

Lesson 11 Run Expectancy Matrix

Kevin Cummiskey

2/6/2020

Run Expectancy Matrix

An important concept in sabermetrics is the *run expectancy matrix*. For each combination of base runners and outs, it tells us the average number of runs scored in the remainder of the inning. Here is a run expectancy table for 2010-2015.

Base Runners				2010-2015		
1B	2B	3B		0 outs	1 outs	2 outs
—	—	—		0.481	0.254	0.098
1B	—	—		0.859	0.509	0.224
—	2B	—		1.100	0.664	0.319
1B	2B	—		1.437	0.884	0.429
—	—	3B		1.350	0.950	0.353
1B	—	3B		1.784	1.130	0.478
—	2B	3B		1.964	1.376	0.580
1B	2B	3B		2.292	1.541	0.752

What does the run expectancy matrix suggest about the following?

- (1) bunting with no outs and a runner on first.
- (2) intentionally walking a batter with runners on second and third with one out.
- (3) making the first or third out at third base.

Create a run expectancy matrix from retrosheet play-by-play data

The following code is from Chapter 5 of “Analyzing Baseball Data with R” by Marchi, Albert, Baumer.

Get the retrosheet play-by-play data

```
library(tidyverse)

#column names
path = "https://raw.githubusercontent.com/maxtoki/baseball_R/master/data/"
file = "fields.csv"
fields <- read_csv(file = paste(path, file, sep=""))

#I recommend cadets download this file to their computers
#it's too big to keep getting from github
# https://raw.githubusercontent.com/kfcaby/MA388_Sabermetrics/master/data/all2016.csv

data2016 <- read_csv(file = "./data/all2016.csv",
                     col_names = pull(fields, Header),
                     na = character())
```

Runs Scored in the Remainder of the Inning

```
#create some new variables
data2016 %>%
  mutate(RUNS = AWAY_SCORE_CT + HOME_SCORE_CT,
         HALF.INNING = paste(GAME_ID, INN_CT, BAT_HOME_ID),
         RUNS.SCORED =
           (BAT_DEST_ID > 3) + (RUN1_DEST_ID > 3) +
           (RUN2_DEST_ID > 3) + (RUN3_DEST_ID > 3)) ->
  data2016

#compute maximum total score for each half inning
data2016 %>%
  group_by(HALF.INNING) %>%
  summarize(Outs.Inning = sum(EVENT_OUTS_CT),
           Runs.Inning = sum(RUNS.SCORED),
           Runs.Start = first(RUNS),
           MAX.RUNS = Runs.Inning + Runs.Start) ->
  half_innings

#compute runs scored in remainder of the inning (ROI)
data2016 %>%
  inner_join(half_innings, by = "HALF.INNING") %>%
  mutate(RUNS.ROI = MAX.RUNS - RUNS) ->
  data2016
```

Create the Run Expectancy Matrix

```
#create a new variable for the current state
data2016 %>%
  mutate(BASES =
         paste(ifelse(BASE1_RUN_ID > '', 1, 0),
```

```

        ifelse(BASE2_RUN_ID > '',1,0),
        ifelse(BASE3_RUN_ID > '',1,0), sep = ""),
    STATE = paste(BASES, OUTS_CT)) ->
data2016

#NRUNNER1 - indicator if 1st base is occupied after the play
data2016 %>%
  mutate(NRUNNER1 =
    as.numeric(RUN1_DEST_ID==1 | BAT_DEST_ID == 1),
    NRUNNER2 =
    as.numeric(RUN1_DEST_ID == 2 | RUN2_DEST_ID == 2 |
      BAT_DEST_ID == 2),
    NRUNNER3 =
    as.numeric(RUN1_DEST_ID == 3 | RUN2_DEST_ID == 3 |
      RUN3_DEST_ID == 3 | BAT_DEST_ID == 3),
    NOUTS = OUTS_CT + EVENT_OUTS_CT,
    NEW.BASES = paste(NRUNNER1,NRUNNER2, NRUNNER3, sep = ""),
    NEW.STATE = paste(NEW.BASES, NOUTS)) ->
data2016

#only consider plays where the runners on base, outs, or runs scored changed
data2016 %>%
  filter((STATE != NEW.STATE) | (RUNS.SCORED > 0)) ->
data2016

#use only complete half-innings
data2016 %>%
  filter(Outs.Inning == 3) -> data2016Complete

#calculate expected number of runs scored for remainder of inning
#for each bases/outs situation
data2016Complete %>%
  group_by(STATE) %>%
  summarize(Mean = mean(RUNS.ROI)) %>%
  mutate(Outs = substr(STATE,5,5)) %>%
  arrange(Outs) -> RUNS

RUNS_out = matrix(round(RUNS$Mean,2), 8,3)
colnames(RUNS_out) = c("0 outs", "1 out", "2 outs")
rownames(RUNS_out) = c("000","001","010","011",
  "100","101","110", "111")

```

Print Run Expectency Matrix as a table

```
library(knitr)

kable(RUNS_out)
```

	0 outs	1 out	2 outs
000	0.50	0.27	0.11
001	1.35	0.94	0.37
010	1.13	0.67	0.31
011	1.93	1.36	0.55
100	0.86	0.51	0.22
101	1.72	1.20	0.48
110	1.44	0.92	0.41
111	2.11	1.54	0.70

Try It

Copy the code above to get the run expectancy matrix.

Using Git

Instead of cutting and pasting the previous code, I highly recommend cloning my git repository to your computer.

1. Download and install git (<https://git-scm.com/downloads>).
2. Open Windows Powershell.
3. Navigate to a directory where you want the files to go. Use the `ls` and `cd` commands.
4. Clone the git repository.

```
git clone https://github.com/kfcaby/MA388_Sabermetrics.git .
```

5. Go into RStudio and open the R Project. You're ready to go!

If I make changes, you can update the repository later using:

```
git pull origin master
```

I recommend you use another directory for your files so they don't get overwritten.