Probability LAB 3

Title: CUNY SPS MDS Data606_LAB3"

Author: Charles Ugiagbe

Date: "9/18/2021"

Getting Started

Load packages

In this lab, we will explore and visualize the data using the `tidyverse` suite of packages. The data can be found in the companion package for OpenIntro labs, **openintro**.

Let's load the packages.

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
```

```
## Warning: package 'tibble' was built under R version 4.1.1
```

```
## Warning: package 'readr' was built under R version 4.1.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(openintro)
```

```
## Warning: package 'openintro' was built under R version 4.1.1
```

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

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

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

Data

Your investigation will focus on the performance of one player: Kobe Bryant of the Los Angeles Lakers. His performance against the Orlando Magic in the 2009 NBA Finals earned him the title *Most Valuable Player* and many spectators commented on how he appeared to show a hot hand. The data file we'll use is called `kobe_basket`.

```
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, 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"~
```

```
head(kobe_basket)
```

```
## # A tibble: 6 x 6
##   vs     game quarter time  description                                    shot
##   <fct> <int> <fct>   <fct> <fct>                                          <chr>
## 1 ORL       1 1       9:47  Kobe Bryant makes 4-foot two point shot        H
## 2 ORL       1 1       9:07  Kobe Bryant misses jumper                      M
## 3 ORL       1 1       8:11  Kobe Bryant misses 7-foot jumper               M
## 4 ORL       1 1       7:41  Kobe Bryant makes 16-foot jumper (Derek Fishe~ H
## 5 ORL       1 1       7:03  Kobe Bryant makes driving layup                H
## 6 ORL       1 1       6:01  Kobe Bryant misses jumper                      M
```

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 length of 1 mean he made one(1) hit within a streak.

There are 1 Hit and I miss in a streak of 1

There are zero(0) Hit and 1 miss in a streak length of 0

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.

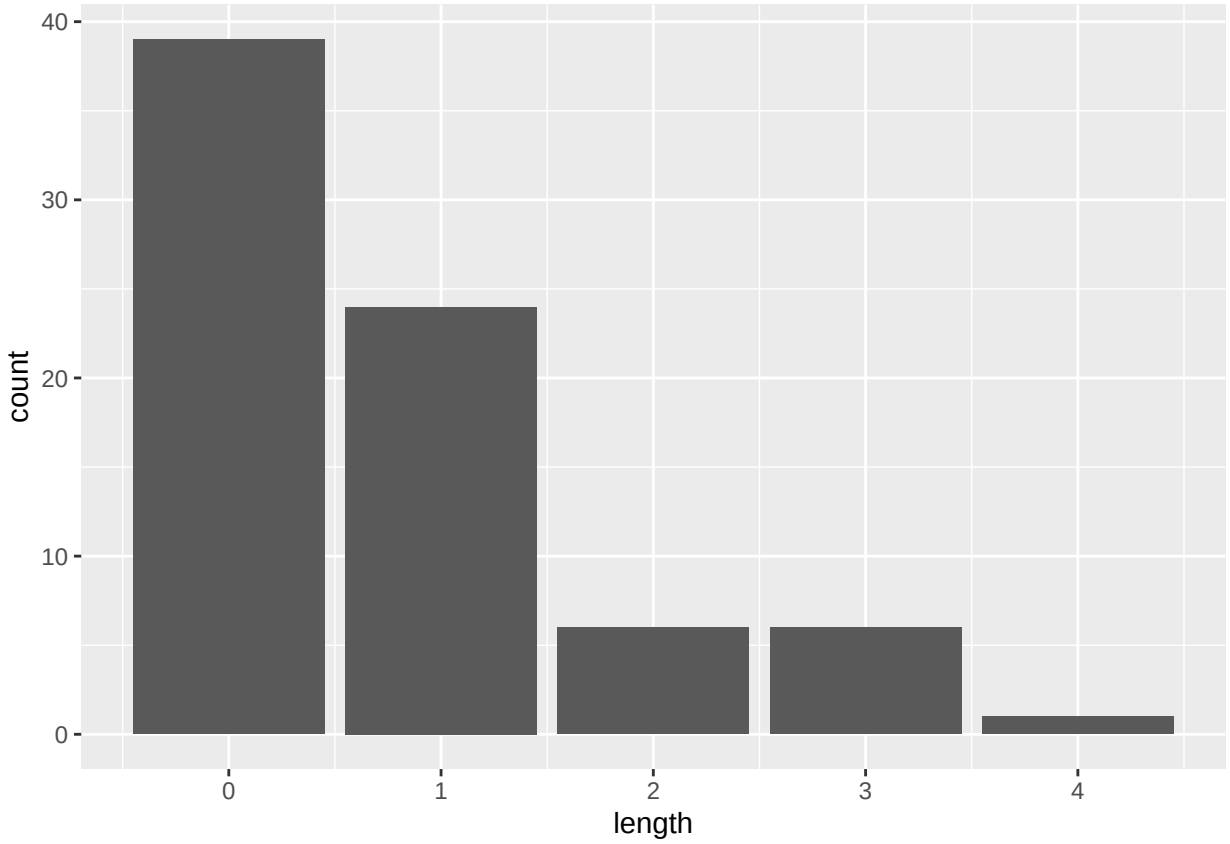
Solution 2:

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

```
table(kobe_streak)
```

```
## kobe_streak
##  0  1  2  3  4
## 39 24  6  6  1
```

```
ggplot(data = kobe_streak, aes(x = length)) +
  geom_bar()
```



The distribution is strongly skewed to the right, suggesting that most of the shots that Kobe took in the 2009 NBA finals were misses.

His typical streak length was 0.

His longest streak of baskets was 4.

Simulations in R

While we don't have any data from a shooter we know to have independent shots, that sort of data is very easy to simulate in R. In a simulation, you set the ground rules of a random process and then the computer uses random numbers to generate an outcome that adheres to those rules. As a simple example, you can simulate flipping a fair coin with the following.

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

```
## [1] "heads"
```

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.

```
table(sim_fair_coin)
```

```
## sim_fair_coin
## heads tails
##    48    52
```

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(229955)
```

In my simulation, heads came up 17 times, tails came up 83 times.

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

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

```
table(sim_unfair_coin)
```

```
## sim_unfair_coin
## heads tails
##    24    76
```

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:

The change the sample function to reflect the new shooting percentage of 45%, we add the Probability funtions of **c(0.45, 0.55)** to the sample function and run the simulation

```
set.seed(295)
```

```
shot_outcomes <- c("H", "M")
sim_basket <- sample(shot_outcomes, size = 133, replace = TRUE, prob = c(0.45, 0.55))
```

```
table(sim_basket)
```

```
## sim_basket
##  H  M
## 67 66
```

More Practice

Comparing Kobe Bryant to the Independent Shooter

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))
```

```
table(sim_streak)
```

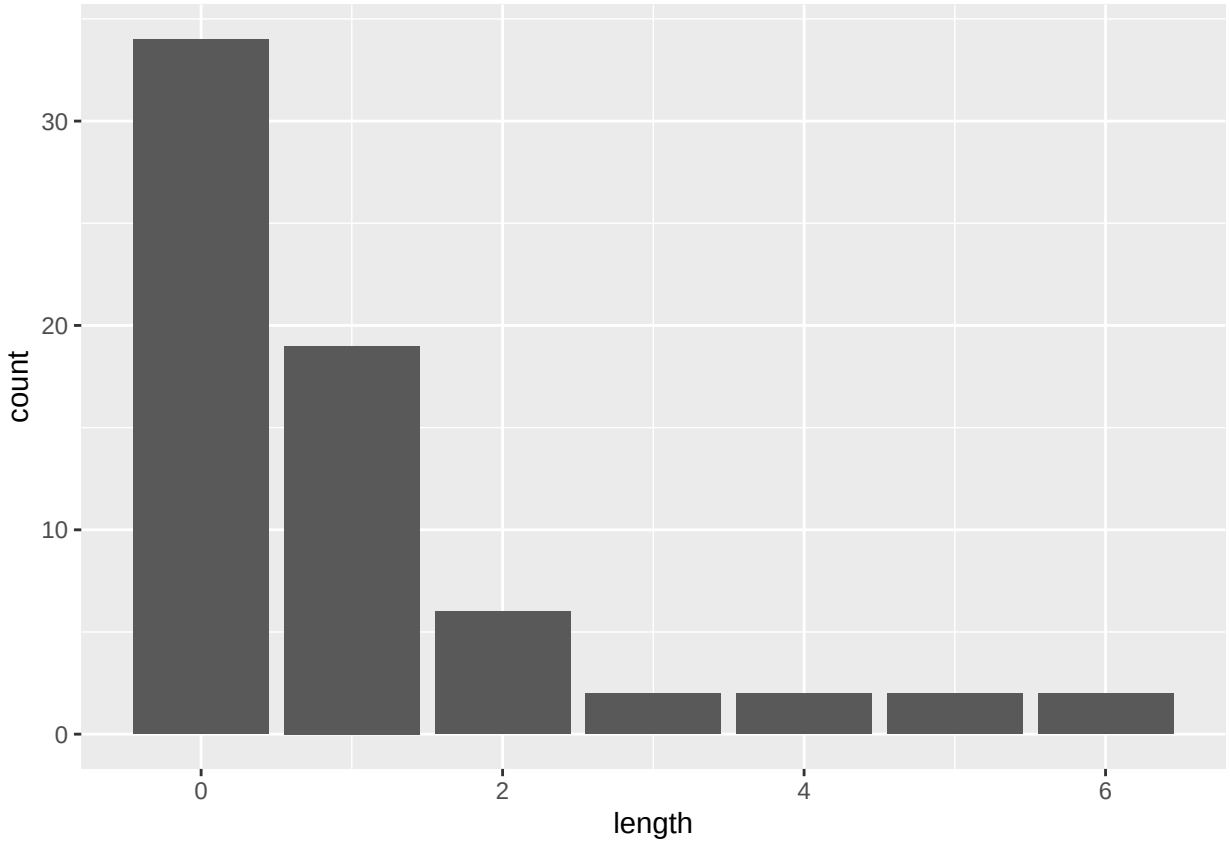
```
## sim_streak
##  0  1  2  3  4  5  6
## 34 19  6  2  2  2  2
```

```
sim_streak1 <- data.frame(table(sim_streak))
```

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:

```
ggplot(data = sim_streak, aes(x = length)) +
  geom_bar()
```



The distribution of the length is strongly skew to the right showing a massive reduction in the count of the length of streak as the streak length increase. Hence there are very few count of multiple streak in the distribution.

The typical streak length for the simulated independent shooter is 0.

The longest streak length for the simulated independent shooter in 133 shots is 6.

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 streak distribution would be relatively similar but not exactly the same Given the low frequencies of high streaks, I would imagine it likely that the range would change. In this simulation, there is a high frequency of 0 length streaks and 1 length streaks and I believe this would be relatively similar.

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:

I observed that the similar rapid drop off in Kobe's streak length was also seen for the independent shooter. This suggests that the probability of consecutive hits decreases in the same way as if the shots were inde-

pendent.

I would argue that the hot hand model does not fit Kobe's shooting patterns because each successive shot is independent of the former.
