Class 5

STA3038
Statistics and Data Science, SKKU

In this lab, we will explore Sean Lahman's historical baseball database, which contains complete seasonal records for all players on all Major League Baseball teams going back to 1871. These data are made available in R via the `Lahman` package. While domain knowledge may be helpful, it is not necessary to follow the example.

```
#install.packages("Lahman")
library(Lahman)
library(dplyr)
```

Sean Lahman's Baseball Database is not just one dataset. Type `help("Lahman-package")` to get an idea of the data tables available. The batting statistics of players are stored in one table (`Batting`), while information about people (most of whom are players) is in a different table (`Master`).

Every row in the `Batting` table contains the statistics accumulated by a single player during a single stint for a single team in a single year. Thus, a player like Manny Ramirez has many rows in the Batting table.

```
manny <- filter(Batting, playerID == "ramirma02")
```

Exercises

1. How many rows are in the data frame `manny`?

Using what we've learned, we can quickly tabulate Ramirez's most common career offensive statistics. For those new to baseball, some additional background may be helpful. A hit (H) occurs when a batter reaches base safely. A home run (HR) occurs when the ball is hit out of the park or the runner advances through all of the bases during that play. Barry Bonds has the record for most home runs (762) hit in a career. A player's batting average (BA) is the ratio of the number of hits to the number of eligible at-bats. The highest career batting average in major league baseball history of 0.366 was achieved by Ty Cobb-season averages above 0.300 are impressive. Finally, runs batted in (RBI) is the number of runners (including the batter in the case of a home run) that score during that batter's at-bat. Hank Aaron has the record for most career RBIs with 2,297.

```
manny %>% summarize(
  span = paste(min(yearID), max(yearID), sep = "-"),
  numYears = n_distinct(yearID), numTeams = n_distinct(teamID),
  BA = sum(H)/sum(AB), tH = sum(H), tHR = sum(HR), tRBI = sum(RBI)
  )
```

```
##        span numYears numTeams        BA   tH tHR tRBI
## 1 1993-2011       19        5 0.3122271 2574 555 1831
```

Notice how we have used the `paste()` function to combine results from multiple variables into a new variable, and how we have used the `n_distinct()` function to count the number of distinct rows. In his 19-year career, Ramirez hit 555 home runs, which puts him in the top 20 among all Major League players.

However, we also see that Ramirez played for five teams during his career. Did he perform equally well for each of them? Breaking his statistics down by team, or by league, is as easy as adding an appropriate `group_by()` command.

Exercises

2. Display Manny Ramirez's records (as done above), grouped by teams he played for.

We began this lab by filtering the `Batting` table for the player with `playerID` equal to `ramirma02`. How did we know to use this identifier? This player ID is known as a *key*, and in fact, `playerID` is the *primary key* defined in the `Master` table. That is, every row in the Master table is uniquely identified by the value of `playerID`. Thus there is exactly one row in that table for which `playerID` is equal to `ramirma02`.

```
Master %>%
  filter(nameLast  == "Ramirez" & nameFirst == "Manny")
```

```
##    playerID birthYear birthMonth birthDay birthCountry        birthState
## 1 ramirma02      1972          5       30         D.R. Distrito Nacional
##       birthCity deathYear deathMonth deathDay deathCountry deathState
## 1 Santo Domingo        NA         NA       NA         <NA>       <NA>
##   deathCity nameFirst nameLast        nameGiven weight height bats throws
## 1      <NA>     Manny  Ramirez Manuel Aristides    225     72    R      R
##        debut  finalGame  retroID   bbrefID deathDate  birthDate
## 1 1993-09-02 2011-04-06 ramim002 ramirma02      <NA> 1972-05-30
```

The `playerID` column forms a primary key in the `Master` table, but it does not in the `Batting` table, since as we saw previously, there were 21 rows with that player ID. In the `Batting` table, the `playerID` column is known as a *foreign key*, in that it references a primary key in another table. For our purposes, the presence of this column in both tables allows us to link them together. This way, we can combine data from the `Batting` table with data in the `Master` table. We do this with `inner_join()` by specifying the two tables that we want to join, and the corresponding columns in each table that provide the link. Thus, if we want to display Ramirez's name in our previous result, as well as his age, we must join the `Batting` and `Master` tables together.

In particular, we use the variable `yearID` in `Batting` table and the variable `birthYear` in `Master` table to compute Ramirez's ages.

```
Batting   %>%
  filter(playerID  == "ramirma02") %>%
  inner_join(Master, by = c("playerID"  = "playerID"))  %>%
  group_by(yearID) %>%
  summarize(
    Age = max(yearID - birthYear), numTeams = n_distinct(teamID),
    BA = sum(H)/sum(AB), tH = sum(H), tHR = sum(HR), tRBI = sum(RBI)
  )   %>%
arrange(yearID)
```

```
## # A tibble: 19 x 7
##    yearID   Age numTeams     BA    tH   tHR  tRBI
##     <int> <int>    <int>  <dbl> <int> <int> <int>
##  1   1993    21        1 0.170      9     2     5
##  2   1994    22        1 0.269     78    17    60
##  3   1995    23        1 0.308    149    31   107
##  4   1996    24        1 0.309    170    33   112
##  5   1997    25        1 0.328    184    26    88
##  6   1998    26        1 0.294    168    45   145
##  7   1999    27        1 0.333    174    44   165
##  8   2000    28        1 0.351    154    38   122
##  9   2001    29        1 0.306    162    41   125
## 10   2002    30        1 0.349    152    33   107
## 11   2003    31        1 0.325    185    37   104
## 12   2004    32        1 0.308    175    43   130
## 13   2005    33        1 0.292    162    45   144
```

```
## 14   2006    34        1 0.321    144    35   102
## 15   2007    35        1 0.296    143    20    88
## 16   2008    36        2 0.332    183    37   121
## 17   2009    37        1 0.290    102    19    63
## 18   2010    38        2 0.298     79     9    42
## 19   2011    39        1 0.0588     1     0     1
```

Notice that even though Ramirez's age is a constant for each season, we have to use a vector operation (i.e., `max()`) in order to reduce any potential vector to a single number. (Will `min()` or `mean()` produce the same result?)

Which season was Ramirez's best as a hitter? One relatively simple measurement of batting prowess is *OPS*, or On-Base Plus Slugging Percentage, which is the simple sum of two other statistics: On-Base Percentage (OBP) and Slugging Percentage (SLG). The former basically measures the percentage of time that a batter reaches base safely, whether it comes via a hit (`H`), a base on balls (`BB`), or from being hit by the pitch (`HBP`). The latter measures the average number of bases advanced per at-bat (`AB`), where a single is worth one base, a double (`X2B`) is worth two, a triple (`X3B`) is worth three, and a home run (`HR`) is worth four. (Note that every hit is exactly one of a single, double, triple, or home run.) Let's add this statistic to our results and use it to rank the seasons.

```
mannyBySeason <- Batting %>%
  filter(playerID == "ramirma02") %>%
  inner_join(Master, by = c("playerID"  = "playerID")) %>%
  group_by(yearID) %>%
  summarize(
    Age = max(yearID - birthYear), numTeams = n_distinct(teamID),
    BA = sum(H)/sum(AB), tH = sum(H), tHR = sum(HR), tRBI = sum(RBI),
    OBP = sum(H + BB + HBP) / sum(AB + BB + SF + HBP),
    SLG = sum(H + X2B + 2*X3B + 3*HR) / sum(AB)
  ) %>%
  mutate(OPS   = OBP + SLG) %>%
  arrange(desc(OPS))
mannyBySeason
```

```
## # A tibble: 19 x 10
##    yearID   Age numTeams     BA    tH   tHR  tRBI    OBP    SLG   OPS
##     <int> <int>    <int>  <dbl> <int> <int> <int>  <dbl>  <dbl> <dbl>
##  1   2000    28        1 0.351    154    38   122 0.457  0.697  1.15
##  2   1999    27        1 0.333    174    44   165 0.442  0.663  1.11
##  3   2002    30        1 0.349    152    33   107 0.450  0.647  1.10
##  4   2006    34        1 0.321    144    35   102 0.439  0.619  1.06
##  5   2008    36        2 0.332    183    37   121 0.430  0.601  1.03
##  6   2003    31        1 0.325    185    37   104 0.427  0.587  1.01
##  7   2001    29        1 0.306    162    41   125 0.405  0.609  1.01
##  8   2004    32        1 0.308    175    43   130 0.397  0.613  1.01
##  9   2005    33        1 0.292    162    45   144 0.388  0.594  0.982
## 10   1996    24        1 0.309    170    33   112 0.399  0.582  0.981
## 11   1998    26        1 0.294    168    45   145 0.377  0.599  0.976
## 12   1995    23        1 0.308    149    31   107 0.402  0.558  0.960
## 13   1997    25        1 0.328    184    26    88 0.415  0.538  0.953
## 14   2009    37        1 0.290    102    19    63 0.418  0.531  0.949
## 15   2007    35        1 0.296    143    20    88 0.388  0.493  0.881
## 16   1994    22        1 0.269     78    17    60 0.357  0.521  0.878
## 17   2010    38        2 0.298     79     9    42 0.409  0.460  0.870
## 18   1993    21        1 0.170      9     2     5 0.2    0.302  0.502
## 19   2011    39        1 0.0588     1     0     1 0.0588 0.0588 0.118
```

We see that Ramirez's OPS was highest in 2000. But 2000 was the height of the steroid era, when many sluggers were putting up tremendous offensive numbers. As data scientists, we know that it would be more instructive to put Ramirez's OPS in context by comparing it to the league average OPS in each season???the resulting ratio is often called OPS+. To do this, we will need to compute those averages. Because there is missing data in some of these columns in some of these years, we need to invoke the `na.rm` argument to ignore that data.

```
mlb <- Batting %>%
  filter(yearID %in% 1993:2011) %>%
  group_by(yearID) %>%
  summarize(lgOPS =
    sum(H + BB + HBP, na.rm = TRUE) / sum(AB + BB + SF + HBP, na.rm = TRUE)  +
    sum(H + X2B + 2*X3B  + 3*HR,  na.rm = TRUE)  / sum(AB,  na.rm = TRUE))
```

Next, we need to match these league average OPS values to the corresponding entries for Ramirez. We can do this by joining these tables together, and computing the ratio of Ramirez's OPS to that of the league average.

```
mannyRatio <- mannyBySeason %>%
  inner_join(mlb, by = c("yearID" = "yearID")) %>%
  mutate(OPSplus = OPS / lgOPS) %>%
  select(yearID, Age, OPS, lgOPS, OPSplus) %>%
  arrange(desc(OPSplus))
mannyRatio
```

```
## # A tibble: 19 x 5
##    yearID   Age   OPS lgOPS OPSplus
##     <int> <int> <dbl> <dbl>   <dbl>
##  1   2000    28 1.15  0.782   1.48
##  2   2002    30 1.10  0.748   1.47
##  3   1999    27 1.11  0.778   1.42
##  4   2006    34 1.06  0.768   1.38
##  5   2008    36 1.03  0.749   1.38
##  6   2003    31 1.01  0.755   1.34
##  7   2001    29 1.01  0.759   1.34
##  8   2004    32 1.01  0.763   1.32
##  9   2005    33 0.982 0.749   1.31
## 10   1998    26 0.976 0.755   1.29
## 11   1996    24 0.981 0.767   1.28
## 12   1995    23 0.960 0.755   1.27
## 13   2009    37 0.949 0.751   1.26
## 14   1997    25 0.953 0.756   1.26
## 15   2010    38 0.870 0.728   1.19
## 16   2007    35 0.881 0.758   1.16
## 17   1994    22 0.878 0.763   1.15
## 18   1993    21 0.502 0.736   0.682
## 19   2011    39 0.118 0.720   0.163
```

In this case, 2000 still ranks as Ramirez's best season relative to his peers, but notice that his 1999 season has fallen from 2nd to 3rd. Since by definition a league batter has an OPS+ of 1, Ramirez posted 17 consecutive seasons with an OPS that was at least 15% better than the average across the major leagues-a truly impressive feat.

Finally, not all joins are the same. An `inner_join()` requires corresponding entries in both tables. Conversely, a `left_join()` returns at least as many rows as there are in the first table, regardless of whether there are matches in the second table. Thus, an `inner_join()` is bidirectional, whereas in a `left_join()`, the order in which you specify the tables matters.

Ramirez appears in the all-star games for 12 years. By using `left_join()`, for seasons when Ramirez did not play for the all-star, `NA`'s will be returned.

```
mannyAllstar <- AllstarFull %>% filter(playerID == "ramirma02")

mannyBySeason %>%
  left_join(mannyAllstar, by = c("yearID" = "yearID")) %>%
  select(yearID, Age, OPS, GP, startingPos)
```

```
## # A tibble: 19 x 5
##    yearID   Age   OPS    GP startingPos
##     <int> <int> <dbl> <int>       <int>
##  1   2000    28 1.15      0          NA
##  2   1999    27 1.11      1           9
##  3   2002    30 1.10      1           7
##  4   2006    34 1.06      0          NA
##  5   2008    36 1.03      1           7
##  6   2003    31 1.01      0          NA
##  7   2001    29 1.01      1           7
##  8   2004    32 1.01      1           7
##  9   2005    33 0.982     1           7
## 10   1996    24 0.981    NA          NA
## 11   1998    26 0.976     1          NA
## 12   1995    23 0.960     1          NA
## 13   1997    25 0.953    NA          NA
## 14   2009    37 0.949    NA          NA
## 15   2007    35 0.881     1          NA
## 16   1994    22 0.878    NA          NA
## 17   2010    38 0.870    NA          NA
## 18   1993    21 0.502    NA          NA
## 19   2011    39 0.118    NA          NA
```

Exercises

3. In the above code chunk, if `inner_join()` is used in place of `left_join()`, what will be the number of rows of the resulting table?
4. Confirm that Barry Bonds has the record for most home runs (762) hit in a career. For this, list top 20 players' names with the most home runs, and confirm that Manny is in the top 20. Note that you will need to join the `Batting` and `Master` tables together to display the players' name instead of the player ID.
5. Name every pitcher in baseball history who has accumulated at least 300 wins (`W`) and at least 3,000 strikeouts (`SO`). Use `Pitching` table.
6. Display a table with 10 most recent World Series MVP awardees. Include their names and ages. The following code chunk is a good start.

```
AwardsPlayers %>% filter(awardID == "World Series MVP")
```