What Distinguishes Hall of Fame Batters Statistically?

Calvin Makelky, A01872013

November 13, 2016

Problem: what stats differentiate a hall of fame batter with a batter who was eligible to be in the hall of fame, but did not make it?

Solution: combine all hall of fame batters with a random sample of batters who did not make it into the hall of fame and run a classification tree with being in the hall of fame as the response variable. This will give us what statistics and what point of the statistics distinguish the hall of famers.

Since 1898, almost every season has had 154 games or more according to http://research.sabr.org/journals/schedule-changes-since-1876. 10 seasons are needed in order to be considered for the Baseball Hall of Fame. The website that let you search for career stats on players by different criteria is baseball-reference.com. Unfortunately, baseball-reference will not let you search by seasons played. To get a sample of players who played 10 seasons, I saw that they all played 1000 games, so I searched based on this. Because a player can't be considered for the Hall of Fame until 5 years after retirement and can be considered for 10 years, I determined to only look at players whose last year was 2001 (2016-15). Since I only want the full career of players, I set the debut season to on or after 1898. Here is the query I used to find these non-Hall of Fame batters who played for ten or more seasons.

"Spanning Multiple Seasons or entire Careers, From 1898 to 2001, not a Hall Of Fame Member (as mlb players), Played 50% of games at C, 1B, 2B, 3B, SS, LF, CF, RF, DH or PHR, (requiring year_max<=2001, year_min>=1898 and At least 1000 games)"

Baseball-reference will not let you randomly select or see more than 10 players in the results without a subscription, so I used some random criteria to sort by that wouldn't bias the data. I sorted by most greatest first season, birth month, birth day, name, and sorted by lowest first season, birth month, birth day, and name. I exported all of this data into a csv file. To make sure I did not have any duplicated, I sorted the file in Excel by name. There was no duplicates, so there is a little over 80 batters who did not make the hall of fame but could have in my random sample. I would have liked more, but all other sortable criteria could have biased the sample (like height, games played, or weight).

For the hall of famers, I downloaded a .csv file of the career stats of non-pitchers. I excluded two players who predominatly played in the negoro leaugues since this only contains MLB data. There is over a 100 hall of fame batters, and my random sample of non-hall of famers has about a 100 batters.

```r
hof = read.csv("hof_batting.csv", header=TRUE)
hof$HOF = 1
#create new column, hof, coded as 1 that will be the response variable

nhof = read.csv("nonHOF.csv", header=TRUE)
nhof$HOF = 0
#create new column, hof, coded as 0 that will be the response variable
nhof$Yrs = nhof$To - nhof$From + 1
#calculate how many seasons a player has played in their career
nhof= subset(nhof, Yrs>9)
#removing players with less than 10 years

nhof = subset(nhof, select = -c(Rk, Name, X, Age, CS, Pos, Tm, IBB, HBP,
                                SF, From, To, From.1, To.1, GDP))
hof = subset(hof, select = -c(Rk, Name, Inducted, X, X.1, ASG, WAR.pos, X.2, X.3, X.4, X.5))
#getting rid of unmatching columns and useless columns
```

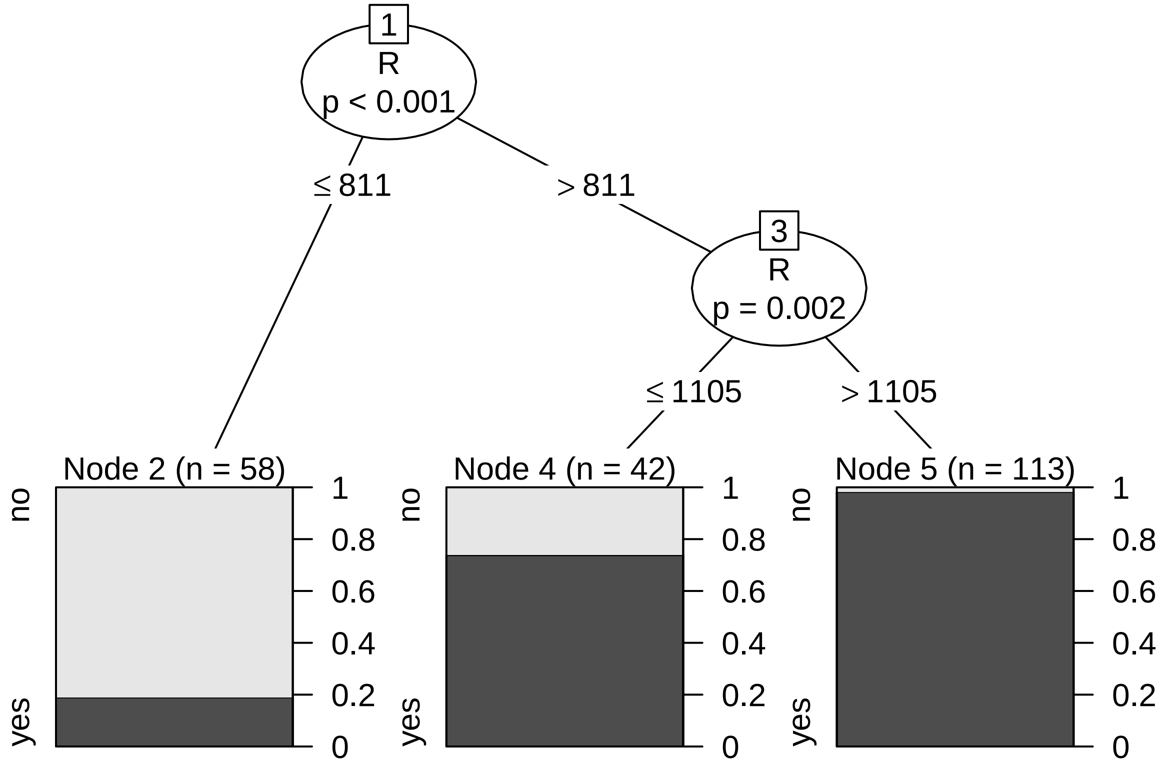
```r
hof = rbind(hof, nhof)
#combining hall of fame batter dataset with non-hall of fame batter dataset


hof$HOF <- factor(hof$HOF,
  levels = 0:1, labels = c("no", "yes"))
#better to have response variable factor type than numeric

library(party)
```

```
## Loading required package: grid

## Loading required package: mvtnorm

## Loading required package: modeltools

## Loading required package: stats4

## Loading required package: strucchange

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich
```

```r
hof.ct = ctree(HOF ~ Yrs + G  + AB + R + H + X2B + X3B +
HR + RBI + SB + BB + SO + SH + BA + OBP + SLG + OPS + PA, data = hof)
#constructing classification tree model
table(predict(hof.ct), hof$HOF)
```

```
##
##        no yes
##   no    47  11
##   yes   13 142
```

```r
plot(hof.ct)
```



The classification tree is very interesting. The only term the tree split off from is the number of career runs, which it does twice. Interpreting the plit we can see that every single batter than had more than 1105 career runs was a hall of famer. Around 80% of the batters who had less than 811 career runs are not in the hall of fame. And just below 80% of batters who had more than 811 career runs but less than or equal to 1105 career runs were in the hall of fame.

The table of prediction vs actual shows that 189 of the 213 batters were correctly classified as hall of famers or not. The Percent Correctly Classified then equals 100*(189/213) = 88.73%. This is a pretty good PCC, and this shows the model works well.

The Variable importance part of the summary shows that number of hits, at-bats, RBIs, runs, doubles, and games played are all very important. On the other hand, batting average, slugging, OPS, homeruns, strike-outs, and stolen bases had basically no importance in the model. The reason we don't see many of the important variables in the plot is because they are pruned from the tree in the ctree function that runs the best tree/model in terms of using as few predictors as possible without losing significant accuracy.