CUNY SPS DATA606 Lab3 - Probability

Chinedu Onyeka

September 16th, 2021

Getting Started

Load packages

```
library(tidyverse)
library(openintro)
```

```
glimpse(kobe_basket)
```

```
## Rows: 133
## Columns: 6
## $ vs          <fct> ORL, ORL, ORL, ORL, ORL, ORL, ORL, ORL, ORL, ORL, ORL, ORL~
## $ game        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ quarter     <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3~
## $ time        <fct> 9:47, 9:07, 8:11, 7:41, 7:03, 6:01, 4:07, 0:52, 0:00, 6:35~
## $ description <fct> Kobe Bryant makes 4-foot two point shot, Kobe Bryant misse~
## $ shot        <chr> "H", "M", "M", "H", "H", "M", "M", "M", "M", "H", "H", "H"~
```

1. What does a streak length of 1 mean, i.e. how many hits and misses are in a streak of 1? What about a streak length of 0?

Solution 1:

A streak of 1 means there is 1 hit and one miss, while a streak of 0 means there is 0 hit (1 miss) and then another miss.

2. Describe the distribution of Kobe's streak lengths from the 2009 NBA finals. What was his typical streak length? How long was his longest streak of baskets? Make sure to include the accompanying plot in your answer.
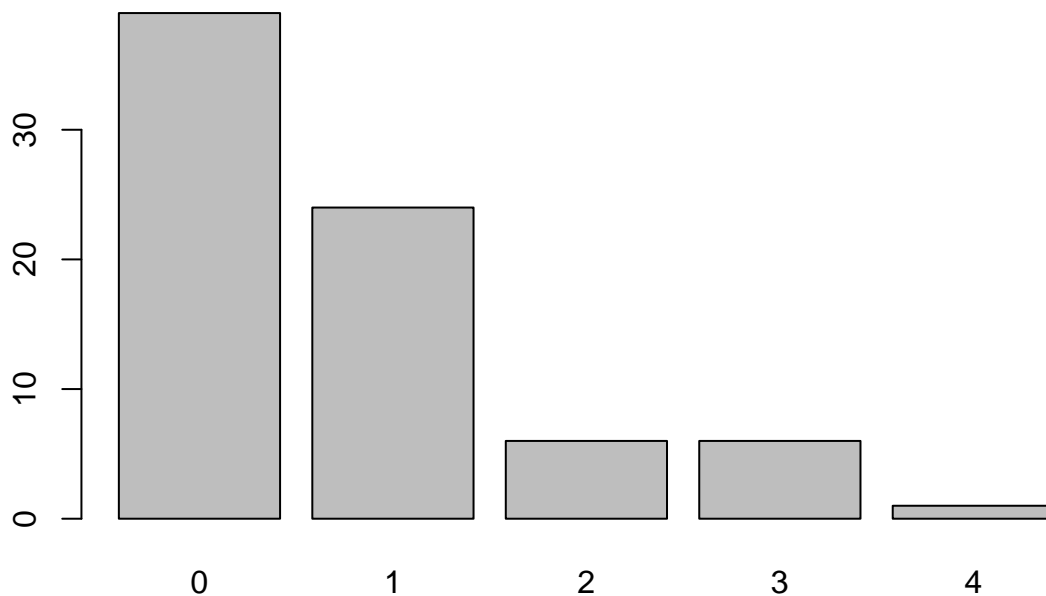
Solution 2:

```
kobe_streak <- calc_streak(kobe_basket$shot)
```

```
summary(kobe_streak)
```

```
##     length
## Min.    :0.0000
## 1st Qu.:0.0000
## Median :0.0000
## Mean    :0.7632
## 3rd Qu.:1.0000
## Max.    :4.0000
```

This further shows that the distribution is skewed to the right.
Kobe's typical streak length is 0 and his longest streak is 4.

```r
kobe_streak <- table(kobe_streak)
barplot(kobe_streak)
```


Simulations in R

As a simple example, you can simulate flipping a fair coin with the following.

```r
coin_outcomes <- c("heads", "tails")
sample(coin_outcomes, size = 1, replace = TRUE)
```

```
## [1] "tails"
```

If you wanted to simulate flipping a fair coin 100 times, you could either run the function 100 times or,
more simply, adjust the `size` argument, which governs how many samples to draw (the `replace = TRUE`
argument indicates we put the slip of paper back in the hat before drawing again). Save the resulting vector
of heads and tails in a new object called `sim_fair_coin`.

```
sim_fair_coin <- sample(coin_outcomes, size = 100, replace = TRUE)
```

To view the results of this simulation, type the name of the object and then use `table` to count up the number of heads and tails.

```
sim_fair_coin
```

```
##   [1] "tails" "tails" "heads" "heads" "heads" "heads" "tails" "tails" "heads"
##  [10] "tails" "heads" "heads" "heads" "heads" "tails" "tails" "heads" "tails"
##  [19] "tails" "tails" "heads" "heads" "heads" "heads" "tails" "heads" "tails"
##  [28] "heads" "tails" "heads" "tails" "tails" "heads" "tails" "tails" "heads"
##  [37] "tails" "tails" "tails" "tails" "tails" "heads" "heads" "heads" "tails"
##  [46] "heads" "heads" "tails" "heads" "heads" "heads" "tails" "tails" "heads"
##  [55] "heads" "tails" "tails" "tails" "heads" "heads" "heads" "heads" "heads"
##  [64] "heads" "tails" "heads" "heads" "tails" "heads" "tails" "heads" "tails"
##  [73] "tails" "heads" "tails" "heads" "tails" "heads" "tails" "tails" "heads"
##  [82] "heads" "tails" "tails" "heads" "tails" "tails" "heads" "tails" "heads"
##  [91] "heads" "heads" "tails" "tails" "heads" "tails" "heads" "tails" "tails"
## [100] "tails"
```

```
table(sim_fair_coin)
```

```
## sim_fair_coin
## heads tails
##    51    49
```

Since there are only two elements in `coin_outcomes`, the probability that we "flip" a coin and it lands heads is 0.5. Say we're trying to simulate an unfair coin that we know only lands heads 20% of the time. We can adjust for this by adding an argument called `prob`, which provides a vector of two probability weights.

```
sim_unfair_coin <- sample(coin_outcomes, size = 100, replace = TRUE,
                          prob = c(0.2, 0.8))
```

`prob=c(0.2, 0.8)` indicates that for the two elements in the `outcomes` vector, we want to select the first one, `heads`, with probability 0.2 and the second one, `tails` with probability 0.8. Another way of thinking about this is to think of the outcome space as a bag of 10 chips, where 2 chips are labeled "head" and 8 chips "tail". Therefore at each draw, the probability of drawing a chip that says "head"" is 20%, and "tail" is 80%.

3. In your simulation of flipping the unfair coin 100 times, how many flips came up heads? Include the code for sampling the unfair coin in your response. Since the markdown file will run the code, and generate a new sample each time you *Knit* it, you should also "set a seed" **before** you sample. Read more about setting a seed below.

Solution 3:

```
set.seed(110)
sim_unfair_coin <- sample(coin_outcomes, size = 100, replace = TRUE, prob = c(0.2, 0.8))
table(sim_unfair_coin)
```

```
## sim_unfair_coin
## heads tails
##    16    84
```

Answer: 16 flips came up heads in this simulation with a seed of 110.

Simulating the Independent Shooter

Simulating a basketball player who has independent shots uses the same mechanism that you used to simulate a coin flip. To simulate a single shot from an independent shooter with a shooting percentage of 50% you can type

```
shot_outcomes <- c("H", "M")
sim_basket <- sample(shot_outcomes, size = 1, replace = TRUE)
```

To make a valid comparison between Kobe and your simulated independent shooter, you need to align both their shooting percentage and the number of attempted shots.

4. What change needs to be made to the `sample` function so that it reflects a shooting percentage of 45%? Make this adjustment, then run a simulation to sample 133 shots. Assign the output of this simulation to a new object called `sim_basket`.

Solution 4:

```
set.seed(110)
sim_basket <- sample(shot_outcomes, size = 133, replace = TRUE, prob = c(0.45, 0.55))
sim_basket
```

```
##    [1] "H" "M" "H" "H" "H" "M" "H" "H" "H" "H" "H" "M" "M" "H" "H" "M" "M" "M"
##   [19] "H" "M" "H" "H" "H" "M" "H" "M" "M" "H" "H" "H" "H" "M" "M" "M" "H" "M"
##   [37] "H" "H" "M" "H" "H" "H" "M" "H" "H" "H" "M" "M" "H" "M" "H" "H" "M" "H"
##   [55] "M" "M" "M" "H" "M" "M" "H" "H" "H" "H" "M" "M" "M" "M" "M" "M" "H" "H"
##   [73] "H" "M" "H" "M" "M" "H" "M" "H" "M" "M" "H" "H" "H" "M" "M" "M" "M" "M"
##   [91] "M" "M" "H" "H" "M" "M" "H" "M" "M" "M" "M" "M" "H" "H" "M" "M" "M" "M"
##  [109] "M" "H" "M" "H" "M" "M" "H" "H" "H" "M" "M" "H" "M" "M" "H" "H" "M" "H"
##  [127] "M" "H" "H" "M" "M" "H" "M"
```

5. Using `calc_streak`, compute the streak lengths of `sim_basket`, and save the results in a data frame called `sim_streak`.

Solution 5:

```
sim_streak <- calc_streak(sim_basket)
```

6. Describe the distribution of streak lengths. What is the typical streak length for this simulated independent shooter with a 45% shooting percentage? How long is the player's longest streak of baskets in 133 shots? Make sure to include a plot in your answer.
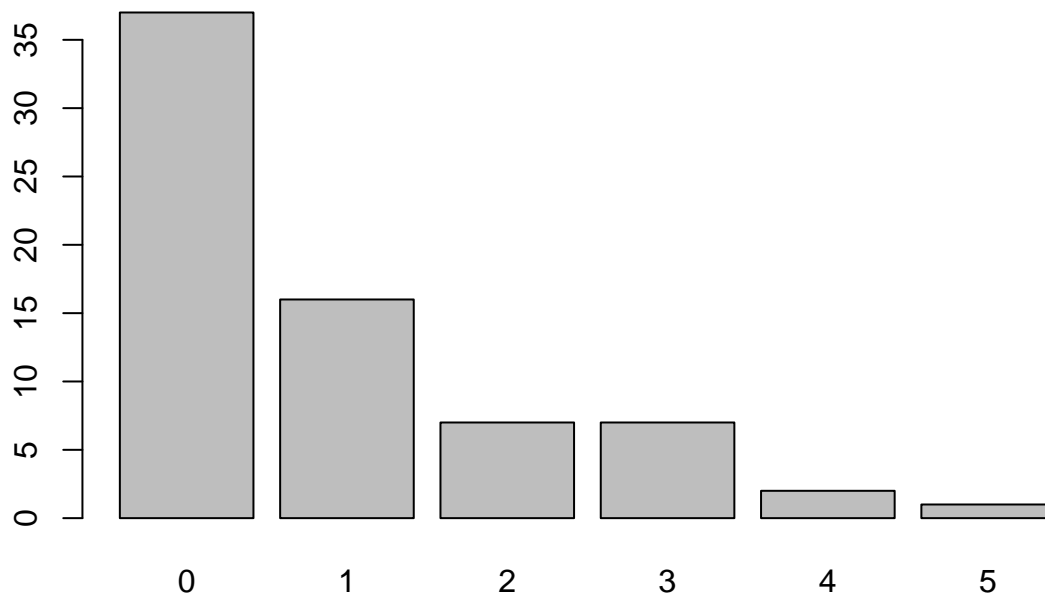
Solution 6:

```r
#Create a table of streaks
(sim_streak_table <- table(sim_streak))
```

```
## sim_streak
##  0  1  2  3  4  5
## 37 16  7  7  2  1
```

#Barplot

```r
barplot(sim_streak_table)
```


#Check the summary

```r
summary(sim_streak)
```

```
##       length
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.9143
##  3rd Qu.:1.0000
##  Max.   :5.0000
```

The distribution is skewed to the right and the player's typical streak is 0. Also, the player's longest streak is 5.

7. If you were to run the simulation of the independent shooter a second time, how would you expect its streak distribution to compare to the distribution from the question above? Exactly the same? Somewhat similar? Totally different? Explain your reasoning.

Solution 7:

The distribution will be the same since we set a seed. If we do not set a seed, the distribution would look somewhat similar but may have different maximum streaks although I think the typical streak would still be 0.

8. How does Kobe Bryant's distribution of streak lengths compare to the distribution of streak lengths for the simulated shooter? Using this comparison, do you have evidence that the hot hand model fits Kobe's shooting patterns? Explain.
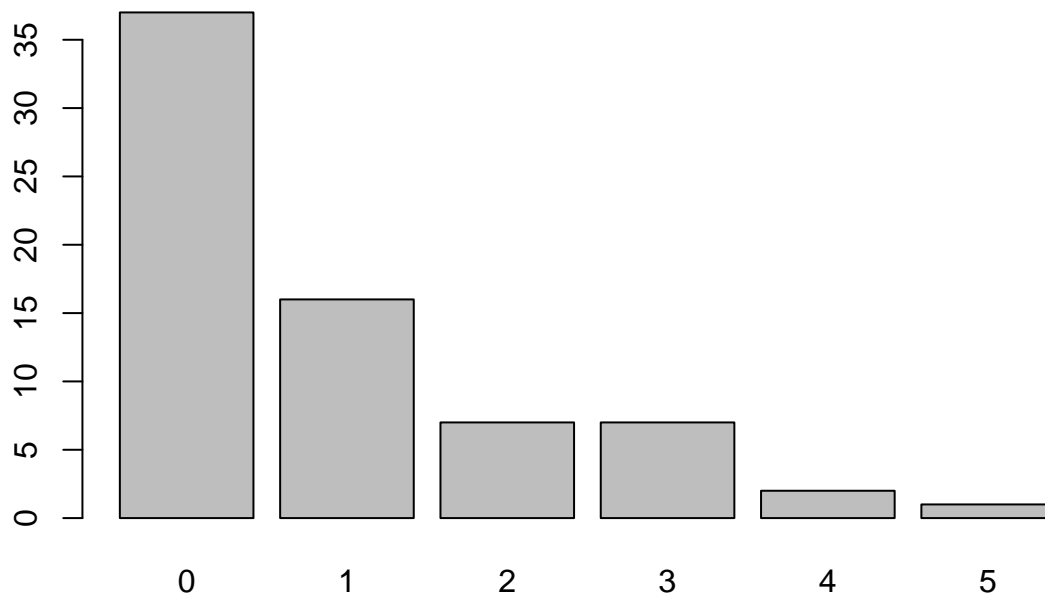
Solution 8:

Kobe barplot:

```
barplot(kobe_streak)
```
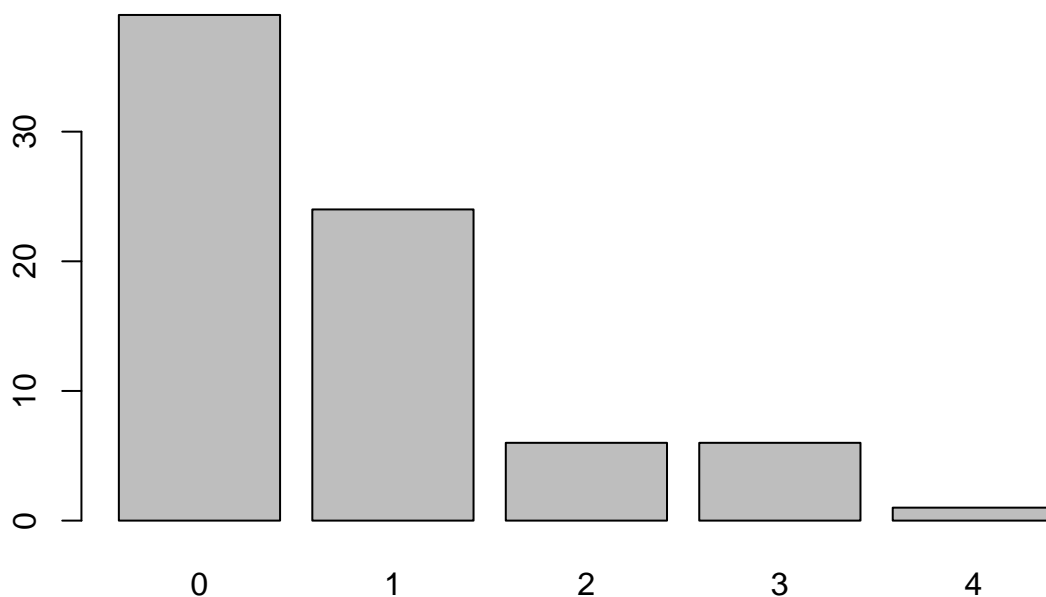

Simulated Shooter barplot:

```
barplot(sim_streak_table)
```

Kobe's distribution is somewhat similar to that of the simulated shooter as both as right skewed although they have different maximum streaks. It can be seen that the most streaks for both Kobe and the simulated shooter is 0, but the simulated shooter has a 5 streak while Kobe did not. Even though both distributions are similar, I cannot conclude that hot hand model fits Kobe's shoot patterns. The simulated shooter had a shooting percentage of 45% while Kobe was random. The similarity in both distributions may just be by chance or may have some relationship. However, the information provided is not sufficient to make any conclusions.