NFL - expected points and decision making

Michael Lopez, Skidmore College

Overview

In this lab, we'll simulate part of an NFL game.

Formally, we are going to simulate the expected points for NFL teams that begin drives at their own 20-yard line. Recall from Tuesday's lecture, where we learned that possession of the ball at a team's own 20-yard line is worth roughly three-tenths of a point to that team. Given that expointed points are defined as the weighted average of every **next** score, part of the goal of today's lab is to derive what that three-tenths of a point means.

But there's more. On Tuesday, we also identified that coaches sometimes make strange strategic choices on fourth down. In addition to simulating NFL plays, we will also try to identify the effects of suboptimal coaching decisions. As you drive down the field in a simulated drive, some of the class will make optimal decisions, while others will make decisions that NFL coaches currently make.

Bear in mind which coach you are (*this will be assigned by your professor*).

Simulators acting as current coaches will making decisions matching the ones depicted on the right side of this chart. This reflects how coaches are currently calling games. On each fourth down, refer to this chart to either punt, kick a field goal, or go for it.

The other half of the class will make decisions as if they were the 4th down bot, the creation of the New York Times, and one that tends to make more aggressive choices. For reference, see the left side of this chart. On each fourth down (at least for the team starting on offense), refer to this chart to either punt, kick a field goal, or go for it.

Ultimately, using drives starting 20 yards from a team's own end zone, our goal is to estimate the difference in expected points between the Bot's strategy and a traditional coaches' strategy, while also (hopefully) having some fun.

Formalities

Let's load the R packages that we'll need.

```r
library(RCurl)
library(mosaic)
```

Next, we'll load three different data sets. These correspond to field goals (`FG`), punts (`punt`), and traditional offensive plays (`rush.pass`). Each file downloads from the course github page - note that the `rush.pass` file is large, and may take a minute or two to load.

```r
url1<- getURL("https://raw.githubusercontent.com/statsbylopez/StatsSports/master/FG.csv")
url2<- getURL("https://raw.githubusercontent.com/statsbylopez/StatsSports/master/PUNT.csv")
url3<- getURL("https://raw.githubusercontent.com/statsbylopez/StatsSports/master/rush.pass.csv")

FG <- read.csv(text = url1)
punt <- read.csv(text = url2)
rush.pass <- read.csv(text = url3)

head(FG)
head(punt)
head(rush.pass)
```

1. What do you think are the variables in each data set? Check with your professor before moving on.

A final step before getting to the simulation is to load a set of functions that we are going to use. These are also stored on the github page, and can be accessed here.

```
source("https://raw.githubusercontent.com/statsbylopez/StatsSports/master/FourthFunctions.R")
```

The three functions are `Sample.FG`, `Sample.punt`, and `Sample.RP`, which correspond to field goals, punts, and traditional runs and passes. You'll use these three functions next.

Simulate expected points

Each of the files that you just loaded contain plays from the last fifteen years of NFL games. That's a rich data set, and its rich enough that we can resample plays given only a few parameters. Each of the R functions are designed to take certain inputs, link to the appropriate database of real NFL plays, randomly sample from these plays, and output a play result.

Let's start by going through each function.

Runs and passes

First, `Sample.RP` takes each plays down, distance, and yard line (derived as yards from the offensive team's own end zone), and picks a run or a pass in the data set with those corresponding characteristics. For example,

```
set.seed(1)
Sample.RP(1, 10, 20)
```

will randomly sample a 1st-10 play with the ball 20 yards from the offensive team's own end zone. In this case, the offensive team called a pass, picking up 21 yards and recording a first down. (*Note*: The `set.seed()` command fixes the random number generator in R, which ensures we each observe the same correponding play (for now)).

Up next, the offensive team has a 1st-10 with the ball 41 yards from its own end zone. The code you'd want to enter in this case is:

```
Sample.RP(1, 10, 41)
```

On this play, the offensive team picked up 2 yards by running the ball, setting up a 2nd-8, with the ball 43 yards from its end zone.

```
Sample.RP(2, 8, 43)
```

The drive continues until (i) the offensive team or defensive team has recorded a touchdown, (ii) the defensive team takes possession after a turnover or (iii) its fourth down. On fourth down, consult the chart, as well as the following two descriptions, to make the next move.

Punts

Punts (`Sample.punt` function) are a function of the offensive team's `yfog`, and will result in the other team taking possession.

```
Sample.punt(40)
```

Field goals

Second, field goals (`Sample.FG` function) are a function of the offensive team's `yfog`, and will either result in (i) 3 points or (ii) the defensive team taking possession.

```
Sample.FG(77)
```

Outcome

Each set of successive plays defines one interation/simulation of expected points. The outcome we are interested in is the first points scored, relative to the team that started with the ball on offense. So, an outcome of (7) refers to a touchdown scored by the offensive team (even if it comes after the first drive), while an outcome of (-7) refers to a touchdown scored by the team initially starting on defense.

Important note: Simulators using the *4th down bot* should only make optimal decisions when their team has the ball on offense (as they are to start the simulation). When playing offense for the other team, make the more traditional coaching decisions.

Final notes:

Certain plays won't work:

```
Sample.FG(30)
Sample.punt(92)
Sample.RP(4, 30, 50)
```

In each of these cases, there are not enough plays (long field goals, punts from near the opponent's end zone, or long fourth down plays that aren't punts) in the data. If that's the case, see your professor and I can advise. However, my guess is that these plays won't come up often, if at all.

Additionally, note a few assumptions that we are making. First, we assume each play is independent of the one that came beforehand. Second, we assume each team is of even strength. The first assumption is fair, while the second one may be a bit of a stretch.

Next, penalties proved too difficult to include at this iteration. Becuse these events can either help or hurt an offense, hopefully things come close to evening out.

Finally, there are a few types of extreme plays that I discarded. These include kickoffs, blocked kick returns for touchdowns, punt return touchdowns, and punt return turnovers. And there are one or two simplications on traditional runs and passes that discount extreme plays, like those with double turnovers.

In any case, let's have some fun, and let's to pretend to score some points.

Simulation of expected points.

At this point, you and your partner team up with a known coaching strategy. Each time points are scored, record the net total (relative to the initial team on offense) on a separate piece of paper, and start the simulation over.

```
set.seed(2878)   ### Last four digits of your cell phone. Only do this once.
Sample.RP(1, 10, 20)
```

See me if you have any questions - and good luck!