

## Lab 2: Probability

### Hot Hands

Basketball players who make several baskets in succession are described as having a “hot hand”. Fans and players have long believed in the hot hand phenomenon, which refutes the assumption that each shot is independent of the next. However, a 1985 paper by Gilovich, Vallone, and Tversky collected evidence that contradicted this belief and showed that successive shots are independent events<sup>†</sup>. This paper started a great controversy that continues to this day, as you can see by Googling “hot hand basketball”.

We do not expect to resolve this controversy today. However, in this lab we hope to show you one approach to problems such as these. The goals for this lab are to (1) think about the effects of independent and dependent events, (2) learn how to simulate shooting streaks in R and to (3) compare a simulation to actual data in order to determine if the hot hand phenomenon appears to be real.

### Saving your code

Click on File → New → R Script. This will open a blank document above the console. As you go along you can copy and paste your code here and save it. This is a good way to keep track of your code and be able to reuse it later. To run your code from this document you can either copy and paste it into the console, highlight the code and hit the Run button, or highlight the code and hit `command+enter`.

You’ll also want to save this document containing your code. To do so click on the disk icon. The first time you hit save RStudio will ask you to give the file a name, you can name it anything you like. Once you hit save you’ll see the file appear under Files. You can reopen this file anytime by simply clicking on it.

### Getting Started

We will focus our investigation 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 of Most Valuable Player and many spectators commented on how he appeared to show a hot hand. Let’s pull up some data from those games and look at the first several rows.

```
download.file("http://www.openintro.org/stat/data/kobe.RData", destfile =  
  "kobe.RData")  
load("kobe.RData")  
head(kobe)
```

---

<sup>†</sup>“The Hot Hand in Basketball: On the Misperception of Random Sequences”, Gilovich, T., Vallone, R., Tversky, A., 1985. *Cognitive Psychology*, 17, pp. 295-314.

In this dataframe, every row records a shot taken by Kobe Bryant. If he made the shot, a hit,  $H$ , is recorded in the column named `basket`, otherwise, a miss,  $M$ , is recorded.

Just looking at the string of hits and misses, it can be difficult to gauge whether or not it seems like Kobe was shooting with a hot hand. One way we can approach this is by considering the fact that hot hand shooters tend to go on shooting streaks. For this lab, we define the length of a shooting streak to be the *number of consecutive baskets made until a miss occurs*.

For example, in Game 1 Kobe had the following sequence of hits and misses from his nine shot attempts in the first quarter:

$H \ M \ | \ M \ | \ H \ H \ M \ | \ M \ | \ M \ | \ M$

To verify this use the following command:

```
kobe[1:9, 6]
```

Within the nine shot attempts, there are six streaks, which are separated by a “|” above. Their lengths are one, zero, two, zero, zero, zero (in order of occurrence).

**Exercise 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?

To calculate the lengths of all of the shooting streaks in the dataset, we can use the custom function `calcStreak()` and then look at the distribution.

```
kobe_streak <- calcStreak(kobe$basket)
barplot(table(kobe_streak))
```

Note that instead of making a histogram, we chose to make a barplot. A barplot is preferable here since our variable is discrete - counts - instead of continuous.

**Exercise 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?

## Compared to What?

We’ve shown that Kobe had some long shooting streaks, but are they long enough to indicate that he had hot hands? What can we compare them to?

To answer these questions, let’s return to the idea of *independence*. Two processes are independent if the outcome of one process doesn’t effect the outcome of the second. So if each shot that a player takes is an independent process, having made or missed your first shot will not effect the probability that you will make or miss your second shot.

A shooter with a hot hand will have shots that are *not* independent of one another. Specifically, if the shooter makes his first shot, he will have a *higher* probability of making his second shot.

Let's assume for a moment that Kobe does have a hot hand. During his career, the percentage of time Kobe makes a basket (i.e. his shooting percentage) is about 45%, or in probability notation,

$$P(\text{shot 1} = H) = 0.45$$

If he makes the first shot and has a hot hand (*not* independent shots), then the probability that he makes his second shot would go up to, let's say, 60%,

$$P(\text{shot 2} = H | \text{shot 1} = H) = 0.60$$

As a result of these increased probabilities, you'd expect Kobe to have longer streaks. Compare this to the situation where Kobe does *not* have a hot hand, where each shot is independent of the next. If he hit his first shot, the probability that he makes the second is,

$$P(\text{shot 2} = H | \text{shot 1} = H) = 0.45$$

In other words, making the first shot did nothing to effect the probability that he'd make his second shot. If Kobe's shots are independent, then he'd have the same probability of hitting every shot regardless of his past shots: 45%.

Now that we've phrased the situation in terms of independent shots, let's return to the question: how do we tell if Kobe's shooting streaks are long enough to indicate that he has hot hands? We can compare his streak lengths to someone without hot hands: an independent shooter.

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

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

The vector `outcomes` can be thought of as a hat with two slips of paper in it: one of these slips says "heads" and the other says "tails". The function `sample()` draws one slip from the hat and tells us if it was a head or a tail.

Run the second line several times. Just like when flipping a coin, sometimes you'll get a heads, sometimes you'll get a tails, but in the long run, you'd expect to get roughly equal numbers of each.

If you wanted to simulate flipping 50 fair coins, you could either run the function 50 times or, more simply, adjust the `size` argument. Also, save the resulting set of heads and tails in a new object called `sim_coin`.

```
sim_coin <- sample(outcomes, size = 50, 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_coin  
table(sim_coin)
```

Since there are only two elements in `outcomes`, the probability that we “flip” a coin and it lands heads is 0.5. Say we're trying to simulate a coin that we know only lands heads 20% of the time. We can adjust for this by updating the `outcomes` vector.

```
outcomes <- as.factor(c("heads", "tails", "tails", "tails", "tails"))
```

Note that we've named the new vector `outcomes`, the same name that we gave to the previous vector with only two elements. In this situation, R overwrites the old object with the new one, so always make sure that you don't need the information in an old vector before reassigning its name.

Go ahead and try sampling from this vector. About what proportion of flips end with heads? Because only one of the five elements in the vector is heads and because each element has an equal chance of being selected, about 20% of the flips should result in heads.

One more thing to say about that `outcomes` vector. If you got tired of typing “tails” over and over again, you can use a function called `rep()`. It will replicate the first argument how ever many times you specify in the second argument.

```
rep("tails", times = 4)
```

The output of this function will be a vector of length 4. Note that you can repeat anything - words or numbers - and feed the output directly into the `c()` function like you did before.

```
outcomes <- c("heads", rep("tails", times = 4))
```

Again, we see it's possible to nest one function inside another. You can read this from the inside out as, “We're creating a vector of four ‘tails’ and combining that into a bigger vector with the word ‘heads’”. If you prefer the step-by-step approach, you can always give each function its own line and bring the output from one line to the next using objects. Also remember that if you want to learn more about `rep()` or any other function, you can always check out its help file.

?rep

## Simulating the Independent Shooter

Simulating a basketball player who has independent shots uses the same mechanism that we use to simulate flipping a coin. To simulate a single shot from an independent shooter with a shooting percentage of 50% we type,

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

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

**Exercise 3** How many  $H$  and  $T$  should be in the `outcomes` vector so that it reflects a shooting percentage of 45%. Make this adjustment then run a simulation using the `sample()` function that samples from this vector 133 times with replacement. Assign the output of this simulation to a new object called `sim_basket`.

With the results of the simulation in our workspace, we have the data necessary to compare Kobe to our independent shooter. We can look at Kobe's data right alongside our simulated data.

```
kobe$basket
sim_basket
```

Both datasets record the results of 133 shot attempts, each with the same shooting percentage of 45%. We know that our simulated data is from a shooter that has independent shots, that is, does not have a hot hand.

**Exercise 4** Would you expect the `sim_basket` you created to have the same sequence of hits and misses as those created by other students in the class? Confirm this by comparing your `sim_basket` to others'.

## On your own

### Comparing Kobe Bryant to the Independent Shooter

1. Calculate the streak lengths of `sim_basket` and describe the streak length distribution. 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?
2. 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.
3. Calculate streak length for the simulated player using the function `calcStreak`. How does Kobe Bryant's distribution of streak lengths from Exercise 2 compare to the simulated distribution from Exercise 3? Using this comparison, do you have evidence that Kobe has a hot hand? Explain.
4. What concepts from the textbook are covered in this lab? What concepts, if any, are not covered in the textbook? Have you seen these concepts elsewhere, e.g. lecture, discussion section, previous labs, or homework problems? Be specific in your answer.

## Notes

This lab was adapted for OpenIntro by Andrew Bray and Mine Çetinkaya-Rundel from a lab written by the faculty and TAs of UCLA Statistics.