate data
  x <- runif(100) # Here, E[x]=0.5

  #--- step 3 ---#
  # calculate sample mean
  mean_x <- mean(x)
  estimate_storage_mean[i] <- mean_x
}
```

Sample Mean: Step 5

Compare your estimates with the true parameter

```r
mean(estimate_storage_mean)
[1] 0.500199
hist(estimate_storage_mean)
```



Monte Carlo Simulation: Example 2

Question

What happens to β_1 if $E[u|x] \neq 0$ when estimating $y = \beta_0 + \beta_1 x + u$?

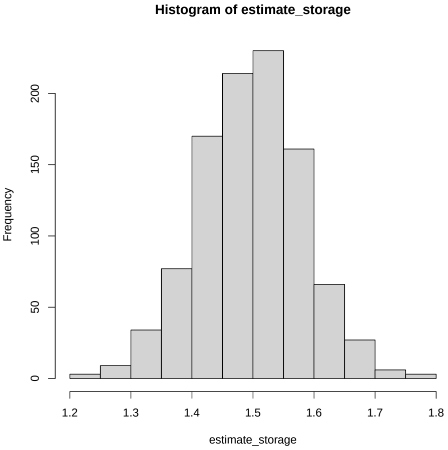
Example 2

R code: Example 2

```r
#--- Preparation ---#
B <- 1000 # the number of iterations
N <- 100 # sample size
estimate_storage <- rep(0,B) # estimates storage

#--- repeat steps 1-3 B times ---#
for (i in 1:B){
  #--- steps 1 and 2:  ---#
  mu <- rnorm(N) # the common term shared by both x and u
  x <- rnorm(N) + mu # independent variable
  u <- rnorm(N) + mu # error
  y <- 1 + x + u # dependent variable
  data <- data.table(y=y,x=x)

  #--- OLS ---#
  reg <- lm(y~x,data=data) # OLS
  estimate_storage[i] <- reg$coef['x']
}
```

Example 2

```r
hist(estimate_storage)
```



MC Simulation: Exercise 1

Problem

Using MC simulations, find out how the variance of error term affects the variance of OLS estimators

Model set up

$$y = \beta_0 + \beta_1 x + u_1$$
$$y = \beta_0 + \beta_1 x + u_2$$

- $x \sim N(0, 1)$
- $u_1 \sim N(0, 1)$ and $u_2 \sim N(0, 9)$
- $E[u_1|x] = 0$ and $E[u_2|x] = 0$

Question

What should you expect?

Example 3: Estimation of the Variance of OLS Estimators

True Variance of $\hat{\beta}_1$

$$V(\hat{\beta}_1) = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{\sigma^2}{SST_X} \tag{1}$$

Estimated Variance of $\hat{\beta}_1$

$$\hat{V}(\hat{\beta}_1) = \frac{\hat{\sigma}^2}{SST_X} = \frac{\sum_{i=1}^{n}\hat{u}_i^2}{n-2} \times \frac{1}{SST_X} \tag{2}$$

R code: Example 3

```r
set.seed(903478)

#--- Preparation ---#
B <- 10000 # the number of iterations
N <- 100 # sample size
beta_storage <- rep(0,B) # estimates storage for beta
V_beta_storage <- rep(0,B) # estimates storage for V(beta)
x <- rnorm(N) # x values are the same for every iteration
SST_X  <- sum((x-mean(x))^2)

#--- repeat steps 1-3 B times ---#
for (i in 1:B){
  #--- steps 1 and 2:  ---#
  u <- 2*rnorm(N) # error
  y <- 1 + x + u # dependent variable
  data <- data.frame(y=y,x=x)

  #--- OLS ---#
  reg <- lm(y~x,data=data) # OLS
  beta_storage[i] <- reg$coef['x']
  V_beta_storage[i] <- vcov(reg)['x','x']
}
```

Example 3

True Variance

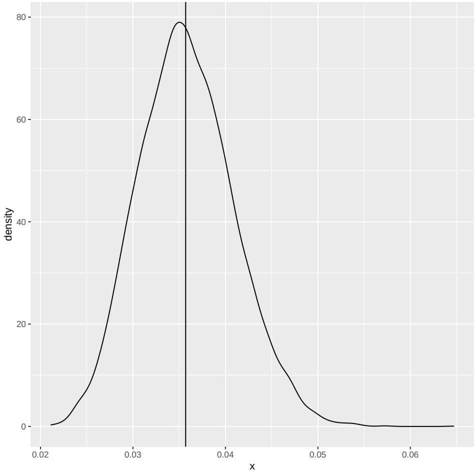
$$V(\hat{\beta}) = 4/112.07 = 0.0357 \tag{3}$$

check

```
    var(beta_storage)
[1] 0.03562348
```

Your Estimates of Variance of $\hat{\beta}_1$?

```r
#=== mean ===#
mean(V_beta_storage)
[1] 0.03579118
```



MC Simulation: Exercise 2

Problem

Using MC simulations, find out how the variation in x affects the OLS estimators

Model set up

$$y = \beta_0 + \beta_1 x_1 + u$$
$$y = \beta_0 + \beta_1 x_2 + u$$

- $x_1 \sim N(0, 1)$ and $x_2 \sim N(0, 9)$
- $u \sim N(0, 1)$
- $E[u_1|x] = 0$ and $E[u_2|x] = 0$

Question

What should you expect?