# DATA 621 - Homework #1

*Cheryl Bowersox, Christopher Martin, Robert Sellers, Edwige Talla Badjio*

*June 19, 2016*

## Contents



# 1 OBJECTIVE

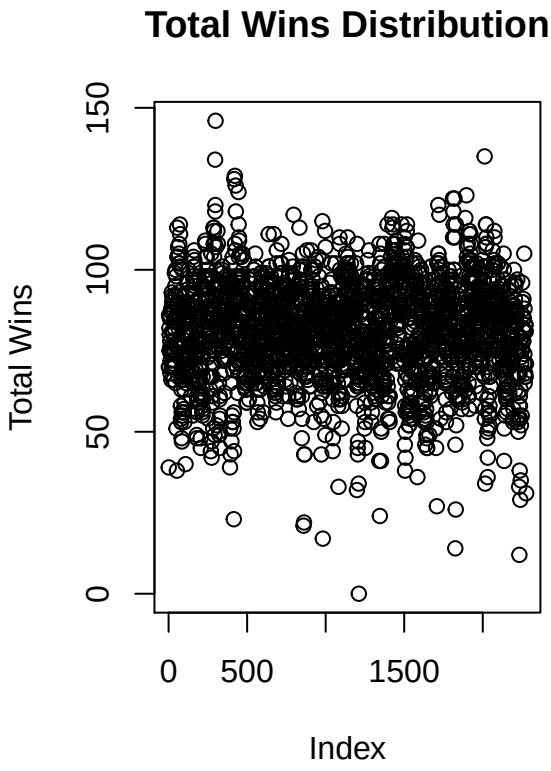
To build an optimal multiple linear regression model for annual number of wins per team based on predictors. The following is a procedure for the selection of three different regression models and the selection of a preferred method. The data source is of data statistics.

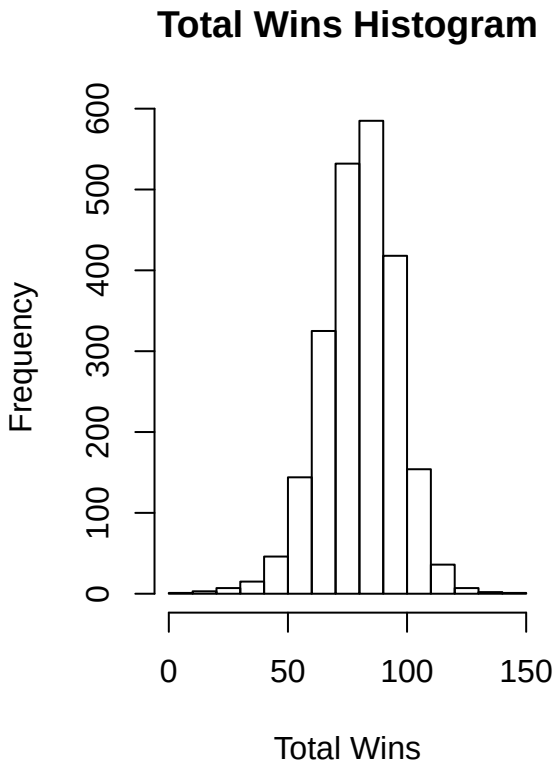
---

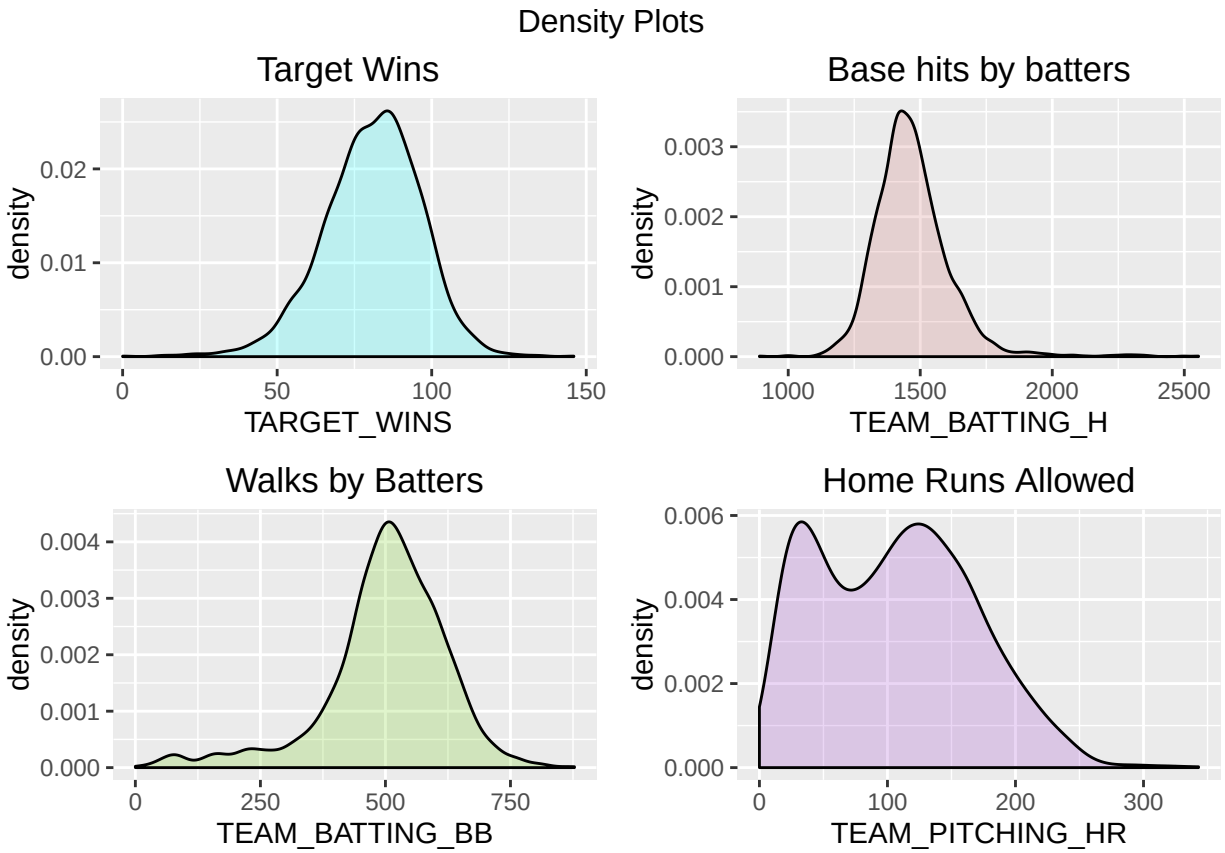
# 2 DATA EXPLORATION

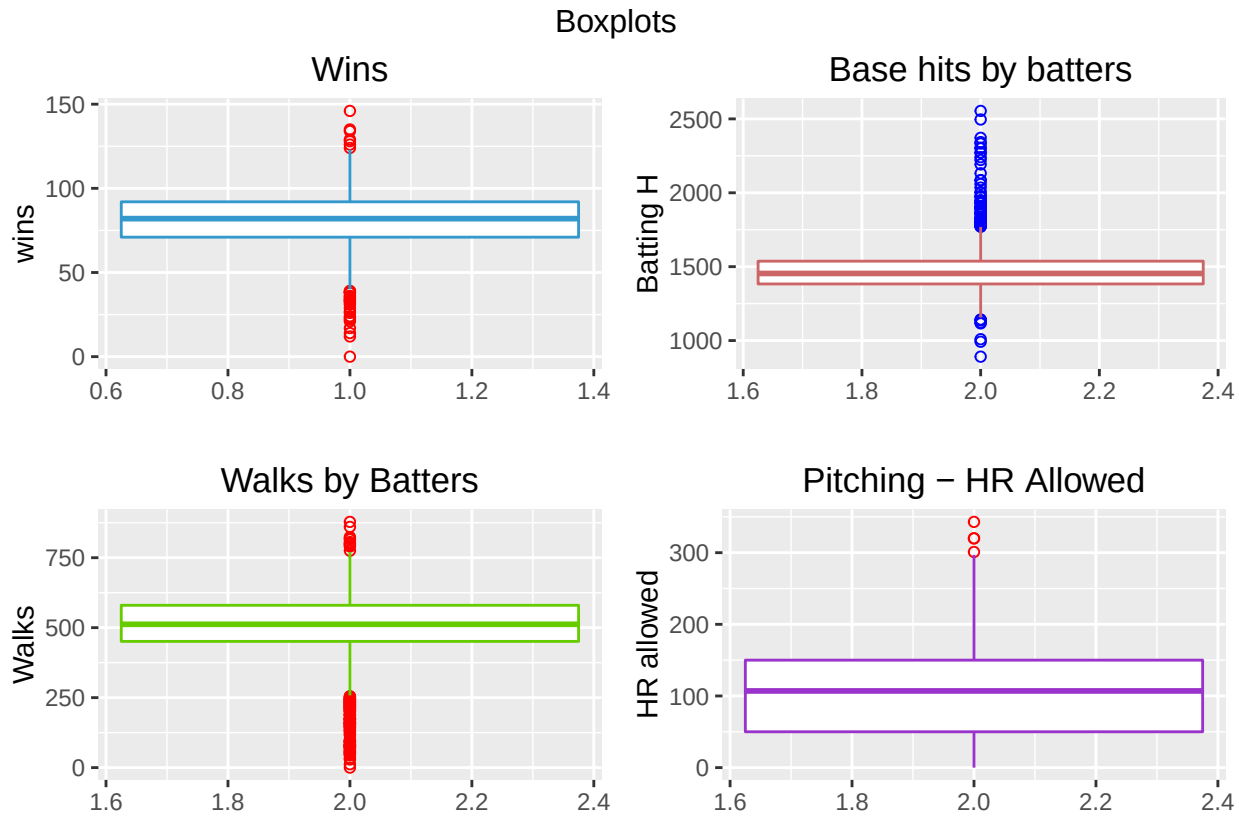
The data has 2276 rows, with each record representing a professional baseball team from the years 1871 to 2006 inclusive. Each record has the performance of the team for the given year, with all of the statistics adjusted to match the performance of a 162 game season.

The response variable is the number of wins, or TARGET_WINS, which has an approximate mean of 80.8 and standard deviation of 15.8. It follows a relatively normal, uniform distribution and we will assume there are no outliers.


Total Wins Distribution
Total Wins
0
50
100
150
0
500
1500
Index


Total Wins Histogram
Frequency
0
100
200
300
400
500
600
0
50
100
150
Total Wins


Density Plots
Target Wins
density
0.02
0.01
0.00
0
50
100
150
TARGET_WINS
Base hits by batters
density
0.003
0.002
0.001
0.000
1000
1500
2000
2500
TEAM_BATTING_H
Walks by Batters
density
0.004
0.003
0.002
0.001
0.000
0
250
500
750
TEAM_BATTING_BB
Home Runs Allowed
density
0.006
0.004
0.002
0.000
0
100
200
300
TEAM_PITCHING_HR




The explanatory/predictor variables are the remaining variables that may have a positive or negative impact on the number of wins.

# 3 DATA PREPARATION

There are several NAs which appear to be values that should have a 0 value (for example, it is perfectly valid for there to be no field double plays in a given match, or no batters hit by pitch in a given match):

```
trainingdata[is.na(trainingdata)] <- 0
#this instead?  Other than this, we just run the models to not include NA?
#trainingdata$TEAM_BASERUN_CS[is.na(trainingdata)] <- 0
```

For columns with values for HBP (hit by pitch) and DP (double plays), the minimum values are >0 and there are a large number of missing values (10% of DP's and 92% of HBP's). While many of these values could very well be missing, it is highly likely that there are 0 values. To simplify this process, a value of 0 has been given to these missing values, while other columns containing missing values (CS, SB, SO) are removed from once the model is run.

```
#Perhaps only changing those without 0 values to 0 values, then removing the rest when the model runs
trainingdata$TEAM_BATTING_HBP[is.na(trainingdata$TEAM_BATTING_HBP)] <- 0
trainingdata$TEAM_FIELDING_DP[is.na(trainingdata$TEAM_FIELDING_DP)] <- 0

#in TEAM_BASERUN_SB there are only 2 0 values (2 matches with no stolen bases), with the next lowest be
trainingdata$TEAM_BASERUN_SB[is.na(trainingdata$TEAM_BASERUN_SB)] <- 0
```

```
#in TEAM_BASERUN_CS there is only 1 0 values (1 match with a caught stealing), with the next lowest bei
trainingdata$TEAM_BASERUN_CS [is.na(trainingdata$TEAM_BASERUN_CS)] <- 0

#in TEAM_BASERUN_SB there is only 1 0 values (1 match with a caught stealing), with the next lowest bei
trainingdata$TEAM_BASERUN_SB[is.na(trainingdata$TEAM_BASERUN_SB)] <- 0

#in TEAM_PITCHING_SO and BATTING_SO there are 20 0 values (20 matches with strikeouts), with the next l
```

Additionally, since (in baseball) the attacking team is the team whose turn it is at bat it might be interesting to see their success rates for those occurrences where a successful hit-at-bat occurs:

```
'add column for total non-homerun hits (all base hits)'
```

```
## [1] "add column for total non-homerun hits (all base hits)"
```

```
trainingdata$TEAM_BATTING_HITSBASE <- trainingdata$TEAM_BATTING_H + trainingdata$TEAM_BATTING_2B + trai
```

```
'add column for all base situations'
```

```
## [1] "add column for all base situations"
```

```
trainingdata$TEAM_BATTING_ALLBASE <- trainingdata$TEAM_BATTING_H + trainingdata$TEAM_BATTING_2B + train

#TEAM_BATTING_H was reduced to become TEAM_BATTING_1B, as the combination of base hits simplifies hits

trainingdata$TEAM_BATTING_1B<-trainingdata$TEAM_BATTING_H- trainingdata$TEAM_BATTING_2B- trainingdata$TE

#TOTAL_BASES'
trainingdata$TOTAL_BASES<-(trainingdata$TEAM_BATTING_2B*2)+(trainingdata$TEAM_BATTING_3B*3)+(trainingda

#add column for all positive batting situations (base + homerun)'
trainingdata$TEAM_BATTING_ALLPOS <- trainingdata$TEAM_BATTING_H + trainingdata$TEAM_BATTING_BB + trainin

#add column for all positive attacking situations (base + homerun + steals)'
trainingdata$TEAM_BATTING_ALLPOSATTCK <- trainingdata$TEAM_BATTING_H + trainingdata$TEAM_BATTING_2B + t
```

There is also another team playing: the defending team. The defending team is mostly effective with the pitcher, so some added variables for pitching could be interesting:

```
'add column for negative pitches (bases)'
```

```
## [1] "add column for negative pitches (bases)"
```

```
trainingdata$TEAM_PITCHING_BASESGIVEN <- trainingdata$TEAM_PITCHING_H + trainingdata$TEAM_PITCHING_BB
```

```
'add column for negative pitching (bases + HRs given)'
```

```
## [1] "add column for negative pitching (bases + HRs given)"
```

```
trainingdata$TEAM_PITCHING_BASESGIVEN <- trainingdata$TEAM_PITCHING_H + trainingdata$TEAM_PITCHING_BB +
```

```
'add column for success in offence vs failures in pitching'
```

```
## [1] "add column for success in offence vs failures in pitching"
```

```
trainingdata$TEAM_POSRATIO <- trainingdata$TEAM_BATTING_ALLPOSATTCK / trainingdata$TEAM_PITCHING_BASESGI
```

Stolen Bases and caught stolen would be better combined into "Stolen Base Percentage".

```
trainingdata$SBPERCENT<-trainingdata$TEAM_BASERUN_SB/
(trainingdata$TEAM_BASERUN_SB+trainingdata$TEAM_BASERUN_CS)
trainingdata$TEAM_BASERUN_SB<-NULL
trainingdata$TEAM_BASERUN_CS<-NULL
```

---

# 4 BUILD MODELS

Current optimal results from stepwise method

```
stepResults
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TOTAL_BASES + TEAM_PITCHING_SO + TEAM_FIELDING_E +
##     TEAM_BATTING_2B + TEAM_PITCHING_H + TEAM_FIELDING_DP + TEAM_BATTING_HR +
##     TEAM_BATTING_ALLBASE + TEAM_POSRATIO + TEAM_PITCHING_BASESGIVEN +
##     SBPERCENT, data = trainingdata)
##
## Coefficients:
##               (Intercept)               TOTAL_BASES
##                 57.237336                   0.063369
##          TEAM_PITCHING_SO           TEAM_FIELDING_E
##                  -0.013188                  -0.070191
##           TEAM_BATTING_2B           TEAM_PITCHING_H
##                  -0.051100                   0.013190
##          TEAM_FIELDING_DP           TEAM_BATTING_HR
##                  -0.045366                  -0.095345
##     TEAM_BATTING_ALLBASE             TEAM_POSRATIO
##                  -0.032853                 -22.249397
## TEAM_PITCHING_BASESGIVEN                  SBPERCENT
##                  -0.009819                   3.043169
```

---

# 5 SELECT MODELS

---

# 6 APPENDIX