NBA Moneyball

Narcel Reedus

October 11, 2017

Summary

NBA Moneyball uses multiple regressions of NBA stats and previous Win/Loss team records to predict the chances of teams reaching the playoffs. The term Moneyball was made popular by the 2011 movie starring Brad Pit as Billy Beane, the coach of the flailing Oakland A's, who used a data analytics approach to winning ball games.

This regression as an .8127 accuracy.

Load libraries into RStudio

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.2

##
## 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
```

```
library(magrittr)
```

```
## Warning: package 'magrittr' was built under R version 3.4.2
```

```
library(ggplot2)
```

Read data

```
NBA_train <- read.csv("C:/Users/narce/OneDrive/Documents/GitHub/NBA/NBA/NBA_train.csv")

NBA <-NBA_train
```

Examine the structure of the data (835 observations and 20 variables)

```
str(NBA)
```

```
## 'data.frame':    835 obs. of  20 variables:
##  $ SeasonEnd: int  1980 1980 1980 1980 1980 1980 1980 1980 1980 1980 ...
##  $ Team     : Factor w/ 37 levels "Atlanta Hawks",..: 1 2 5 6 8 9 10 11 12 13 ...
##  $ Playoffs : int  1 1 0 0 0 0 0 1 0 1 ...
##  $ W        : int  50 61 30 37 30 16 24 41 37 47 ...
##  $ PTS      : int  8573 9303 8813 9360 8878 8933 8493 9084 9119 8860 ...
##  $ oppPTS   : int  8334 8664 9035 9332 9240 9609 8853 9070 9176 8603 ...
##  $ FG       : int  3261 3617 3362 3811 3462 3643 3527 3599 3639 3582 ...
##  $ FGA      : int  7027 7387 6943 8041 7470 7596 7318 7496 7689 7489 ...
##  $ X2P      : int  3248 3455 3292 3775 3379 3586 3500 3495 3551 3557 ...
##  $ X2PA     : int  6952 6965 6668 7854 7215 7377 7197 7117 7375 7375 ...
##  $ X3P      : int  13 162 70 36 83 57 27 104 88 25 ...
##  $ X3PA     : int  75 422 275 187 255 219 121 379 314 114 ...
##  $ FT       : int  2038 1907 2019 1702 1871 1590 1412 1782 1753 1671 ...
##  $ FTA      : int  2645 2449 2592 2205 2539 2149 1914 2326 2333 2250 ...
##  $ ORB      : int  1369 1227 1115 1307 1311 1226 1155 1394 1398 1187 ...
##  $ DRB      : int  2406 2457 2465 2381 2524 2415 2437 2217 2326 2429 ...
##  $ AST      : int  1913 2198 2152 2108 2079 1950 2028 2149 2148 2123 ...
##  $ STL      : int  782 809 704 764 746 783 779 782 900 863 ...
##  $ BLK      : int  539 308 392 342 404 562 339 373 530 356 ...
##  $ TOV      : int  1495 1539 1684 1370 1533 1742 1492 1565 1517 1439 ...
```

Variables definitions:

SeasonEnd: Year the season ended Team: Name of team
Playoffs: Binary variable for playoff appearance
W: Wins (regular season)
PTS: Points scored (regular season)
oppPTS:Oopponent points scored (regular season)
FG, FGA: Field goals (including three pointers)
X2P, X2PA: 2-pointers
X3P, X3PA: 3-pointers
FT, FTA: Free throws
ORB, DRB: Offensive and defensive rebounds
AST: Assists
STL: Steals
BLK: Blocks
*TOV: Turnovers

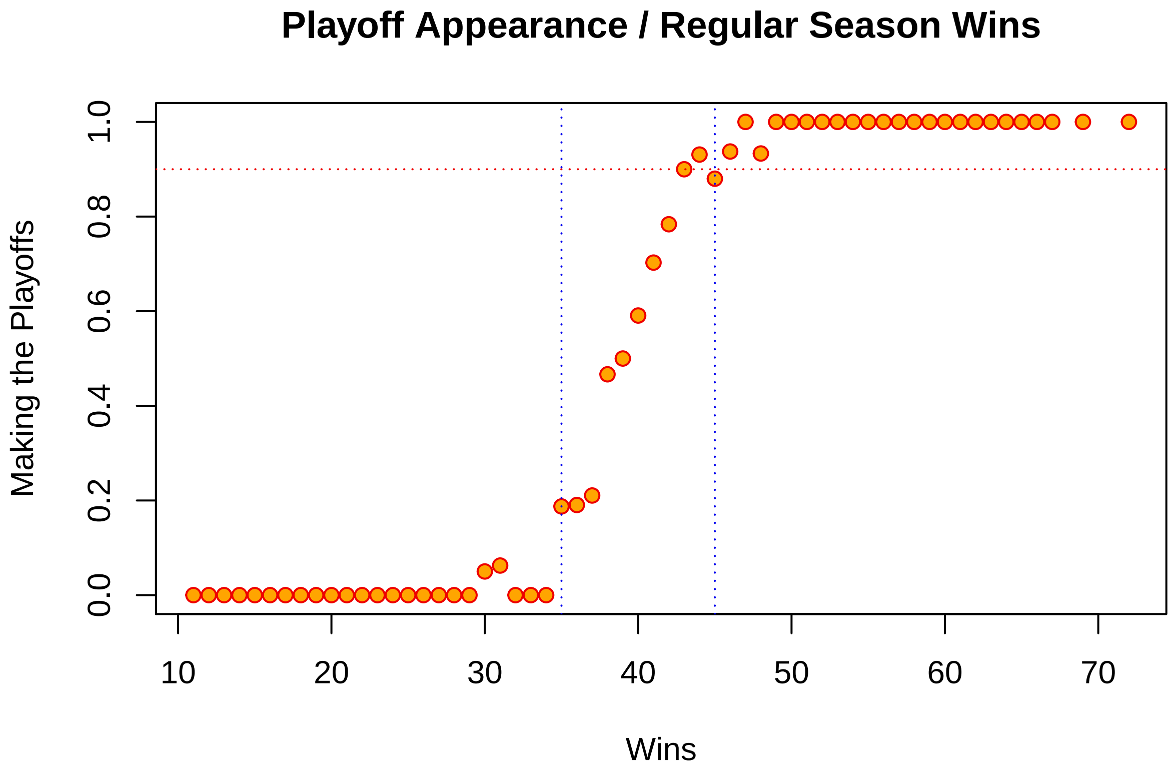
Determine how many regular season wins needed to make playoffs by grouping the data by the number of wins and creating new features: number of playoff appearances and the percentage of playoff appearances (fracPO divided by total number of wins)

```
tmp<- group_by(NBA, W) %>% summarise(nTot = n(), nPO = sum(Playoffs), fracPO = nPO/nTot)

View(tmp)
```

Plot graphs

```
plot(tmp$W, tmp$fracPO, pch = 21, col = "red2", bg = "orange",
     xlab = "Wins", ylab = "Making the Playoffs", main = "Playoff Appearance / Regular Season Wins")
abline(h = 0.9, lty = 3, col = "red2")
abline(v = 35, lty = 3, col = "blue2")
abline(v = 45, lty = 3, col = "blue2")
```



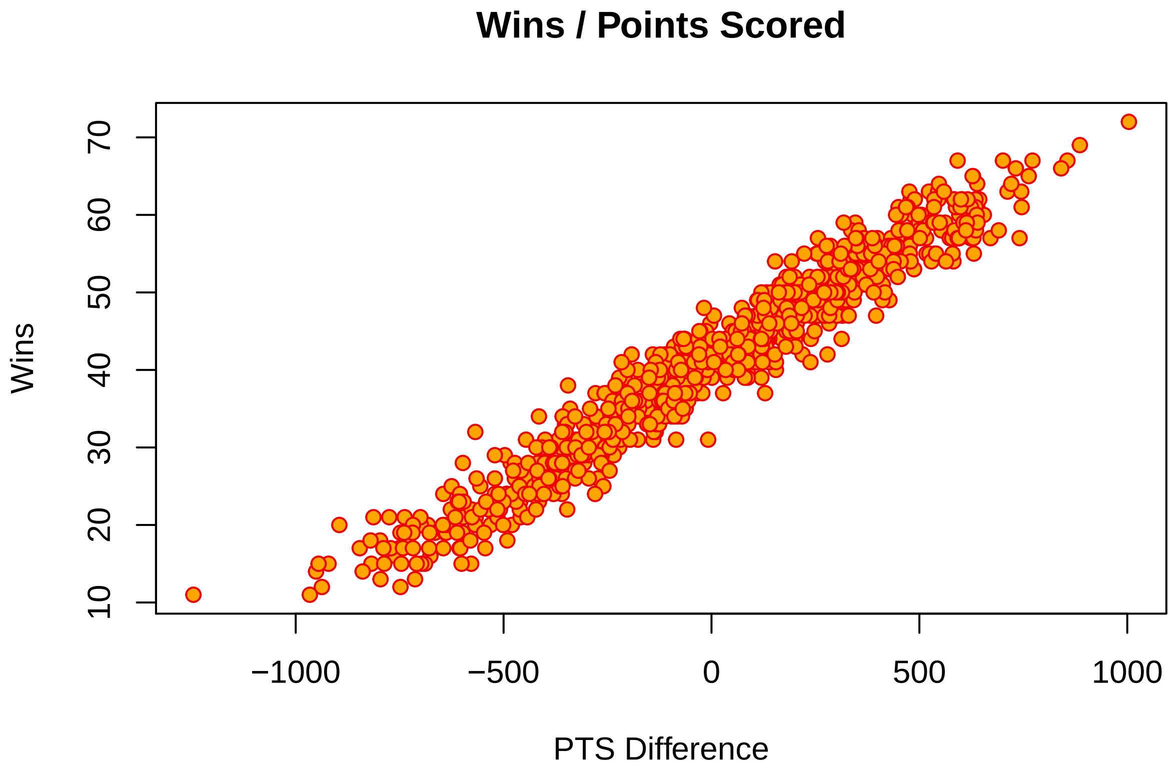
Predictions

Results show that (45>) wins have a (90>) chance of making it to the playoffs Predicting wins by calculating the difference between points scored (PTS) and points allowed (oppPTS)

```
NBA$PTSdiff <- NBA$PTS - NBA$oppPTS
```

Check the linear relationship between PTSdiff and Wins

```
plot(NBA$PTSdiff, NBA$W, pch = 21, col = "red2", bg = "orange",
     xlab = "PTS Difference", ylab = "Wins", main = "Wins / Points Scored")
```



This is the linear regression model for wins: PTSdiff = independent variable and W = dependent variable

```
Wins_Reg <- lm(W ~ PTSdiff, data = NBA)

summary(Wins_Reg)
```

```
##
## Call:
## lm(formula = W ~ PTSdiff, data = NBA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.7393 -2.1018 -0.0672  2.0265 10.6026
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.100e+01  1.059e-01   387.0   <2e-16 ***
## PTSdiff     3.259e-02  2.793e-04   116.7   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.061 on 833 degrees of freedom
## Multiple R-squared:  0.9423, Adjusted R-squared:  0.9423
## F-statistic: 1.361e+04 on 1 and 833 DF,  p-value: < 2.2e-16
```

```
W = 41 + 0.0325*(NBA$PTSdiff)
```

The linear regression model computes the PTSdiff needed attain W=> 45, PTSdiff>122.8

This formula will predict points scored. Dependent variable = PTS, independent variable = (X2PA, X3PA, FTA, ORB, DRB, AST, ST, BL, TOV)

```
PointsRS <- lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + DRB + TOV + STL + BLK, data = NBA)

summary(PointsRS)
```

```
##
## Call:
## lm(formula = PTS ~ X2PA + X3PA + FTA + AST + ORB + DRB + TOV +
##     STL + BLK, data = NBA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -527.40 -119.83    7.83  120.67  564.71
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.051e+03  2.035e+02 -10.078   <2e-16 ***
## X2PA         1.043e+00  2.957e-02  35.274   <2e-16 ***
## X3PA         1.259e+00  3.843e-02  32.747   <2e-16 ***
## FTA          1.128e+00  3.373e-02  33.440   <2e-16 ***
## AST          8.858e-01  4.396e-02  20.150   <2e-16 ***
## ORB         -9.554e-01  7.792e-02 -12.261   <2e-16 ***
## DRB          3.883e-02  6.157e-02   0.631   0.5285
## TOV         -2.475e-02  6.118e-02  -0.405   0.6859
## STL         -1.992e-01  9.181e-02  -2.169   0.0303 *
## BLK         -5.576e-02  8.782e-02  -0.635   0.5256
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 185.5 on 825 degrees of freedom
## Multiple R-squared:  0.8992, Adjusted R-squared:  0.8981
## F-statistic: 817.3 on 9 and 825 DF,  p-value: < 2.2e-16
```

The summary shows significant correlation between X2PA, X3PA, FTA, AST, ORB and Wins (each having 3 stars). Independent variables DRB, TOV, STL, BLK do not show a strong correlation to determining Win R^2 value = 0.8992 showing a good linear relationship between the independent and dependent variables to points scored.

Sum of Squared Errors

```
SSE <- sum(PointsRS$residuals^2)
print(SSE)
```

```
## [1] 28394314
```

Root Mean Error

```
RMSE <- sqrt(SSE/PointsRS$df.residual)
RMSE
```

```
## [1] 185.5191
```

```
mean(NBA$PTS)
```

```
## [1] 8370.24
```

Fractional error = 2.2%

It may be interesting to check the correlations between the variables that we included in this first model, to get some hints as to collinearity, which could be relevant to know if we wanted to remove some variables.

Running model #2 removed TOV

```
PointsRS2 <- lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + DRB + STL + BLK, data = NBA)

summary(PointsRS2)
```

```
##
## Call:
## lm(formula = PTS ~ X2PA + X3PA + FTA + AST + ORB + DRB + STL +
##     BLK, data = NBA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -526.79 -121.09    6.37  120.74  565.94
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.077e+03  1.931e+02 -10.755   <2e-16 ***
## X2PA         1.044e+00  2.951e-02  35.366   <2e-16 ***
## X3PA         1.263e+00  3.703e-02  34.099   <2e-16 ***
## FTA          1.125e+00  3.308e-02  34.023   <2e-16 ***
## AST          8.861e-01  4.393e-02  20.173   <2e-16 ***
## ORB         -9.581e-01  7.758e-02 -12.350   <2e-16 ***
## DRB          3.892e-02  6.154e-02   0.632   0.5273
## STL         -2.068e-01  8.984e-02  -2.301   0.0216 *
## BLK         -5.863e-02  8.749e-02  -0.670   0.5029
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 185.4 on 826 degrees of freedom
## Multiple R-squared:  0.8991, Adjusted R-squared:  0.8982
## F-statistic: 920.4 on 8 and 826 DF,  p-value: < 2.2e-16
```

By removing TOV $R^2 = 0.8991$ which is higher than model1 at 0.8992. In model3 DRB is removed

```
PointsRS3 <- lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + STL + BLK, data = NBA)
summary(PointsRS3)
```

```
##
## Call:
## lm(formula = PTS ~ X2PA + X3PA + FTA + AST + ORB + STL + BLK,
##     data = NBA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -523.79 -121.64    6.07  120.81  573.64
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.015e+03  1.670e+02 -12.068  < 2e-16 ***
## X2PA         1.048e+00  2.852e-02  36.753  < 2e-16 ***
## X3PA         1.271e+00  3.475e-02  36.568  < 2e-16 ***
## FTA          1.128e+00  3.270e-02  34.506  < 2e-16 ***
## AST          8.909e-01  4.326e-02  20.597  < 2e-16 ***
## ORB         -9.702e-01  7.519e-02 -12.903  < 2e-16 ***
## STL         -2.276e-01  8.356e-02  -2.724  0.00659 **
## BLK         -3.882e-02  8.165e-02  -0.475  0.63462
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 185.4 on 827 degrees of freedom
## Multiple R-squared:  0.8991, Adjusted R-squared:  0.8982
## F-statistic:  1053 on 7 and 827 DF,  p-value: < 2.2e-16
```

The model3 Rˆ is the same at Model1 at 0.8991

In model4 BLK is removed

```
PointsRS4 <- lm(PTS ~ X2PA + X3PA + FTA + AST + ORB + STL, data = NBA)
summary(PointsRS4)
```

```
##
## Call:
## lm(formula = PTS ~ X2PA + X3PA + FTA + AST + ORB + STL, data = NBA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -523.33 -122.02    6.93  120.68  568.26
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.033e+03  1.629e+02 -12.475  < 2e-16 ***
## X2PA         1.050e+00  2.829e-02  37.117  < 2e-16 ***
## X3PA         1.273e+00  3.441e-02  37.001  < 2e-16 ***
## FTA          1.127e+00  3.260e-02  34.581  < 2e-16 ***
## AST          8.884e-01  4.292e-02  20.701  < 2e-16 ***
## ORB         -9.743e-01  7.465e-02 -13.051  < 2e-16 ***
## STL         -2.268e-01  8.350e-02  -2.717  0.00673 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 185.3 on 828 degrees of freedom
## Multiple R-squared:  0.8991, Adjusted R-squared:  0.8983
## F-statistic:  1229 on 6 and 828 DF,  p-value: < 2.2e-16
```

The model4 Rˆ is the same at Model1 and model3 at 0.8991 A closer look at SSE and RMSE of model4

```
SSE_4 <-sum(PointsRS4$residuals^2)
RSME_4 <- sqrt(SSE_4/nrow(NBA))

SSE_4
```

```
## [1] 28421465
```

```
RSME_4
```

```
## [1] 184.493
```

The values for Model4 (PointsRS4) are SSE = 28421464.9 and RMSE = 184.493 compared the the model1 RMSE = 185.5191 (essentially the same)

Time for predictions Read in NBA_test data

```r
NBA_test <- read.csv("C:/Users/narce/OneDrive/Documents/GitHub/NBA/NBA/NBA_test.csv")
```

Attempting to predict using model4, the number of points in the 2012-2013 season using predict() and the new NBA_test.csv

```r
PointsPredictions <- predict(PointsRS4, newdata = NBA_test)
summary(PointsPredictions)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    7507    7843    8005    8032    8169    8620
```

To determine the accuracy of model4 (0.8991) the SSE, SST, RMSE must be analyzed

```r
SSE <- sum((PointsPredictions - NBA_test$PTS)^2)
SSE
```

```
## [1] 1079739
```

```r
SST <- sum((mean(NBA$PTS) - NBA_test$PTS)^2)
SST
```

```
## [1] 5765192
```

R2 is calculated with 1 minus the sum of squared errors divided by the total sum of squares

```r
R2 <- 1 - SSE/SST
R2
```

```
## [1] 0.8127142
```

```r
RMSE <- sqrt(SSE/nrow(NBA_test))

RMSE
```

```
## [1] 196.3723
```

The values for Model4 (PointsRS4) are SSE = 28421464.9 and RMSE = compared the the model1 RMSE = 185.5191 (essentially the same) The predictions: model4 RMSE = 184.493 vs NBA_test = 196.37 model4 SSE = 28421464 vs NBA_test = 1079739 R^2 = 0.8127