IMT 573: Module 6 Lab

Conditional Probability

Jenny Skytta

Due: May 8, 2022

Collaborators: Kevin Ruiz

Objectives

Conditional probability is a concept core to modeling data. In this lab exercise, we will work on framing questions in terms of conditional probabilities and computing probabilities to answer those questions. As you work through these questions you will be given opportunities to practice your data manipulation skills, as well as visualization skills. I encourage you to all explain the data anlaysis in written explanations.

Instructions

Before beginning this assignment, please ensure you have access to R and RStudio; this can be on your own personal computer or on the IMT 573 R Studio Cloud.

1. Open the `06_lab_condprob.Rmd` and save a copy to your local directory. Supply your solutions to the assignment by editing `06_lab_condprob.Rmd`.
2. First, replace the "YOUR NAME HERE" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures, and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do no need four different visualizations of the same pattern.
4. Collaboration on problem sets is fun and useful, and I encourage it, but each student must turn in an individual write-up in their own words as well as code/work that is their own. Regardless of whether you work with others, what you turn in must be your own work; this includes code and interpretation of results. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment.
6. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit`. When the PDF report is generated rename the knitted PDF file to `lab6_YourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.

Setup

In this lab you will need, at minimum, the following R packages.

```
# Load standard libraries
library(tidyverse)
library(knitr) # this will keep code on the page!
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
tinytex::install_tinytex()
```

If a baseball team scores X runs, what is the probability it will win the game?

This is the question we will explore in this lab (adapted from Decision Science News, 2014). We will use a dataset of baseball game statistics from 2010-2013.

Baseball is played between two teams who take turns batting and fielding. A run is scored when a player advances around the bases and returns to home plate. More information about the dataset can be found at http://www.retrosheet.org/.

Data files can be found data folder on RStudio Cloud. Load them into one data.frame in R as shown below. Comment this code to demonstrate you understand how it works.

Note: More information about the dataset can be found at http://www.retrosheet.org/

```
# Data can be obtained from http://www.retrosheet.org/ Data
# do not have column names on them. You can obtain column
# names from
# http://www.dangoldstein.com/flash/bball/cnames.txt

# Read in the column names
colNames <- read.csv("data/cnames.txt", header = TRUE)

# Create an empty object to store the data
baseballData <- NULL
for (year in seq(2010, 2013, by = 1)) {
    # Loop through years to get all data
    mypath <- paste("data/GL", year, ".TXT", sep = "")  # Create the path name for the file
    # cat(mypath,'\n') # Tell me what file I am working on
    # Read in the file and bind to data with correct column
    # names
    baseballData <- rbind(baseballData, read.csv(mypath, col.names = colNames$Name))
    baseballData <- as_tibble(baseballData)
}
baseballData
```

```
## # A tibble: 9,716 x 161
##        Date Numberofgame Day   Visitor VisitorLeague VisitorGameNum Home
##       <int>        <int> <chr> <chr>   <chr>                  <int> <chr>
##  1 20100405            0 Mon   MIN     AL                         1 ANA
##  2 20100405            0 Mon   CLE     AL                         1 CHA
##  3 20100405            0 Mon   DET     AL                         1 KCA
##  4 20100405            0 Mon   SEA     AL                         1 OAK
##  5 20100405            0 Mon   TOR     AL                         1 TEX
##  6 20100405            0 Mon   SDN     NL                         1 ARI
```

```
##  7 20100405            0 Mon   CHN     NL                          1 ATL
##  8 20100405            0 Mon   SLN     NL                          1 CIN
##  9 20100405            0 Mon   SFN     NL                          1 HOU
## 10 20100405            0 Mon   COL     NL                          1 MIL
## # ... with 9,706 more rows, and 154 more variables: HomeLeague <chr>,
## #   HomeGameNum <int>, VisitorScore <int>, HomeScore <int>, Outs <int>,
## #   DayorNight <chr>, Completion <chr>, Forfeit <lgl>, Protest <chr>,
## #   ParkID <chr>, Attendance <int>, DurationMinutes <int>,
## #   VisitingLineScores <chr>, HomeLineScores <chr>, Vat.bats <int>,
## #   Vhits <int>, Vdoubles <int>, Vtriples <int>, Vhomeruns <int>, VRBI <int>,
## #   Vsacrificehits <int>, Vsacrificeflies <int>, Vhit.by.pitch <int>,
## #   Vwalks <int>, Vintentionalwalks <int>, Vstrikeouts <int>,
## #   Vstolenbases <int>, Vcaughtstealing <int>, Vgroundedintodoubleplays <int>,
## #   Vawardedfirstoncatcherinterference <int>, Vleftonbase <int>,
## #   Vpitchersused <int>, Vindividualearnedruns <int>, Vteam.arnedruns <int>,
## #   Vwildpitches <int>, Vbalks <int>, Vputouts <int>, Vassists <int>,
## #   Verrors <int>, Vpassed.balls <int>, Vdouble.plays <int>,
## #   Vtriple.plays <int>, Hat.bats <int>, Hhits <int>, Hdoubles <int>,
## #   Htriples <int>, Hhomeruns <int>, HRBI <int>, Hsacrificehits <int>,
## #   Hsacrificeflies <int>, Hhit.by.pitch <int>, Hwalks <int>,
## #   Hintentionalwalks <int>, Hstrikeouts <int>, Hstolenbases <int>,
## #   Hcaughtstealing <int>, Hgroundedintodoubleplays <int>,
## #   Hawardedfirstoncatcherinterference <int>, Hleftonbase <int>,
## #   Hpitchersused <int>, Hindividualearnedruns <int>, Hteam.arnedruns <int>,
## #   Hwildpitches <int>, Hbalks <int>, Hputouts <int>, Hassists <int>,
## #   Herrors <int>, Hpassed.balls <int>, Hdouble.plays <int>,
## #   Htriple.plays <int>, X78 <chr>, X79 <chr>, X80 <chr>, X81 <chr>, X82 <chr>,
## #   X83 <chr>, X84 <chr>, X85 <chr>, X86 <chr>, X87 <chr>, X88 <chr>,
## #   X89 <chr>, X90 <chr>, X91 <chr>, X92 <chr>, X93 <chr>, X94 <chr>,
## #   X95 <chr>, X96 <chr>, X97 <chr>, X98 <chr>, X99 <chr>, X100 <chr>,
## #   X101 <chr>, X102 <chr>, X103 <chr>, X104 <chr>, X105 <chr>, X106 <chr>,
## #   X107 <chr>, ...
```

Select the following relevant columns and create a new local data.frame to store the data you will use for your analysis.

- Date
- Home
- Visitor
- HomeLeague
- VisitorLeague
- HomeScore
- VisitorScore

```
baseball_selected <- baseballData %>%
    filter(HomeLeague == "NL" & VisitorLeague == "NL") %>%
    select(Date, Home, Visitor, HomeLeague, VisitorLeague, HomeScore,
        VisitorScore)
```

Considering only games between two teams in the National League, compute the conditional probability of the team winning given X runs scored, for $X = 0, ..., 10$. Do this separately for Home and Visitor teams.

- Design a visualization that shows your results.

- Discuss what you find.

```
# Probability (Wins | X) = P(Wins (union)  X) / P( X)

#creating descriptive stats
baseball_probs <- baseball_selected %>% #selecting down
  mutate("X" =  abs(HomeScore - VisitorScore)) %>% #adding absolute value of win
  mutate("Winner" = if_else(VisitorScore > HomeScore, "Visitor", "Home")) %>%
  group_by(Winner) %>% # creating winner column by if/else and grouping by winner
  summarise("Wins" = n(), "Mean" = mean(X), "Median" = median(X), "SD" = sd(X))
#summarizing total, mean, median and standard deviations

bball = baseball_selected %>% #filtering by National Leagues
  filter(VisitorLeague=="NL" & HomeLeague=="NL") %>%
  mutate(HW = HomeScore>VisitorScore, #creating col HW and VW by winner
         VW = VisitorScore>HomeScore)

bball=with(bball,data.frame( #pulling scores into Runs column
  Runs=c(HomeScore,VisitorScore),
  outcome=c(HW,VW),#pulling win outcome boolean into outcome column
  Team=c(rep("Home",nrow(bball)),rep("Visitor",nrow(bball)))
)) #pulling team specification into Team column

runs = bball %>%
  group_by(Runs,Team) %>% #grouping by runs, team
  summarise(Probability_Winning=round(mean(outcome),4),
            obs=length(outcome)) #calculating probability by
```

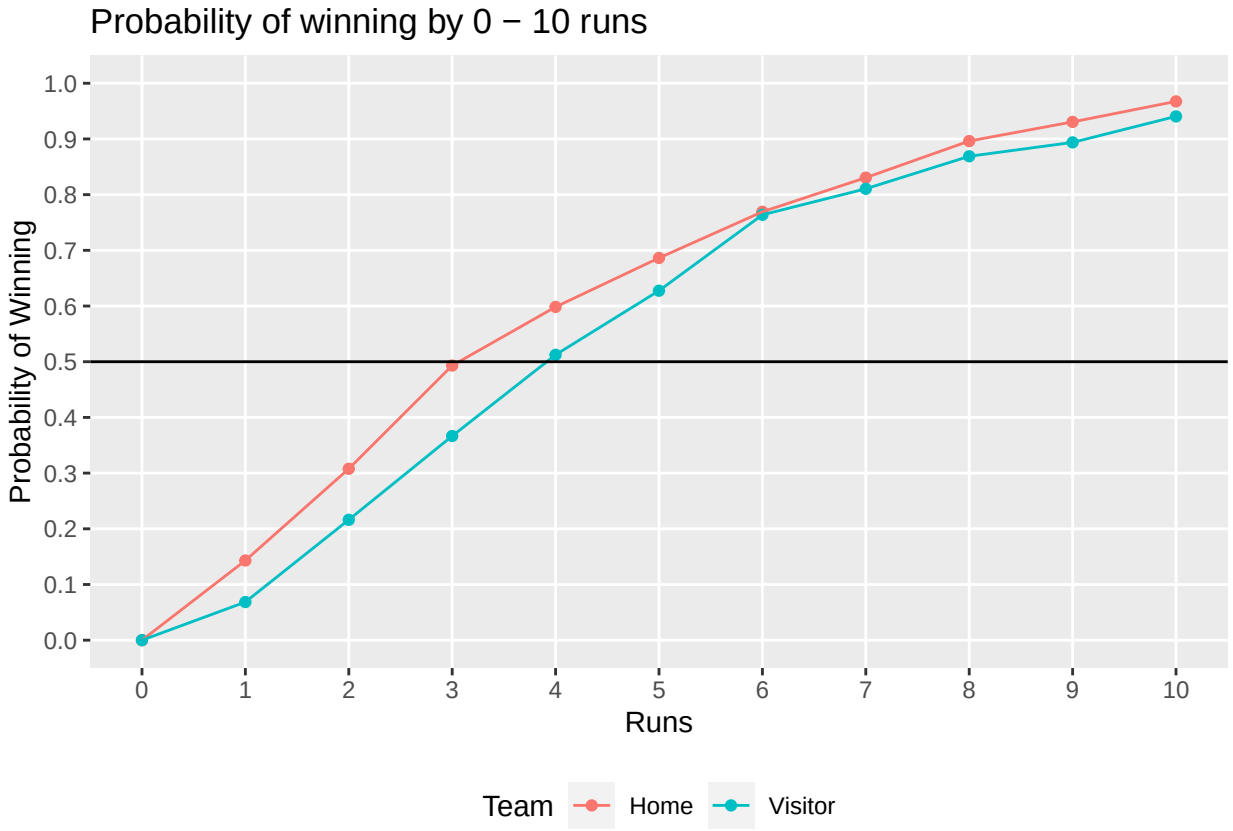
```
## `summarise()` has grouped output by 'Runs'. You can override using the `.groups` argument.
```

```
LIM=11 #set the limit for the plot

runs_plot <- ggplot(subset(runs,Runs<LIM),
                    aes(x=Runs,y=Probability_Winning,group=Team,color=Team)) +
                  geom_point() +
                  geom_line() +
                  theme(legend.position="bottom",panel.grid.minor=element_blank()) +
                  scale_x_continuous(breaks=0:LIM) +
                  scale_y_continuous(limits=c(0,1),breaks=seq(0,1,.1)) +
                  labs(title = "Probability of winning by 0 - 10 runs",x="Runs",
                    y="Probability of Winning") +
                  geom_hline(yintercept=.5)

ggsave(plot = runs_plot,file="runs.png",height=6,width=6)

show(runs_plot)
```



Extra Credit: Repeat the above problem, but now consider the probability of winning given the number of hits.

Citations

Baseball: Probability of winning conditional on runs, hits, walks and errors http://www.decisionsciencenews.com/?p=4755

Code written above is from the previous course IMT 511 which used the below text to support class scripts. https://www.google.com/books/edition/Programming_Skills_for_Data_Science/BnB6DwAAQBAJ?hl=en&gbpv=1&printsec=frontcover —