

pset 1 econ 21020

Matthew Zhao

Description

This RMarkdown file is meant to be a more readable equivalent to the R file by the same name. This corresponds with the first pset of Econometrics.

Here we open the ggplot2 library and set the seed for reproducibility.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
set.seed(1000)
```

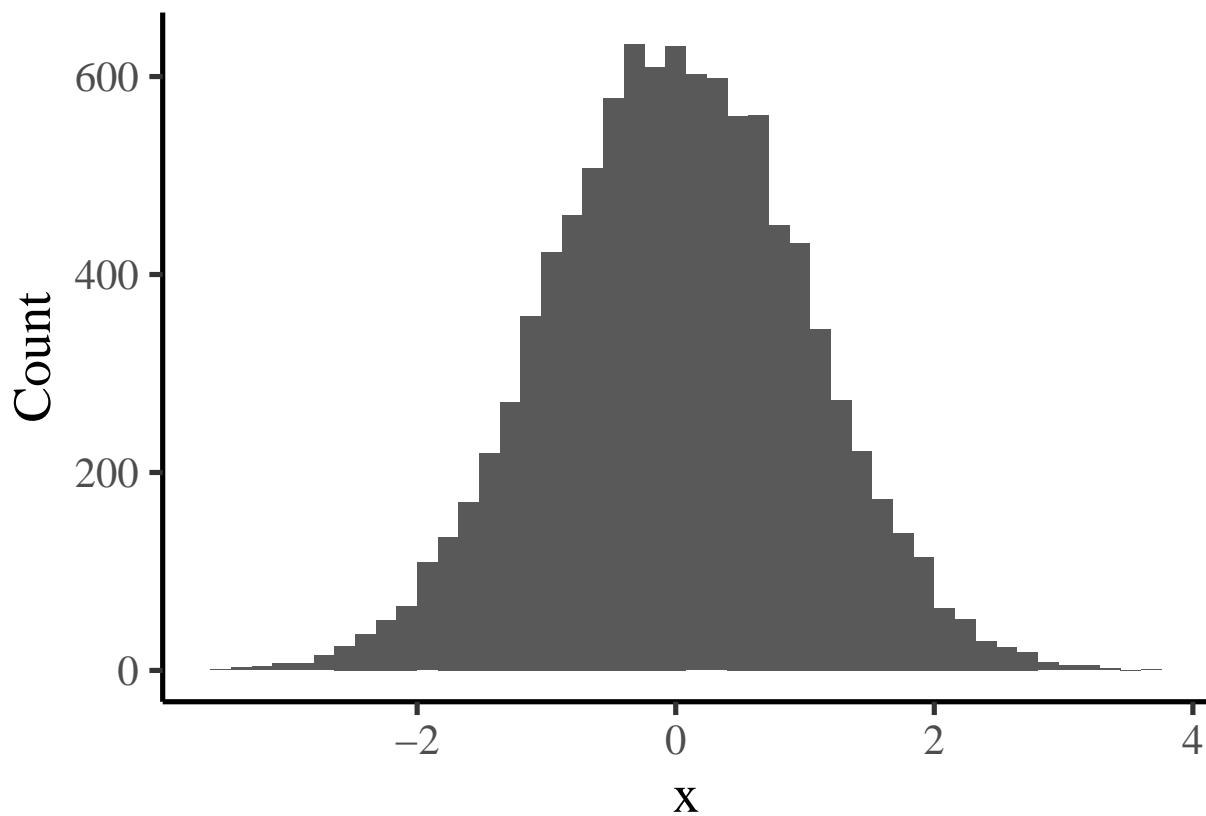
Problem 8

Part (a) Generate a vector of n draws from a standard normal rv

```
n <- 10000
mu <- 0
sigma <- 1
x <- rnorm(n, mu, sigma)
```

Plot a histogram of the draws using ggplot2

```
ggplot() +
  geom_histogram(aes(x = x), binwidth = 0.16) +
  ylab("Count") + xlab("x") +
  theme_classic(base_size = 20) +
  theme(text = element_text(size = 20, family="serif"))
```

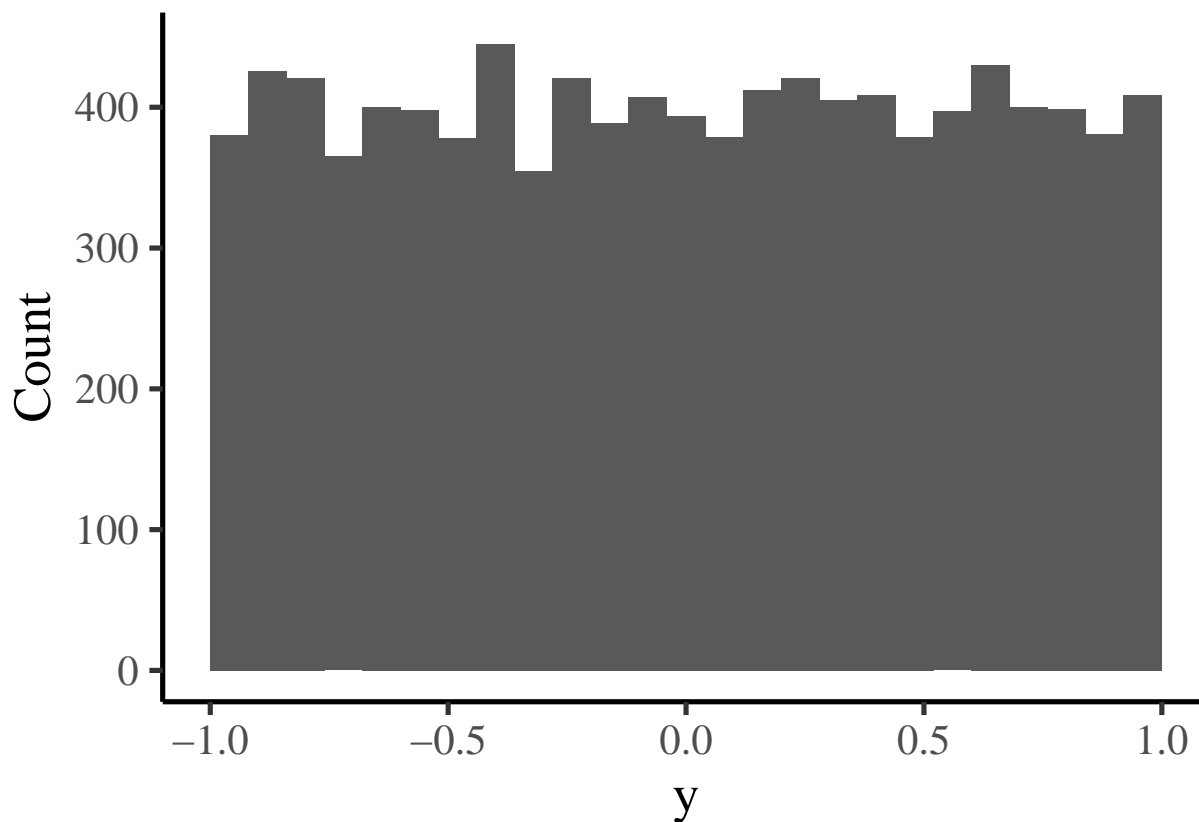


Part (b) Generate a vector of n draws from a uniform(-1, 1) rv

```
n <- 10000
min_y <- -1
max_y <- 1
y <- runif(n, min_y, max_y)
```

Plot a histogram of the draws using ggplot2

```
ggplot() +
  geom_histogram(aes(x = y), binwidth = 0.08) +
  ylab("Count") + xlab("y") +
  theme_classic(base_size = 20) +
  theme(text = element_text(size = 20, family = "serif"))
```



Problem 9

Part (a)

Part (b) Define a custom function that returns draws from a Bernoulli rv

```
my_rbernoulli <- function (n, p) {
  x <- ifelse(runif(n,0,1)<=p,0,1)
  return(x)
}
```

Test the custom Bernoulli generator function

```
set.seed(1000)
x <- my_rbernoulli(10000, 0.5)
length(x) == 10000 # should return TRUE
```

```
## [1] TRUE
```

```
mean(x) # should be a number near 0.5
```

```
## [1] 0.4975
```

Part (c) Define a custom function that returns draws from a Binomial rv Test the custom Bernoulli generator function

```
my_rbinomial <- function(n, p, m){  
  for (y in 1:n) {  
    ifelse(y==1,x <- as.vector(sum(my_rbernoulli(m,p))),  
          x <- append(x,sum(my_rbernoulli(m,p))))  
  }  
  return(x)  
}
```

Test the custom Binomial generator function

```
set.seed(1000)  
x <- my_rbinomial(10000, 0.5, 10)  
length(x) == 10000 # should return TRUE
```

```
## [1] TRUE
```

```
mean(x) # should be a number near 5
```

```
## [1] 4.9974
```