

HW 6: NFL/NBA analytics

Stats and sports class

Preliminary notes for doing HW

1. All files should be knit and compiled using R Markdown. Knit early and often! I do not recommend waiting until the end of the HW to knit.
2. All questions should be answered completely, and, wherever applicable, code should be included.
3. If you work with a partner or group, please write the names of your teammates.
4. Copying and pasting of code is a violation of the Skidmore honor code

Part I

HW Grade

Return to Homework 4 and assign yourself a grade:

- 1-3 out of 5 points: Most questions attempted, minimal effort
- 4 of 5 points: All questions attempted, complete effort, graded questions incorrect
- 4.5 of 5 points: All questions attempted, complete effort, graded questions partially correct
- 5 of 5 points: All questions attempted, graded questions perfect

Solutions to HW 5 posted on Github

Part II: Football wrap-up

Readings are up on football.

1. Read **Kickers are Forever** <https://fivethirtyeight.com/features/kickers-are-forever/>
 - How does the model in Footnote 5 compare to the models we fit in class? What does the author say about the primary driver of field goal success?
 - More generally, what does the author say about how field goal success redefines the fourth down bot model?
2. Read **It sucks to kick in the cold**. <https://statsbylopez.com/2016/01/08/it-sucks-to-kick-in-the-cold/>
 - One variable not included in our model of field goal kickers was wind chill. Describe the impact of wind chill on field goal success – be precise, and use numbers from the graph.
 - Assume Kicker A kicks in Chicago, where the wind chill is often cold, while Kicker B kicks in Detroit, which plays indoors, where there is never any wind. In the model we fit in class, assume Kicker A and Kicker B appeared identical. Describe which kicker you think is actually better, now knowing that your model did not include wind chill.

Part III: Basketball

Here, we'll be implementing logistic regression to estimate the probability of successful NBA shots. We'll also link to themes from our football unit - such as expected points added - and increase our visual literacy by sampling some of the `ggplot()` package in R. As one additional tool, I'll walk through a few examples of how we can **clean** what is originally a messy data set.

```
library(tidyverse)
nba_shot <- read_csv("https://raw.githubusercontent.com/JunWorks/NBAstat/master/shot.csv")
```

Data Cleaning

The `nba_shot` data contain roughly 200,000 shots from the 2014-2015 season. This is awesome. However, much like real life, things are never as easy as they seem. Let's start by summarizing our data:

```
head(nba_shot)
summary(nba_shot)
```

1. What does each row refer to, and what are each of the columns?
2. Identify some issues that you see in the data set by looking at the output of the `summary()` command. For example, look for missing values or measurements that, from a basketball perspective, don't make sense.

Interestingly, a handful of shots are missing shot-clock information. There could be several explanations for this - an error in the data collection process, a broken shot clock, etc - and if we had more time, it may be worth exploring why this information is missing.

In the meantime, there are several ways of dealing with missing data - a whole book, in fact - but for today's purposes, we will make some assumptions and drop any rows with missing data. This will make our eventual analysis much easier.

```
nba_shot <- na.omit(nba_shot)
```

Note that dropping missing rows in this data set is more reasonable, given that the number of rows that we dropped accounts for less than 5% of the overall data.

There are some other issues that we should double check. The variable `PTS` contains the number of points that each shot was worth, and the variables `PTS_TYPE` and `SHOT_DIST` indicate the type of shot and distance.

```
nba_shot %>% count(PTS_TYPE)
ggplot(nba_shot, aes(SHOT_DIST)) +
  geom_histogram() +
  facet_wrap(~PTS_TYPE)
nba_shot %>% filter(PTS_TYPE == 3) %>% summarize(min_dist = min(SHOT_DIST))
nba_shot %>% filter(PTS_TYPE == 3, SHOT_DIST < 21) %>% dim()
```

In the second table, there are roughly 1400 shots which are listed as three pointers, even though they were taken from a distance of less than 21 feet. Given that the three-point line is at least 22 feet from the basket, it does not make sense to use these observations.

Here's one way of getting rid of the funny rows.

```
nrow(nba_shot)
nba_shot <- nba_shot %>%
  filter(SHOT_DIST >= 21 | PTS_TYPE == 2)
nrow(nba_shot)
```

3. How many rows were dropped using the above filtering?

Expected points

All else being equal, what's the most efficient shot in the NBA?

Let's start by comparing the success rates of two-point shots to three-point shots.

```
nba_shot %>%  
  group_by(PTS_TYPE) %>%  
  summarise(success_rate = mean(SHOT_RESULT == "made"))
```

4. Identify the expected point totals from all two-point shots and three-point shots in the 2014-15 season. Which one was preferred?

Let's look at certain players.

```
nba_shot %>%  
  filter(playerName == "Stephen Curry") %>%  
  group_by(PTS_TYPE) %>%  
  summarise(success_rate = mean(SHOT_RESULT == "made"))
```

```
nba_shot %>%  
  filter(playerName == "Kevin Garnett") %>%  
  group_by(PTS_TYPE) %>%  
  summarise(success_rate = mean(SHOT_RESULT == "made"))
```

5. For Curry and Garnett, calculate their expected point totals on two and three-point shots. What does that suggest about their optimal choices?

Logistic Regression

Logistic regression will be another useful tool to (i) identify impacts of shooting success and (ii) allow us to judge which players have outperformed or underperformed expectations.

Here's one model.

```
fit_1 <- glm(SHOT_RESULT == "made" ~ SHOT_DIST + TOUCH_TIME +  
            DRIBBLES + SHOT_CLOCK + CLOSE_DEF_DIST,  
            data = nba_shot, family = "binomial")
```

6. Estimate the increased odds of a made shot taken with 1 more second left on the shot clock. Then, estimate the increased odds of a made shot with 10 more seconds on the shot clock.

We use the following to get each shot's expected points (given `fit_1()`), as well as the expected points added (`epa`) given the shot result.

```
nba_shot <- nba_shot %>%  
  mutate(predicted.probs = fitted(fit_1),  
         expected.pts = predicted.probs * PTS_TYPE,  
         epa = PTS - expected.pts)
```

7. Look at the first row of the data set. Where do the `predicted.probs` (0.489), `expected.pts` (0.978), and `epa` (1.02) come from?

It's also possible to look at individual shots based on their expected points. For example, here are the six of the most difficult shots

```
head(arrange(nba_shot, expected.pts))
```

Interestingly, what do we notice about the data set's most difficult shot? It went in!

Here's a video (link).

Alternatively, here are the six shots worth the highest expected point totals.

```
head(arrange(nba_shot, -expected.pts))
```

In each of these examples, the shot is a three-pointer with more than an 80% chance of going in.

8. Looking at six *easy* shots above - can you tell where and why our logistic regression model went wrong?

We use a similar procedure to the one we developed with field goal kickers to estimate the cumulative expected points added from NBA shooters.

```
shot_group <- nba_shot %>%  
  group_by(playerName) %>%  
  summarise(total.epa = sum(epa), n.shots = length(epa)) %>%  
  arrange(total.epa)  
head(shot_group)  
tail(shot_group)
```

For those of us who are basketball fans, these names match our expectations. Stephen Curry was worth nearly 300 points alone, relative to expectation, on his outstanding shooting.

Visualizing expected points

Let's graph the metrics we just calculated. In this example, we'll make our first `ggplot()` graph of the semester (*Note*: `ggplot()` is its own package, but it also comes along when we load `mosaic`).

```
ggplot(shot_group, aes(n.shots, total.epa, label = playerName)) +  
  geom_text() +  
  scale_y_continuous("Expected Points Added") +  
  scale_x_continuous("Shot attempts") +  
  ggtitle("Expected points added ~ number of shots, 2014-15 season") +  
  theme_bw()
```

9. Describe the distribution of expected points added as a function of shot attempts. Why does the distribution fan out? Why does the distribution fan out on only one side?
10. Our model of shot probabilities is probably missing some other variables that effect success rates. What ones can you think of?
11. Returning to the issue from question (8). How does Krishna deal with this problem here?