

# Lecture\_4\_MACSS

Auffhammer

2024-09-19

Now let's get ambitious. We are going to generate some data on health status of some made up folks and play random sampling and random assignment.

```
rm(list = ls()) # clear memory
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library('MASS')
```

```
##
## Attaching package: 'MASS'
## The following object is masked from 'package:dplyr':
##
##   select
```

```
library('crosstable')
library('flextable')
set.seed(22092008) # set random number generator seed
N <- 10000 #Population Size
n <- 1000 # Sample Size
mu <- c(0, 0,0)
# a <- 0.5 #Gender Income Covariance
# b <- 0.1 #Gender Insurance Covariance
# c <- 0.8 #Income insurance

# If insurance were randomized
a <- 0.5 # Set to 0.5 as default
b <- 0.0 # Set to 0.1 as default
c <- 0.0 # Set to 0.8 as default

# Some betas for later
b1 <- -1 #Gender Beta
b2 <- -5 # Income Beta
b3 <- -3 #Insurance
shifter <- 30
```

```

Sigma <- matrix(c(1, a, b, a, 1, c, b, c, 1), nrow=3)
data = mvrnorm(N, mu, Sigma, empirical=FALSE)
Gender = data[, 1] # standard normal (mu=0, sd=1)
Income = data[, 2] # standard normal (mu=0, sd=1)
Insurance= data[, 3] # standard normal (mu=0, sd=1)

# Gender and income should be binary
Ins <- Insurance>0
Gend <- Gender>0
cor(Ins,Gend)

## [1] -0.003263571

cor(Ins,Income)

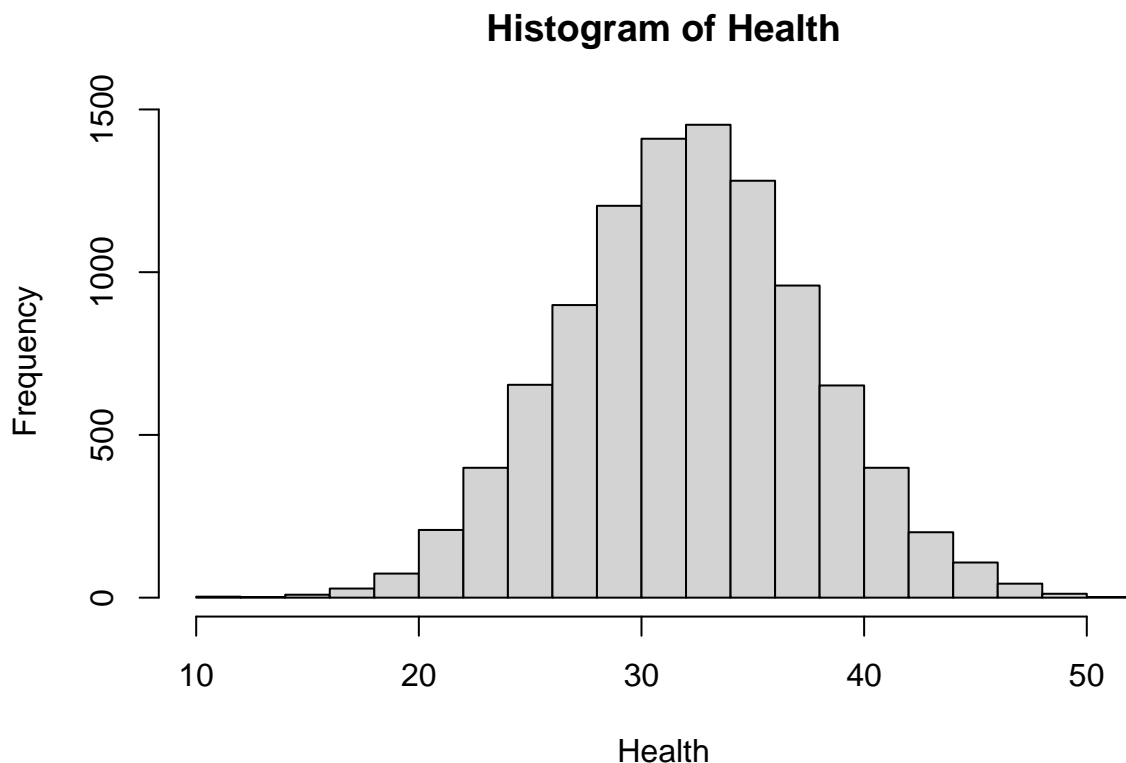
## [1] -0.007736086

cor(Gend,Income)

## [1] 0.4073727

# We are going to generate some arbitrary Health Index
Health <- shifter + rnorm(N,mean=0,sd=1) + b1*Gend + b2*Income + b3* Ins
# Plot my Outcome
hist(Health)

```



```

# Let's make a nice table comparing groups across insurance
mydata <- data.frame(Income, Gend, Health, Ins)
ft1 <- crosstable(mydata,by="Ins", test=TRUE, funs=c(mean=mean, "std error"=sd)) %>%
  as_flextable()

```

```
## Warning in crosstable(mydata, by = "Ins", test = TRUE, funs = c(mean = mean, : Be aware that automat.
```

```
## context, as it would cause extensive alpha inflation otherwise.  
## This warning is displayed once every 8 hours.
```

```
print (ft1)
```

```
## a flextable object.  
## col_keys: `label`, `variable`, `FALSE`, `TRUE`, `test`  
## header has 2 row(s)  
## body has 6 row(s)  
## original dataset sample:  
##      .id label variable      FALSE      TRUE  
## 1 Income Income      mean      0.03      0.01  
## 2 Income Income std error      1.0      1.0  
## 3  Gend  Gend      FALSE 2452 (49.40%) 2512 (50.60%)  
## 4  Gend  Gend      TRUE  2504 (49.72%) 2532 (50.28%)  
## 5 Health Health      mean      30.6      33.6  
##  
##  
## 1          p value: 0.4392 \n(Two Sample t-test)  
## 2          p value: 0.4392 \n(Two Sample t-test)  
## 3 p value: 0.7442 \n(Pearson's Chi-squared test)  
## 4 p value: 0.7442 \n(Pearson's Chi-squared test)  
## 5    p value: <0.0001 \n(Wilcoxon rank sum test)
```

We will play with this more next time.