

Coin Toss and R Markdown Introduction

2024-09-19

R Markdown (src: RStudio)

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Math 2265 Section 4.1

Load Packages

```
library(openintro)
```

```
## Loading required package: airports
```

```
## Loading required package: cherryblossom
```

```
## Loading required package: usdata
```

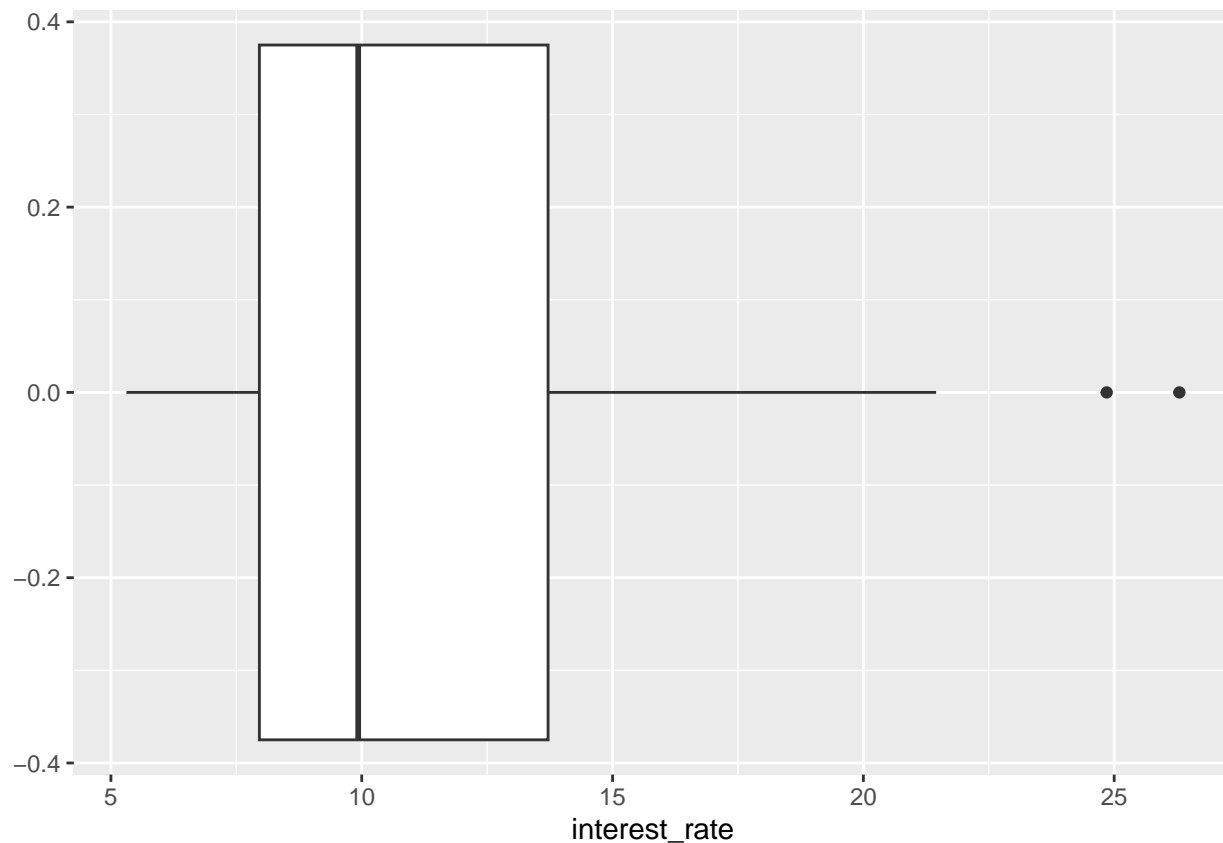
```
library(ggplot2)
```

```
set.seed(2265)
```

Short Recap

In the following, we plot the box plot of the variable `interest_rate` in the dataset `loan50`. See how R Markdown runs R code.

```
ggplot(data=loan50, mapping=aes(x=interest_rate)) +  
  geom_boxplot()
```



Heads and Tails

We make a simulation of a fair coin toss. Recall there is an equal chance of getting a head or tail.

```
coin <- c("H", "T") # define a vector consisting of "H" and "T"
sample(coin, 1)
```

```
## [1] "T"
```

Execute this cell by clicking on the green `|>` button.

Change the code to run the simulation 10 times.

```
coin <- c("H", "T") # define a vector consisting of "H" and "T"
sample(coin, <>)
```

```
## Error: <text>:2:14: unexpected '<'
## 1: coin <- c("H", "T") # define a vector consisting of "H" and "T"
## 2: sample(coin, <
##      ^
```

This will result in an error since we want to grab 10 when there are only two. We can set the `replace` parameter `TRUE`.

```
coin <- c("H", "T") # define a vector consisting of "H" and "T"
sample(coin, 10, replace = TRUE)
```

```
## [1] "H" "H" "H" "H" "T" "H" "H" "H" "T" "H"
```

Run this a few times.

Questions

Question 1. Did you get the even number of heads and tails?

Ans:

Question 2. Complete the following code to simulate tossing a coin 500 times.

```
coin <- c("H", "T") # define a vector consisting of "H" and "T"
result <- sample(coin, <>, replace = TRUE)
table(result) # `table` tallies the result

## Error: <text>:2:24: unexpected '<'
## 1: coin <- c("H", "T") # define a vector consisting of "H" and "T"
## 2: result <- sample(coin, <
##                               ^
```

Run this code a few times.

The Law of Large Numbers As more observations are collected, the proportion of occurrences with a particular outcome converges to the probability of that outcome.

Question 3.

The following code runs 1000 simulations of tossing coins 500 times and saves the number of heads in each simulation to the variable `num_heads`.

```
coin <- c("H", "T") # define a vector consisting of "H" and "T"
num_heads <- replicate(1000, as.numeric(table(sample(coin, 500, replace = TRUE))[1]))
data_coin <- data.frame(
  num_heads
) # it will be used in the next question
num_heads[1:5] # prints the first 5 results
```

```
## [1] 242 252 241 253 255
```

Make a histogram of the simulations with the bin number of your choice.

```
print(str(data_coin))
ggplot(data=data_coin, aes(x=num_heads)) +
  # xlim(0, 500) + # to make it to display x from 0 to 500
  geom_<>(bins=<>)
```

```
## Error: <text>:4:9: unexpected '>'
## 3:   # xlim(0, 500) + # to make it to display x from 0 to 500
## 4:   geom_<>
##       ^
```

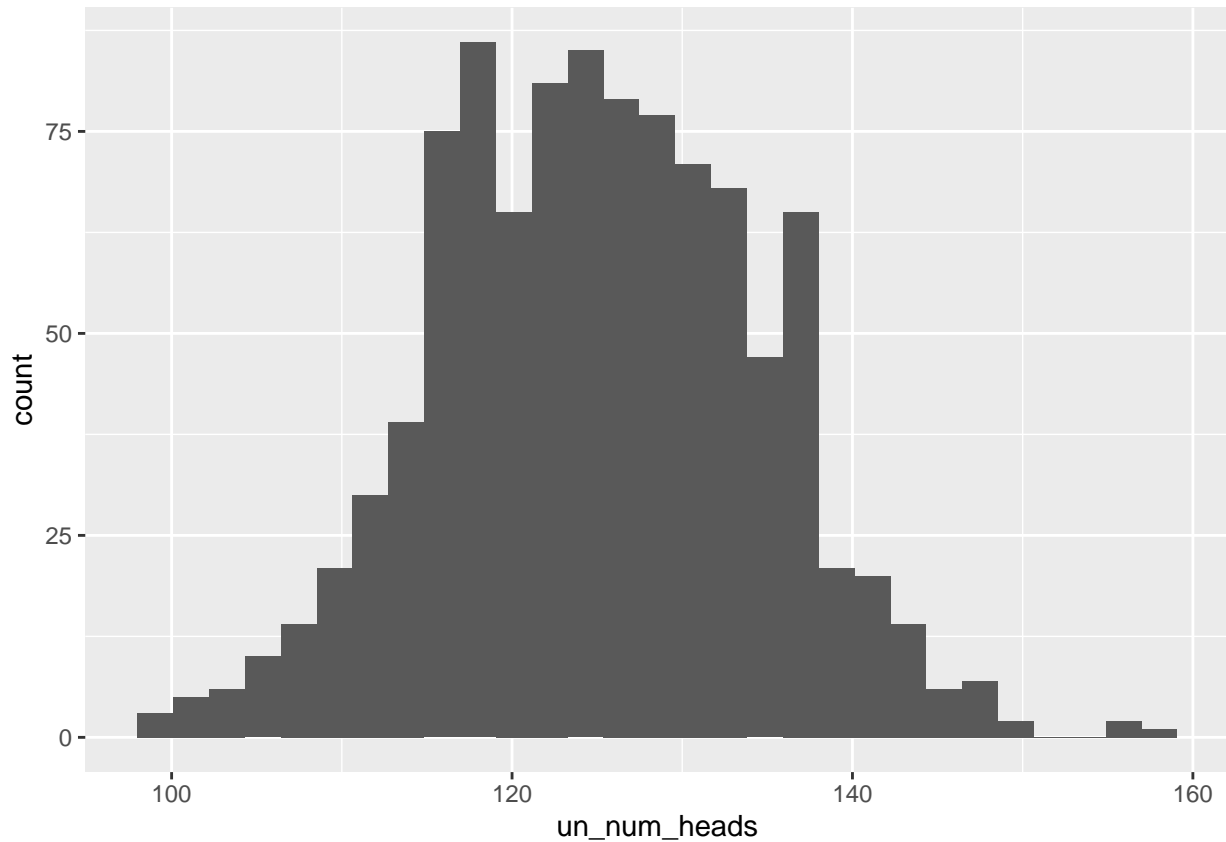
Example of an unfair coin.

In this example, we have an unfair coin where the change of getting tail is 75%. Compare this histogram and the one you made in the previous question.

```

coin <- c("H", "T") # define a vector consisting of "H" and "T"
un_num_heads <- replicate(1000, as.numeric(table(sample(coin, 500, replace = TRUE, prob=c(0.25,0.75))))[1])
data_un_coin <- data.frame(
  un_num_heads
) # it will be used in the next question
ggplot(data=data_un_coin, aes(x=un_num_heads)) +
  # xlim(0, 500) +
  geom_histogram(bins=29)

```



Question 4.

Which simulation has a bigger mean of the number of heads? (fair coin vs. unfair coin above) To answer this question, it may help uncommenting the line `xlim(0, 500) +`.

Ans:

Upload your work to Canvas