Assignment 1: Finding and Exploring Sports Data

Due Friday, September 5 11:59 pm

The goal of this assignment is to practice finding, loading, and exploring publicly available sports data. In the first part of the assignment, you will investigate publicly available data for a sport of your choice. In the second part of this assignment you will practice loading and exploring NCAA softball data in R.

Submit your assignment to Gradescope as a PDF file generated from your R Markdown. Make sure your PDF includes all code, outputs, and written answers to questions. All code and outputs used to answer the written questions must be included in the PDF for full credit.

For an introduction to R markdown, follow **this link**. If you have any issues knitting to PDF, check out the troubleshooting steps **linked here**.

Question 1: Exploring Publicly Available Data [15 points, 5 points per dataset]

Choose one of the following sports: American football, soccer, basketball, baseball, softball, or hockey. For the sport of your choosing, find **three** publicly available datasets from *at least 2 different sources* (e.g., at least two different R packages, websites, etc). Do not use Kaggle datasets. For each dataset, answer the following:

- Which dataset did you chose? Give a brief description. Where did you get this dataset? (If you used an R package to get the data, what is the R package called and where does the R package get the data from). Load the data into R and print the first 5 rows.
- What is the unit of observation of the dataset (i.e., what does each row describe)?
- List at least three of the columns in the dataset.
- What time span does the data cover?
- What is a question you could answer using this dataset?

```r
library(Lahman)
head(Batting,5)
```

```
##    playerID yearID stint teamID lgID  G AB R H X2B X3B HR RBI SB CS BB SO IBB
## 1 aardsda01   2004     1    SFN   NL 11  0 0 0   0   0  0   0  0  0  0  0   0
## 2 aardsda01   2006     1    CHN   NL 45  2 0 0   0   0  0   0  0  0  0  0   0
## 3 aardsda01   2007     1    CHA   AL 25  0 0 0   0   0  0   0  0  0  0  0   0
## 4 aardsda01   2008     1    BOS   AL 47  1 0 0   0   0  0   0  0  0  0  1   0
## 5 aardsda01   2009     1    SEA   AL 73  0 0 0   0   0  0   0  0  0  0  0   0
##   HBP SH SF GIDP
## 1   0  0  0    0
## 2   0  1  0    0
## 3   0  0  0    0
## 4   0  0  0    0
## 5   0  0  0    0
```

Dataset 1: I choose the Batting dataset. This dataset provides MLB statistics which are recorded in the Sean 'Lahman' Baseball Database. I got this dataset from the Lahman library in R. The unit of observation in this dataset is each row representing a player's season batting performance. Three columns in this dataset for each entry are playerID, yearID, and the teamID. The library provides data from 1871 to the latest season of availible MLB statistics - currently, 2023. One question you could answer based on this data is how many teams played in the MLB in the 2023 season.

```r
library(retrosheet)
```

```
##
## For Retrosheet data obtained with this package:
##
## The information used here was obtained free of charge from
## and is copyrighted by Retrosheet. Interested parties may
## contact Retrosheet at "www.retrosheet.org"
```

```r
game_log <- get_retrosheet("game", 2023)
head(game_log,5)
```

```
##         Date DblHdr Day VisTm VisTmLg VisTmGNum HmTm HmTmLg HmTmGNum VisRuns
## 1 2023-03-30      0 Thu   MIL      NL         1  CHN     NL        1       0
## 2 2023-03-30      0 Thu   PIT      NL         1  CIN     NL        1       5
## 3 2023-03-30      0 Thu   ARI      NL         1  LAN     NL        1       2
## 4 2023-03-30      0 Thu   NYN      NL         1  MIA     NL        1       5
## 5 2023-03-30      0 Thu   COL      NL         1  SDN     NL        1       7
##   HmRuns NumOuts DayNight Completion Forfeit Protest ParkID Attendance Duration
## 1      4      51        D       <NA>      NA      NA  CHI11      36054      141
## 2      4      54        D       <NA>      NA      NA  CIN09      44063      182
## 3      8      51        N       <NA>      NA      NA  LOS03      52075      155
## 4      3      54        D       <NA>      NA      NA  MIA02      31397      162
## 5      2      54        N       <NA>      NA      NA  SAN02      45103      176
##     VisLine    HmLine VisAB VisH VisD VisT VisHR VisRBI VisSH VisSF VisHBP
## 1 000000000 00400000x    29    4    0    0     0      0     0     0      0
## 2 001300010 100120000    30    6    1    0     1      4     1     1      0
## 3 110000000 00203201x    28    4    1    0     0      2     0     1      1
## 4 001002200 000003000    32    8    1    0     0      5     0     2      1
## 5 100031200 100100000    44   17    4    0     3      7     0     0      0
##   VisBB VisIBB VisK VisSB VisCS VisGDP VisCI VisLOB VisPs VisER VisTER VisWP
## 1     5      0   12     0     0      2     0      7     4     4      4     0
## 2     9      0   11     2     0      1     0      9     5     4      4     0
## 3     0      0    8     0     0      2     0      1     5     8      8     0
## 4     5      0    5     1     0      0     0      8     4     3      3     0
## 5     1      0   17     1     0      0     0     11     4     2      2     0
##   VisBalks VisPO VisA VisE VisPassed VisDB VisTP HmAB HmH HmD HmT HmHR HmRBI
## 1        0    24   12    1         0     1     0   30   6   0   0    0     3
## 2        0    27    9    1         0     2     0   33   7   1   1    1     3
## 3        0    24    7    1         0     0     0   34  12   2   0    1     8
## 4        0    27    8    0         0     2     0   30   5   3   0    1     3
## 5        1    27    9    3         0     2     0   32   7   3   0    0     2
##   HmSH HmSF HmHBP HmBB HmIBB HmK HmSB HmCS HmGDP HmCI HmLOB HmPs HmER HmTER
## 1    0    0     1    4     0   5    0    0     1    0     7    4    0     0
## 2    0    0     0    6     0  15    0    1     1    0     8    6    5     5
## 3    0    1     0    5     0  12    0    0     0    0     8    4    2     2
## 4    0    0     0    2     0  12    0    0     2    0     2    5    5     5
## 5    0    1     0    1     0  10    0    0     2    2     7    4    7     7
##   HmWP HmBalks HmPO HmA HmE HmPass HmDB HmTP   UmpHID        UmpHNm Ump1BID
## 1    1       0   27  13   1      2    2    0 kulpr901     Ron Kulpa blasc901
## 2    1       0   27   7   0      0    1    0 wegnm901   Mark Wegner drecb901
## 3    0       0   27   8   1      0    2    0 hudsm901 Marvin Hudson wendh902
## 4    1       0   27  17   1      0    1    0 vanol901 Larry Vanover guccc901
## 5    2       0   27   8   1      0    0    0 conrc901  Chris Conroy onorb901
##              Ump1BNm  Ump2BID         Ump2BNm  Ump3BID         Ump3BNm UmpLFID
## 1        Cory Blaser torrc901   Carlos Torres viscj901 Jansen Visconti      NA
## 2     Bruce Dreckman sches901 Stu Scheurwater moorm901   Malachi Moore      NA
## 3 Hunter Wendelstedt tumpj901    John Tumpane blakr901    Ryan Blakney      NA
## 4     Chris Guccione rackd901   David Rackley mosce901   Edwin Moscoso      NA
## 5       Brian O'Nora hobep901      Pat Hoberg cejan901     Nestor Ceja      NA
##   UmpLFNm UmpRFID UmpRFNm VisMgrID       VisMgrNm  HmMgrID      HmMgrNm
## 1  (none)      NA  (none) counc001 Craig Counsell rossd001   David Ross
## 2  (none)      NA  (none) sheld801  Derek Shelton belld002   David Bell
## 3  (none)      NA  (none) lovut001   Tony Lovullo robed001 Dave Roberts
## 4  (none)      NA  (none) showb801 Buck Showalter mckej801  Jack McKeon
## 5  (none)      NA  (none) blacb001    Buddy Black melvb001   Bob Melvin
##     WinPID         WinPNm      PID         PNAme  SavePID         SavePNm
## 1 strom001 Marcus Stroman burnc002 Corbin Burnes     <NA>          (none)
## 2 zastr001  Rob Zastryzny farmb001   Buck Farmer bednd001    David Bednar
## 3 uriaj001    Julio Urias gallz001    Zac Gallen     <NA>          (none)
## 4 schem001   Max Scherzer scott003  Tanner Scott robed002 David Robertson
## 5 marqg001 German Marquez snelb001   Blake Snell     <NA>          (none)
##   GWinRBIID      GWinRBINm VisStPchID     VisStPchNm HmStPchID       HmStPchNm
## 1  swand001 Dansby Swanson   burnc002  Corbin Burnes  strom001  Marcus Stroman
## 2  cruzo001     Oneil Cruz   kellm003   Mitch Keller  greeh001   Hunter Greene
## 3  smitw003     Will Smith   gallz001     Zac Gallen  uriaj001     Julio Urias
## 4  nimmb001  Brandon Nimmo   schem001   Max Scherzer  alcas001 Sandy Alcantara
## 5  cronc002      C.J. Cron   marqg001 German Marquez  snelb001     Blake Snell
##   VisBat1ID        VisBat1Nm VisBat1Pos VisBat2ID      VisBat2Nm VisBat2Pos
## 1  yelic001 Christian Yelich          7  winkj002   Jesse Winker         10
## 2  cruzo001       Oneil Cruz          6  reynb001 Bryan Reynolds          7
## 3  lewik001       Kyle Lewis         10  martk001    Ketel Marte          4
## 4  nimmb001    Brandon Nimmo          8  marts002 Starling Marte          9
## 5  dazay001    Yonathan Daza          8  bryak001    Kris Bryant          9
##   VisBat3ID        VisBat3Nm VisBat3Pos VisBat4ID        VisBat4Nm VisBat4Pos
## 1  adamw002     Willy Adames          6  tellr001     Rowdy Tellez          3
## 2  mccua001 Andrew McCutchen         10  santc002   Carlos Santana          3
## 3  gurrl001  Lourdes Gurriel          7  walkc002 Christian Walker          3
## 4  lindf001 Francisco Lindor          6  alonp001      Pete Alonso          3
## 5  blacc001 Charlie Blackmon         10  cronc002        C.J. Cron          3
##   VisBat5ID           VisBat5Nm VisBat5Pos VisBat6ID      VisBat6Nm VisBat6Pos
## 1  contw002   William Contreras          2  urial001     Luis Urias          5
## 2  smitc008 Canaan Smith-Njigba          9  hayek001 Ke'Bryan Hayes          5
## 3  longe001       Evan Longoria          5  ahmen001     Nick Ahmed          6
## 4  mcnej002         Jeff McNeil          4  canhm001     Mark Canha          7
## 5  monte001    Elehuris Montero          5  mcmar001   Ryan McMahon          4
##   VisBat7ID        VisBat7Nm VisBat7Pos VisBat8ID       VisBat8Nm VisBat8Pos
## 1  mitcg001 Garrett Mitchell          8  andeb006  Brian Anderson          9
## 2  suwij001    Jack Suwinski          8  bae-j001     Ji Hwan Bae          4
## 3  carrc005   Corbin Carroll          8  moreg001  Gabriel Moreno          2
## 4  voged001 Daniel Vogelbach         10  escoe001 Eduardo Escobar          5
## 5  diaze005       Elias Diaz          2  casth001   Harold Castro          7
##   VisBat9ID      VisBat9Nm VisBat9Pos HmBat1ID       HmBat1Nm HmBat1Pos
## 1  turab002   Brice Turang          4 hoern001   Nico Hoerner         4
## 2  hedga001  Austin Hedges          2 indij001 Jonathan India         4
## 3  mccaj003  Jake McCarthy          9 bettm001   Mookie Betts         9
## 4  narvo001   Omar Narvaez          2 arral001    Luis Arraez         4
## 5  tovae001 Ezequiel Tovar          6 grist001  Trent Grisham         8
##   HmBat2ID        HmBat2Nm HmBat2Pos HmBat3ID       HmBat3Nm HmBat3Pos HmBat4ID
## 1 swand001  Dansby Swanson         6 happi001       Ian Happ         7 bellc002
## 2 friet001       TJ Friedl         8 fralj001    Jake Fraley        10 stept001
## 3 freef001 Freddie Freeman         3 smitw003     Will Smith         2 muncm001
## 4 seguj002     Jean Segura         5 coopg002 Garrett Cooper         3 chisj001
## 5 sotoj001       Juan Soto         7 machm001  Manny Machado         5 bogax001
##           HmBat4Nm HmBat4Pos HmBat5ID         HmBat5Nm HmBat5Pos HmBat6ID
## 1   Cody Bellinger         8 manct001     Trey Mancini        10 gomey001
## 2 Tyler Stephenson         2 vos lj001     Jason Vosler         3 myerw001
## 3        Max Muncy         5 martj006    J.D. Martinez        10 perad001
## 4    Jazz Chisholm         8 solej001      Jorge Soler        10 garca003
## 5  Xander Bogaerts         6 cronj001 Jake Cronenworth         3 carpm002
##         HmBat6Nm HmBat6Pos HmBat7ID         HmBat7Nm HmBat7Pos HmBat8ID
## 1      Yan Gomes         2 hosme001      Eric Hosmer         3 wisdp001
## 2      Wil Myers         9 stees001    Spencer Steer         5 bensw001
## 3  David Peralta         7 vargm001    Miguel Vargas         4 outmj002
## 4 Avisail Garcia         9 delab001 Bryan De La Cruz         7 stalj001
## 5 Matt Carpenter        10 nolaa002      Austin Nola         2 kim-h002
##          HmBat8Nm HmBat8Pos HmBat9ID          HmBat9Nm HmBat9Pos Additional
## 1  Patrick Wisdom         5 mastm001 Miles Mastrobuoni         9       <NA>
## 2     Will Benson         7 garcj007       Jose Garcia         6       <NA>
## 3    James Outman         8 rojam002      Miguel Rojas         6       <NA>
## 4 Jacob Stallings         2 wendj002       Joey Wendle         6       <NA>
## 5    Ha-Seong Kim         4 dahld001        David Dahl         9       <NA>
##   Acquisition
## 1           Y
## 2           Y
## 3           Y
## 4           Y
## 5           Y
```

Dataset 2: I choose the Game Log dataset from the retrosheet library. This dataset using MLB game-level data is imported from Retrosheet.org which compiles MLB scores. I got this dataset by using the get method from the retrosheet library which allows me to retrieve the data from the website. The unit of observation in this data is a game log represented as an entry with columns representing various attributes about the game. Three columns in this dataset for each entry are Date, Day, and VisTm - the visiting teamID. The dataset covers game logs for the 2023 season. One question that could be answered using this dataset using the DayNight column is how does playing a game at night vs at day affect the performance of the Toronto Blue Jays?

```r
library(baseballr)
mlb_schedule <- mlb_schedule(season = 2025)
head(mlb_schedule,5)
```

```
## -- MLB Schedule data from MLB.com --------------------------- baseballr 1.6.0 --

## i Data updated: 2025-09-04 11:25:51 CDT

## # A tibble: 5 x 71
##   date       total_items total_events total_games total_games_in_progr~1 game_pk
##   <chr>            <int>        <int>       <int>                  <int>   <int>
## 1 2025-02-20           1            0           1                      0  778869
## 2 2025-02-21           6            0           6                      0  779055
## 3 2025-02-21           6            0           6                      0  778780
## 4 2025-02-21           6            0           6                      0  778760
## 5 2025-02-21           6            0           6                      0  778949
## # i abbreviated name: 1: total_games_in_progress
## # i 65 more variables: game_guid <chr>, link <chr>, game_type <chr>,
## #   season <chr>, game_date <chr>, official_date <chr>, is_tie <lgl>,
## #   game_number <int>, public_facing <lgl>, double_header <chr>,
## #   gameday_type <chr>, tiebreaker <chr>, calendar_event_id <chr>,
## #   season_display <chr>, day_night <chr>, scheduled_innings <int>,
## #   reverse_home_away_status <lgl>, inning_break_length <int>, ...
```

Dataset 3: I choose the MLB Schedule dataset from the baseballr library. This dataset is pulled using MLB's API and returns the schedule for the 2025 season. I got this dataset by using the mlb_schedule function from the baseballr library. The unit of observation in this data is a scheduled game represented as an entry with columns representing various attributes about the game. Three columns in this dataset for each entry are Date, teams__home__team__name, and teams__away__team__name. The dataset covers game logs for the 2025 season. One question that could be answered using this dataset is how many Exhibition games will be player in the 2025 season.

Question 2: NCAA Softball Data Analysis [15 points]

In this problem, we will look at NCAA softball hitting data for the 2024 season using the R package `softballR`. We will install the current version of the package available at https://github.com/sportsdataverse/softballR. If this is your first time using `softballR` un-comment and run the lines of code below to install the `softballR` package.

```
# Install devtools if not already installed
# install.packages("devtools")
# library(devtools)
# Install softballR from Github
# devtools::install_github("tmking2002/softballR")
```

Run the code below to load in the hitting data.

```
library(softballR)
hitting <- softballR::load_ncaa_softball_playerbox(
  season = 2024,
  category = "Hitting"
)
```

Question 2a (1 points). What is the unit of observation of the dataset? Each unit of observation is one player(row) as an entry with certain attributes(columns) to represent the player.

```
head(hitting)
```

```
##              player pos g rbi ab r h x2b x3b tb hr ibb bb hbp sf sh k kl dp gdp
## 1    Bickel, Sydney  SS 1   0  3 1 1   0   0  1  0   0  0   0  0  0 0  0  0   0
## 2     Lucas, Rielly  1B 1   1  3 1 1   1   0  2  0   0  0   0  0  0 0  0  0   0
## 3    Gerlach, Bella  LF 1   1  3 0 1   0   0  1  0   0  0   0  0  0 1  0  0   0
## 4 Michallas, Camryn  3B 1   0  3 0 0   0   0  0  0   0  0   0  0  0 1  0  0   0
## 5     Glanz, Reagan  PH 1   0  1 0 0   0   0  0  0   0  0   0  0  0 0  0  0   0
## 6  Ulrich, Brooklyn  2B 1   0  2 0 0   0   0  0  0   0  0   0  0  0 2  0  0   0
##   tp sb cs picked go fo     team opponent game_id  game_date season
## 1  0  0  0      0  0  2 Marshall Penn St. 4472783 02/09/2024   2024
## 2  0  0  0      0  1  1 Marshall Penn St. 4472783 02/09/2024   2024
## 3  0  0  1      0  1  0 Marshall Penn St. 4472783 02/09/2024   2024
## 4  0  0  0      0  2  0 Marshall Penn St. 4472783 02/09/2024   2024
## 5  0  0  0      0  0  1 Marshall Penn St. 4472783 02/09/2024   2024
## 6  0  0  0      0  0  0 Marshall Penn St. 4472783 02/09/2024   2024
```

Question 2b (1 points). What is the range of game dates included in the dataset? The range of game dates included in the dataset is from 02/08/2024 to 05/30/2024.

```
range(hitting$game_date)
```

```
## [1] "02/08/2024" "05/30/2024"
```

Question 2c (2 points). How many unique teams are in the dataset? How many unique players are in the dataset (note: different players that are on different teams might have the same name)? There are 353 unique teams. There are 7752 unique players.

```
length(unique(hitting$team))
```

```
## [1] 353
```

```
length(unique(hitting$player))
```

```
## [1] 7752
```

Question 2d (1 points). The NCAA tournament began on May 17, 2024. Filter your dataset to only include games that occurred before May 17, 2024.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
hitting <- hitting |>
  filter(as.Date(game_date, format = "%m/%d/%Y") < as.Date("05/17/2024", format = "%m/%d/%Y"))

range(hitting$game_date)
```

```
## [1] "02/08/2024"                     "05/12/2024<br/>*If necessary"
```

Question 2e (3 points). Using the filtered dataset from question 2d, create a new dataset that gives the total number of games (g), at bats (ab), and hits (h) for each player. Order this dataset from most to fewest hits and display the first 5 rows. Which player has the most hits? The player with the most hits is Emma Jackson.

```
filt_hitting <- hitting |>
  group_by(player) |>
  summarise(g = sum(g, na.rm = TRUE), ab = sum(ab,na.rm = TRUE), h = sum(h, na.rm = TRUE)) |>
  arrange(desc(h))

head(filt_hitting,5)
```

```
## # A tibble: 5 x 4
##   player                  g    ab     h
##   <chr>               <dbl> <dbl> <dbl>
## 1 Jackson, Emma         158   450   137
## 2 Smith, Gracie         102   325   113
## 3 Grant, Megan           99   291    93
## 4 Trierweiler, Ashley    63   206    92
## 5 Hill, Kelci            58   205    91
```

Question 2f (3 points). Add a new column to the dataset created in question 2e that calculates each player's batting average (total hits divided by total at-bats). Filter out players with fewer than 50 at-bats. Which player has the highest batting average? Why do you think we filter for a minimum number of at-bats before drawing conclusions? CC Wong had the highest batting average at .47. We filter for a minimum number of hits because low sample size may yield high batting averages based on little data. This ensures the batting_avg represents the statistic correctly and not due to random variance.

```
filt_hitting <- filt_hitting |>
  filter(ab >= 50) |>
  mutate(
    batting_avg = h/ab
  ) |>
  arrange(desc(batting_avg))
head(filt_hitting)
```

```
## # A tibble: 6 x 5
##   player                   g    ab     h batting_avg
##   <chr>                <dbl> <dbl> <dbl>       <dbl>
## 1 Wong, CC                50   151    71       0.470
## 2 Otis, Korbe             58   152    71       0.467
## 3 Jordan, Maryn           41   112    52       0.464
## 4 Clements, Jessica       45   153    71       0.464
## 5 Altamirano, Victoria    51   164    75       0.457
## 6 Trierweiler, Ashley     63   206    92       0.447
```

Question 2g (2 points). Create a new filtered dataset that includes only data from April 2024. Which player had the most hits in April 2024? The player with the most hits in April 2024 is Dakota Daniels tied with 4 other players - Kayla Edwards, Halle Hogan, Caitlin Goldwait, and Tavia Leadon - at 5 hits each.

```
hitting_april <- hitting |>
  mutate(game_date = as.Date(game_date, format = "%m/%d/%Y")) %>%
  filter(
    game_date >= as.Date("2024/04/01") &
    game_date <= as.Date("2024/04/30")
  ) |> arrange(desc(h))

head(hitting_april)
```

```
##              player   pos g rbi ab r h x2b x3b tb hr ibb bb hbp sf sh k kl dp
## 1   Daniels, Dakota CF/LF 1   1  5 0 5   0   0  5  0   0  0   0  0  0 0  0  0
## 2    Edwards, Kayla    LF 1   5  5 5 5   1   0 12  2   0  0   0  0  0 0  0  0
## 3      Hogan, Halle    DP 1   1  5 2 5   2   0 10  1   0  0   0  0  0 0  0  0
## 4 Goldwait, Caitlin    DP 1   1  6 2 5   0   0  5  0   0  0   0  0  0 1  0  0
## 5     Leadon, Tavia    3B 1   1  5 0 5   0   0  5  0   0  0   0  0  0 0  0  0
## 6    Belarde, Aliya    2B 1   1  4 2 4   0   1  6  0   0  0   0  0  0 0  0  0
##   gdp tp sb cs picked go fo            team         opponent game_id  game_date
## 1   0  0  3  0      0  0  0     Alabama A&M Mississippi Val. 4513736 2024-04-06
## 2   0  0  1  0      0  0  0 Army West Point       Holy Cross 4507471 2024-04-07
## 3   0  0  0  0      0  0  0            UTEP      Sam Houston 4508538 2024-04-13
## 4   0  0  0  0      0  0  0          Furman          Wofford 4510081 2024-04-20
## 5   0  0  0  0      0  0  0       Grambling      Southern U. 4509818 2024-04-20
## 6   0  0  0  0      0  0  0            Utah         Utah St. 4506674 2024-04-01
##   season
## 1   2024
## 2   2024
## 3   2024
## 4   2024
## 5   2024
## 6   2024
```

Question 2h (2 points). If you are trying to identify the best hitter on a team, would you rather look at total hits or batting average as your primary metric? Explain your reasoning in 1-3 sentences. What might each capture, and what are the limitations? If I was trying to identify the best hitter on a team, I think batting average would be the primary metric to assess. Batting average serves as a metric to encompass the accuracy by calculating how many hits there are relative to times the player has gone to bat(at bat) which is an important factor to consider since it shows overall performance with consideration for how many opportunities each person got. For example, consider two people each with 20 hits but person A got more chances at bat than person B - person A's batting average would be lower cause they took more attempts to hit 20 balls than person B who did it in less, thus the higher batting average.

Extra Credit: Live Data Collection [3 points]

Watch at least 15 minutes of a sports game on TV or in person and record some data live while watching the game. For example, you could do one of the following:

- Watch 1 quarter of a basketball game and collect data on shot attempts, recording the player who made the attempt and whether it was successful
- Watch 15 minutes of a soccer game and collect data on team possessions, recording the team and possession length
- Watch 1 quarter of a football game and collect data on play type, recording the team, play type (run, pass, field goal attempt, punt), and yards gained/lost

```r
df <- data.frame(
  Team1_BAR = c(54, 7, 129, 86, 52, 94, 4, 89, 20),
  Team2_RMD = c(3, 15, 23, 6, 74, 95, 32, 58, 9)
  )
head(df)
```

```
##   Team1_BAR Team2_RMD
## 1        54         3
## 2         7        15
## 3       129        23
## 4        86         6
## 5        52        74
## 6        94        95
```

Include the data you collected, and a brief reflection on the challenges you faced during the data collection process.

The data I collected was for a match of soccer played between FC Barcelona and Real Madrid in the 2025 Copa Del Rey Final. The main challenge faced in collecting this data is the required data to be inputted by the user. The units for the data is in seconds(s). This process requires time to watch the footage and carefully stop the timer when the possession switches team. Due to this process, it is not exact and one of the main challenges caused by this, is human error. Most of the difficulties such as determining exactly when the possession switches is something a computer can automate but the ask tasks may require manual data entry and thinking from a human.