

Introduction to Sabermetrics Using R

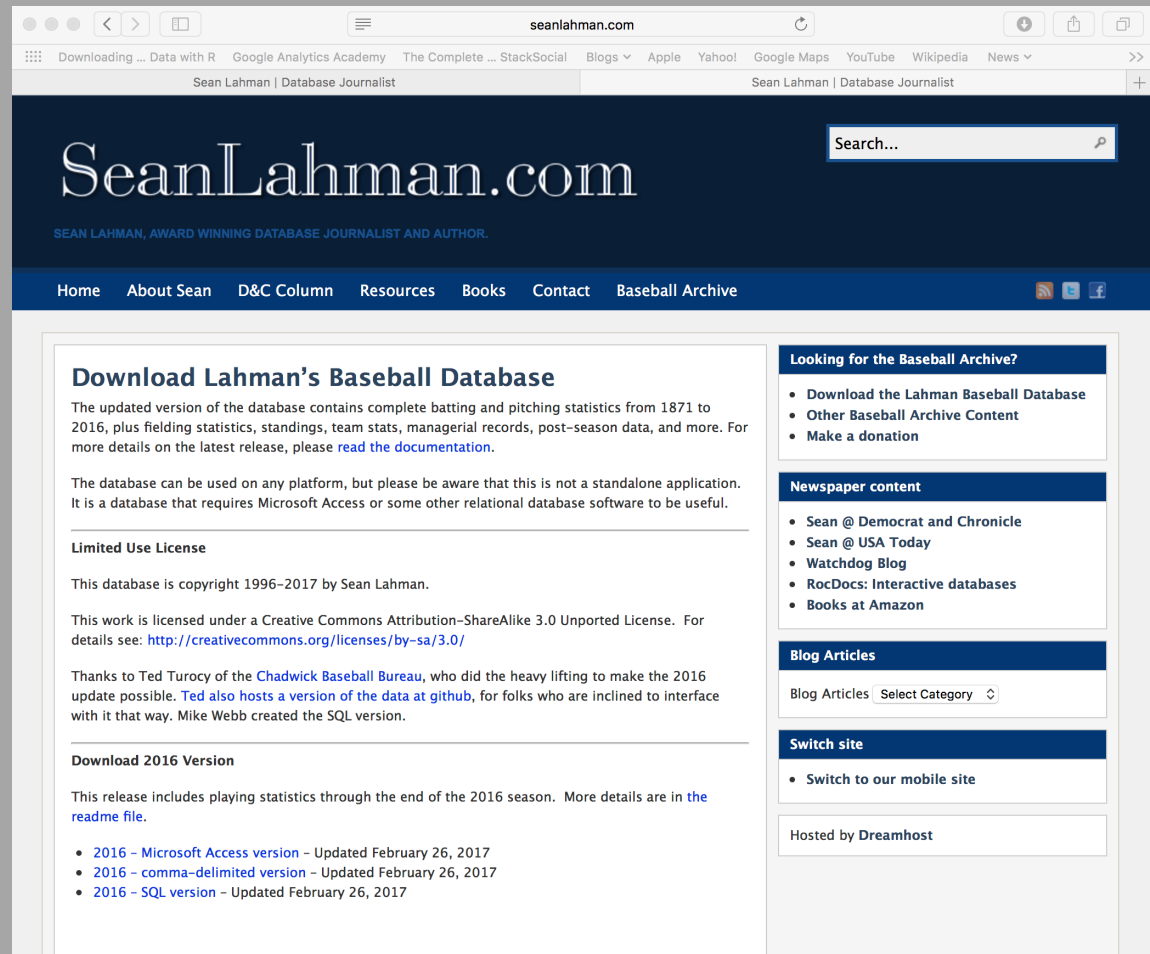
PHIL BANGAYAN
SOCAL CODE CAMP
DECEMBER 3, 2017

Overview

- Introductory talk using R applied to sabermetrics
 - No experience in either topic required
- Cover where to find data and how to process it
 - Lahman season stats to verify the Pythagorean theorem of baseball
 - Retrosheet play-by-play stats to analyze Kershaw's 2015 season
 - PITCHf/x pitch-by-pitch information to look pitch locations from Game 5 of this year's World Series

Lahman Database

- Provides statistics for each player/team by season
- Download by going to <http://seanlahman.com/baseball-archive/statistics>
- Click on “2016 – comma-delimited version”
- Creates folder “baseballdatabank-2017.1”



Lahman Database Tables

- Consists of 24 different tables, with main ones being:
 - Master: list of 19k different players (height, weight, birth, death, IDs)
 - Batting: offensive stats by player by team and year
 - Pitching: pitching stats by player by team and year
 - Teams: records and stats (offensive and defensive) for each team by year
- Documentation on list of tables found in readme2014.txt file
- For this exercise, will look at Teams table

Pythagorean Theorem of Baseball

- Predicts a team's winning percentage using runs scored and runs allowed

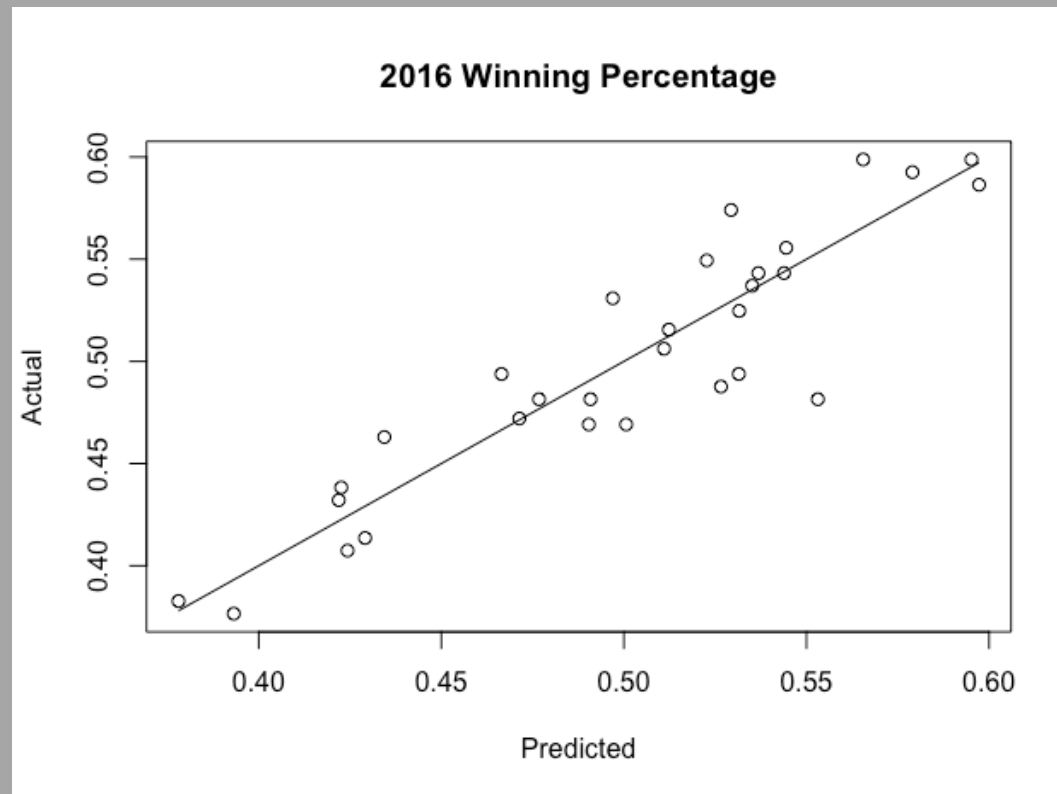
$$\text{Win Pct} = \frac{\text{Wins}}{\text{Wins} + \text{Losses}} \approx \frac{(\text{runs scored})^2}{(\text{runs scored})^2 + (\text{runs allowed})^2}$$

- Use R and the Teams table in Lahman database to verify estimate

Analysis using R

- Copy teams.csv file into R working directory
- Set R working directory appropriately
- Read file using read.csv()
- Examine layout using str()
 - Determine useful fields as teamID, W, L, R, RA
- Limit to 2016 teams using subset()
- Calculate winning percentage, predicted winning percentage, and error
- Calculate error for entire season using mean average deviation (MAD) and root mean square error (RMSE)
- Visualize results

Resulting Plot



Further Analysis

- Why do some teams win more than the predicted value?
 - Examine effect of closers
- What is the effect of scoring an extra xx runs per season?
 - Plug back into formula, using an average value for runs scored
- Is there a better value than 2 for the exponent?
 - Solve by plotting MAD or RMSE for different exponents
 - Or solve for exponent using linear regression

Retrosheet Play-by-Play Data

- Event data for each play of each game in season
- Found at www.retrosheet.org/game.htm
- Go to “Regular Season Event Files” and download “2015” data
- Creates folder “2015eve”
 - Files sorted by home team (e.g. 2015LAN.EVN contains all Events for the National league team Los Angeles played in their home stadium)

Sample Retrosheet Event File

- id, LAN201504060
 - Identifies this section as game in Los Angeles on 2015-04-06, game 0
- info,visteam,SDN
 - Information lines for visiting team, date, time, temperature, umpires, etc.
- start,myerw001,"Wil Myers",0,1,8
 - Starting players, identification, name, visiting team, batting order, position
- play,1,0,myerw001,02,FCH,HP
 - First inning, top half, Wil Myers, on 0-2 count, foul, called strike, hit batter

Pre-processing for Retrosheet Data

- Data needs to be pre-processed before bringing into R
 - Combine all plays from all teams into one file
- Utilize other people's work
 - Chadwick software tools for scoring baseball games
 - Home page at <http://chadwick.sourceforge.net/doc/index.html>
 - Source at <http://sourceforge.net/projects/chadwick/files/>
 - Installs command line tools for parsing event files
 - Analyzing Baseball Data with R function to create CSV from event files
 - <https://baseballwithr.wordpress.com/2014/02/10/downloading-retrosheet-data-and-runs-expectancy/>

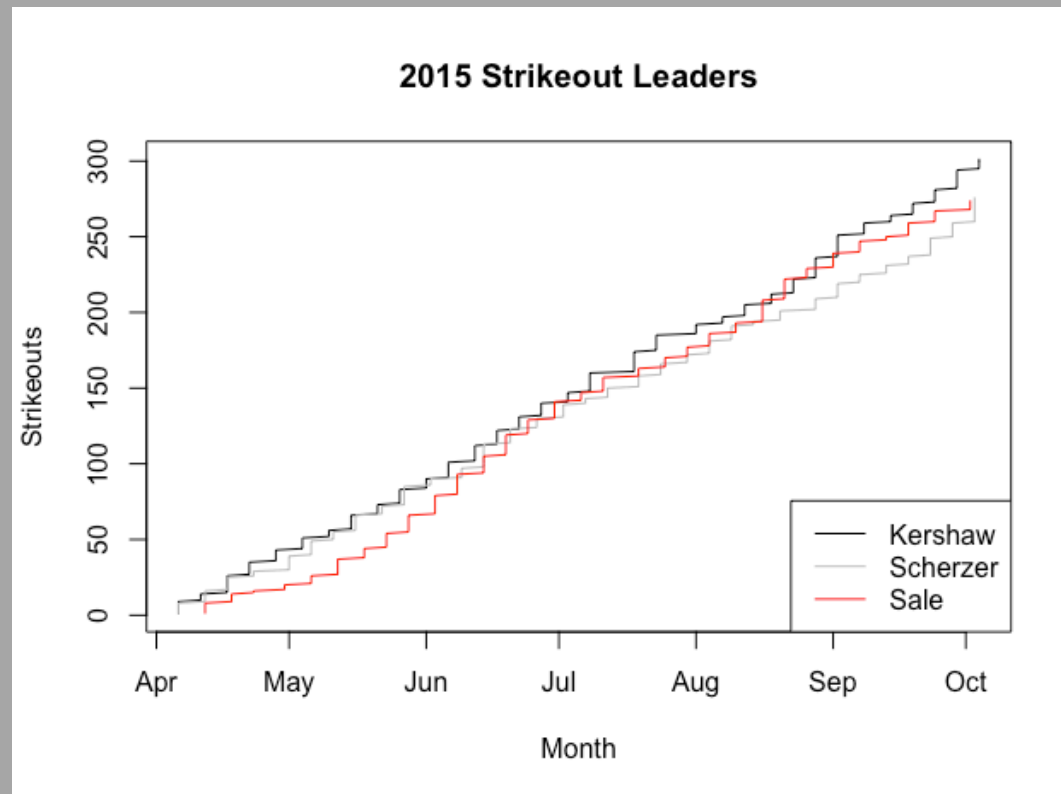
Pre-Processing Steps

- Install Chadwick software
 - Download source
 - Configure, Make, Make install (into /usr/local/lib and /usr/local/bin)
- Create appropriate folders and files
 - “download.folder” (in R current working directory)
 - Subfolders “unzipped” and “zipped” inside “download.folder”
 - “fields.csv” from Analyzing Baseball Data with R github site for headers
- Create data files
 - “source parse.retrosheet2.pbp.R”
 - “parse.retrosheet2.pbp(2015)”

Strikeouts in 2015 Season

- Read in data file generated in pre-processing step
- Limit play-by-play data to strikeouts by Clayton Kershaw
 - Find Kershaw's unique Player ID (kersc001, found in roster2015.csv)
 - Subset by Player ID and Event CD (3 for strikeout)
 - Extract date from unique Game ID
 - Determine cumulative strikeouts
 - Plot cumulative strikeouts by date
- Generalize into a function
- Find cumulative strikeouts for two more players
- Add to existing plot

Resulting Plot

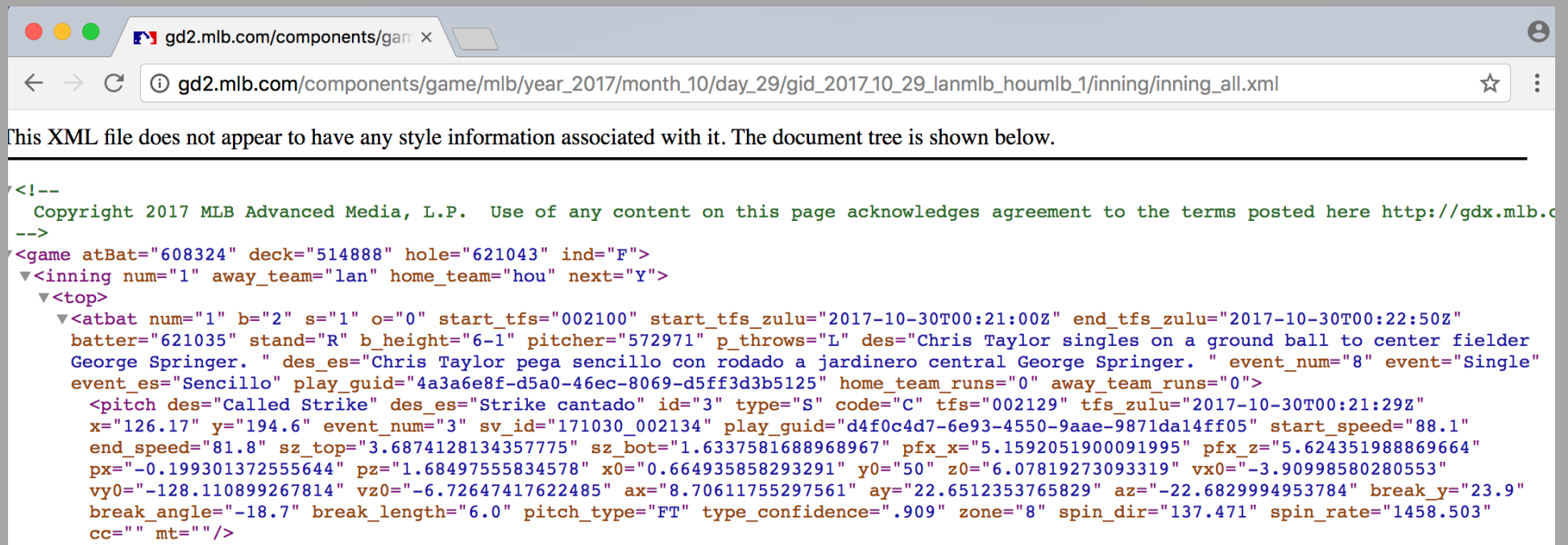


PITCHf/x Database

- Specifics of each pitch of every game since 2008
 - Type of pitch, location, velocity, acceleration, spin, hit location (if hit)
- Found at gd2.mlb.com/components/game/mlb/year_xxxx/month_xx/day_xx
 - Insert numbers for x's above
 - E.g. gd2.mlb.com/components/game/mlb/year_2017/month_10/day_29
- Leads to text webpage with links for all games that day
- Click on desired game, then click on "inning/" and then desired inning (or all)
- Results in XML page describing each pitch

Sample Data

- Just the first pitch from Game 5 of 2017 World Series



```
<!--
Copyright 2017 MLB Advanced Media, L.P. Use of any content on this page acknowledges agreement to the terms posted here http://gdx.mlb.c
-->
<game atBat="608324" deck="514888" hole="621043" ind="F">
  <inning num="1" away_team="lan" home_team="hou" next="Y">
    <top>
      <atbat num="1" b="2" s="1" o="0" start_tfs="002100" start_tfs_zulu="2017-10-30T00:21:00Z" end_tfs_zulu="2017-10-30T00:22:50Z"
        batter="621035" stand="R" b_height="6-1" pitcher="572971" p_throws="L" des="Chris Taylor singles on a ground ball to center fielder
        George Springer. " des_es="Chris Taylor pega sencillo con rodado a jardinero central George Springer. " event_num="8" event="Single"
        event_es="Sencillo" play_guid="4a3a6e8f-d5a0-46ec-8069-d5ff3d3b5125" home_team_runs="0" away_team_runs="0">
          <pitch des="Called Strike" des_es="Strike cantado" id="3" type="S" code="C" tfs="002129" tfs_zulu="2017-10-30T00:21:29Z"
            x="126.17" y="194.6" event_num="3" sv_id="171030_002134" play_guid="d4f0c4d7-6e93-4550-9aae-9871da14ff05" start_speed="88.1"
            end_speed="81.8" sz_top="3.6874128134357775" sz_bot="1.6337581688968967" pfx_x="5.1592051900091995" pfx_z="5.624351988869664"
            px="-0.199301372555644" pz="1.68497555834578" x0="0.664935858293291" y0="50" z0="6.07819273093319" vx0="-3.90998580280553"
            vy0="-128.110899267814" vz0="-6.72647417622485" ax="8.70611755297561" ay="22.6512353765829" az="-22.6829994953784" break_y="23.9"
            break_angle="-18.7" break_length="6.0" pitch_type="FT" type_confidence=".909" zone="8" spin_dir="137.471" spin_rate="1458.503"
            cc="" mt="" />
        </pitch>
      </atbat>
    </top>
  </inning>
</game>
```

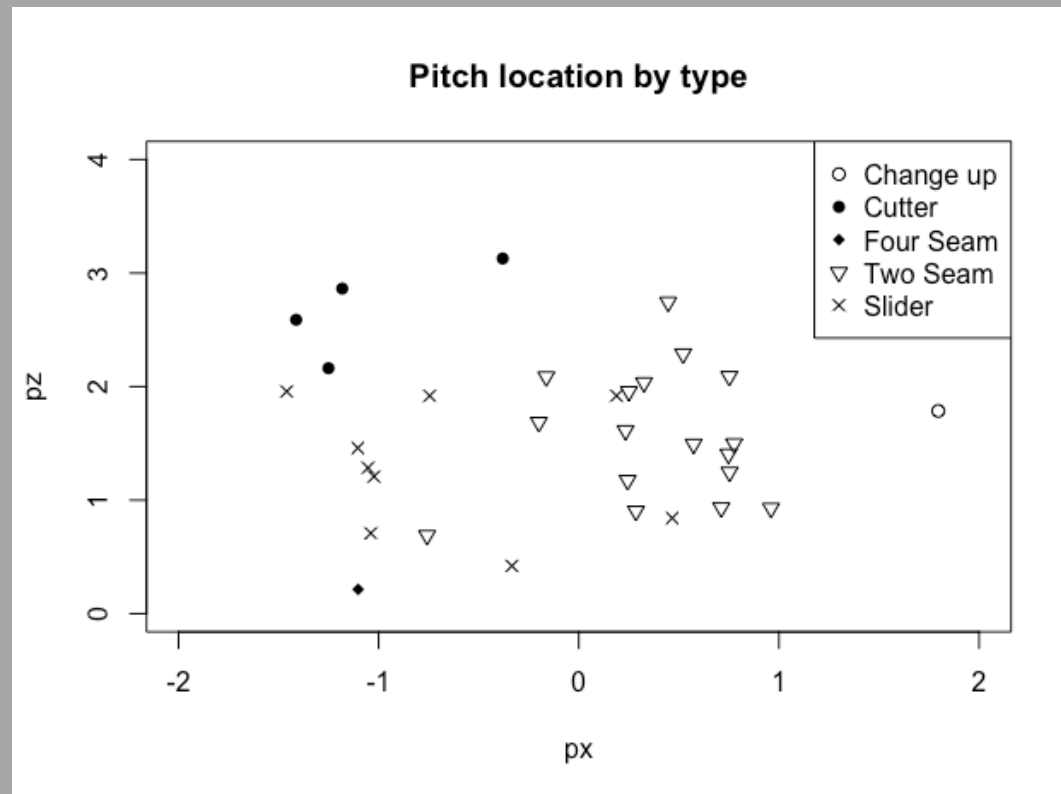

Importing Data into R

- Use XML package, particularly `xmlParse()`
- Helper function `grabXML()`
 - Published in *Analyzing Baseball Data with R*
 - Handles cases of incomplete data
 - Subset either to atbat, pitch, or hip attributes
- Results in data frames interpretable in R

Exercise

- Load Game 5 data (October 29, 2017)
- Focus on top of first inning pitches (n=32)
- Examine types of pitches thrown
 - table() function in R
- Plot location broken out by type of pitches
 - plot() for initial graph
 - points() to add on additional data points

Resulting Plot



Resources

- Websites

- Baseball Reference (www.baseball-reference.com)
- Fan Graphs (www.fangraphs.com)
- Society for American Baseball Research (www.sabr.org)
- MIT Sloan Sports Analytics Conference (sloansportsconference.com)

- Books

- Baseball Hacks by Joseph Adler
- Analyzing Baseball Data with R by Max Marchi, Jim Albert
- The Book by Tom Tango
- Mathletics by Wayne Winston