14_1A Randomization Slope

Michael Sullivan

2023-08-28

First, we will enter the data in Table 1 from Section 14.1A.

```
zestimate <- c(291.5,320,371.5,303.5,351.5,314,332.5,295,313,368)
sell_price <- c(268,305,360,283,340,275,356,300,285,390)
Table1 <- data.frame('Zestimate'=zestimate,'Selling_price'= sell_price)
head(Table1)
##   Zestimate Selling_price
## 1     291.5           268
## 2     320.0           305
## 3     371.5           360
## 4     303.5           283
## 5     351.5           340
## 6     314.0           275
```

The slope of the least-squares regression line treating the Zestimate as the explanatory variable is 1.3059. Does the slope of 1.3059 suggest that higher Zestimates correspond to a higher selling price, or is it possible that the two variables are not positively associated?

First, let's get a sense of how the question may be answered using randomization techniques. So that you can follow along, we use use a fixed seed.

Assuming there is no relation between the Zestimate and selling price, we could randomly assign a selling price to any Zestimate. So, randomly assign the 10 selling prices to the 10 Zestimates as follows.

```
set.seed(33)
sample1 <- sample(Table1$Selling_price, size=10, replace=FALSE)
sample1
##  [1] 390 300 275 283 268 305 356 285 360 340
```

Now, determine the slope of the least-squares regression using sample1 as the response variable.

```
slope <- lm(sample1 ~ Table1$Zestimate)$coefficients[[2]]
slope
## [1] -0.4002403
```

The slope of the least-squares regression model using sample1 as the response variable is -0.4002.

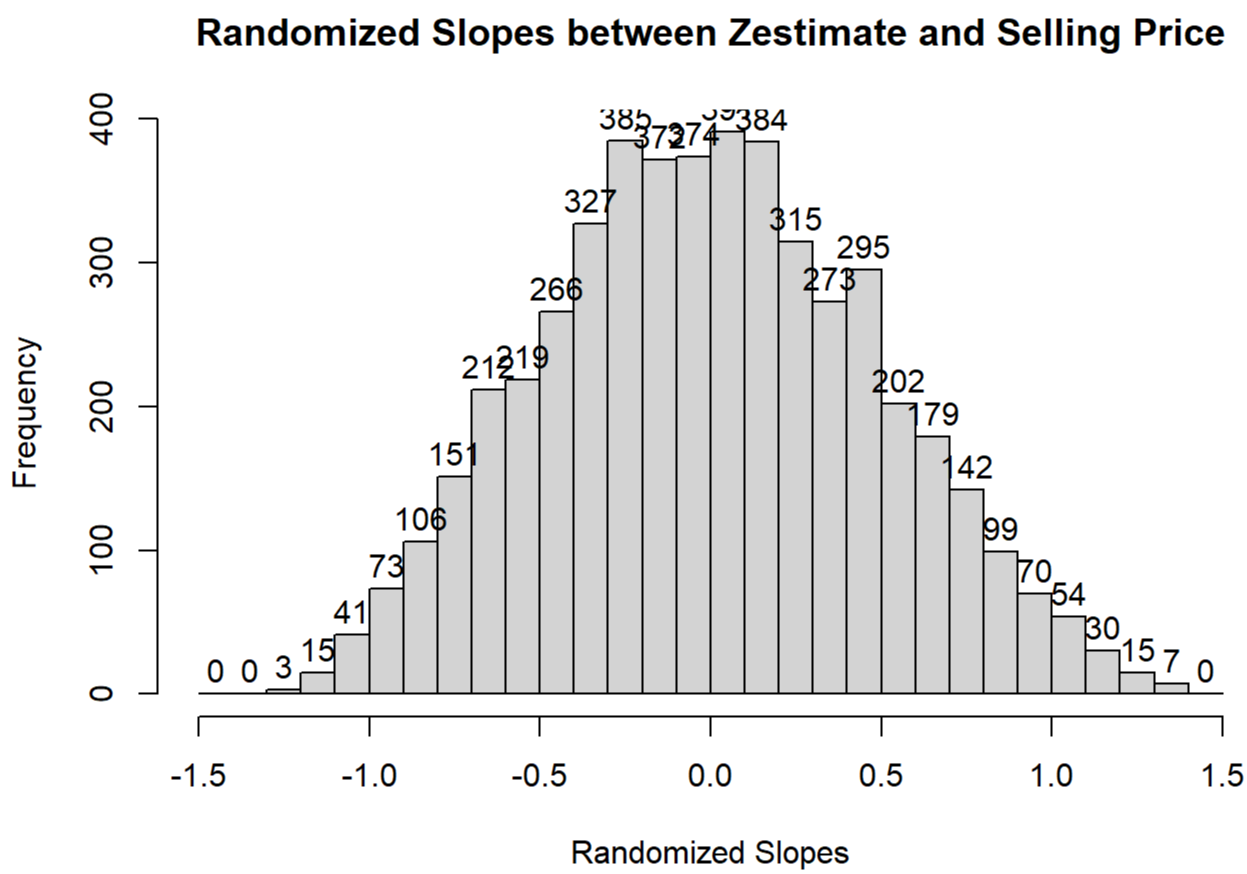
As in the other randomization methods, we want to repeat this process many times to determine the proportion of times (out of 5000) we observe a slope of 1.3059. Use the command below.

```
Randomize <- c(replicate(5000, lm(sample(Table1$Selling_price,size=10,replace=FALSE)~T
able1$Zestimate)$coefficients[[2]]))

#Note Randomize <- c() creates a vector of randomized slopes. The 5000 represents 5000
random samples.  The $coefficients[[2]] is used to grab only the slope coefficient and
store the results in the vector Randomize.
```

Draw a histogram of the randomized least-squares regression slopes.

```
hist(Randomize,labels=TRUE,breaks = seq(-1.5,1.5,0.1),xlab="Randomized Slopes",main="R
andomized Slopes between Zestimate and Selling Price")
```



Now determine the proportion of randomized slopes that are as extreme or more extreme than the observed slope of 1.3059.

```
sum(Randomize>=1.3059)/length(Randomize)

## [1] 0.0014

# sum(Randomize>=1.3059) will count the number of times the vector Randomize has a val
ue 1.3059 or higher.
```

```
# length(Randomize) determines the number of elements in the vector Randomize.
```

The proportion of randomized slopes of 1.3059 or higher is 0.0014.

Mosaic

The Mosaic library has a couple of functions that make performing a randomization test for the slope very easy. The do(n) function and the shuffle() function. The do() function performs a certain command n times. The shuffle() function shuffles the values of a certain variable (such as selling price). First, let's review how to find the coeffficients of the least-squares regression model using the data from Table 1 introduced earlier.

```
library(mosaic)
lm_object <- lm(Selling_price ~ Zestimate, data = Table1)
coefficients(lm_object)
## (Intercept)   Zestimate
## -109.578371    1.305868
```

The coefficient of the explanatory variable "Zestimate" is 1.3059.

To perform a randomization test, shuffle the response variable selling price. Our test statistic is the slope of the least-squares regression, so we only want to extract the slope after randomizing. Let's do this five times.

```
set.seed(33)
Randomize <- do(5) * lm(shuffle(Table1$Selling_price)~Zestimate,data=Table1)
Randomize$Zestimate
## [1] -0.4830733 -0.2448619  0.4549048 -0.6828550  0.6691556
# Note  The command Randomize$Zestimate displays only the coefficient of the slope in
the least-squares regression model.
```

The five slope coefficients are -0.4831, -0.2449, 0.4549, -0.6829, and 0.6692. We want to know the proportion of randomizations that result in a slope coefficient of 1.3059 or higher.

```
set.seed(33)
Randomize <- do(5000) * lm(shuffle(Table1$Selling_price)~Zestimate,data=Table1)
tally(Randomize$Zestimate >= 1.3059)
## X
##  TRUE FALSE
##     7  4993
```

So, 7 of the 5000 randomizations resulted in a slope coefficient of 1.3059 or higher. The estimated P-value is 7/5000 = 0.0014.

We can visualize the null model by drawing a density plot.

```
densityplot(Randomize$Zestimate)
```