Using a Batter's Offensive Statistics to Predict Their Last Season Played Batting Average, Runs Batted-In (RBI's) and Homeruns

DSI Capstone Project

Mario Sanchez, Jr.

August 27, 2019

Question:

Can regression models be created to accurately predict a players batting average, runs batted-in (RBI's) and homeruns of their last season played?

Objectives:

- Obtain baseball data from trusted sources.
- Learn how to aggregate the multiple csv files into the form I need.
- Pick multiple regression models and compare their performance on unseen data.
- Which model performed best for each metric (HR's, RBI's, AVE)?
- Determine which features are most important to predicting each target.
- How can we use this information to improve our ability to predict a players performance using historical data?

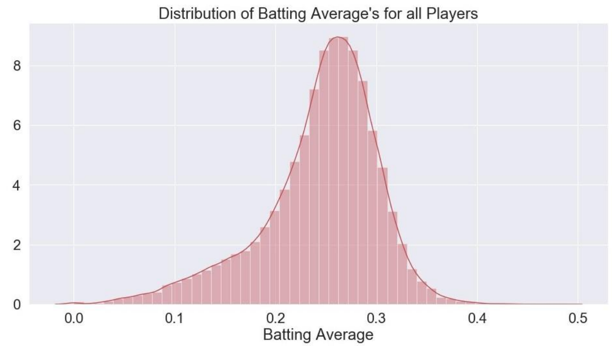
Description of the Data

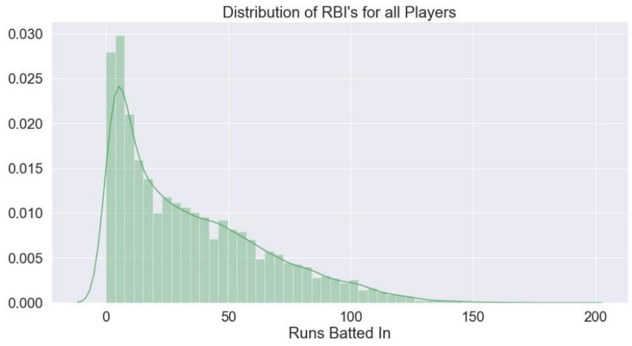
Sources:

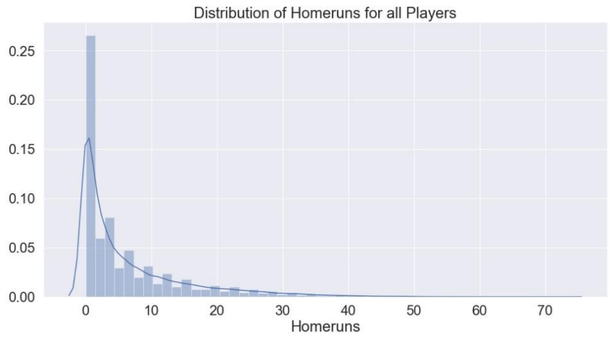
- Chadwick Baseball Bureau (http://www.chadwick-bureau.com)
- Lahman Baseball Database, version 2015-01-24, which is Copyright (C) 1996-2015 by Sean Lahman.
- The tables Parks.csv and HomeGames.csv are based on the game logs and park code table published by Retrosheet. This information is available free of charge from and is copyrighted by Retrosheet. Interested parties may contact Retrosheet at http://www.retrosheet.org.

Final DataFrames containing years: 1900-2018 and players with at least 5 years in the league:

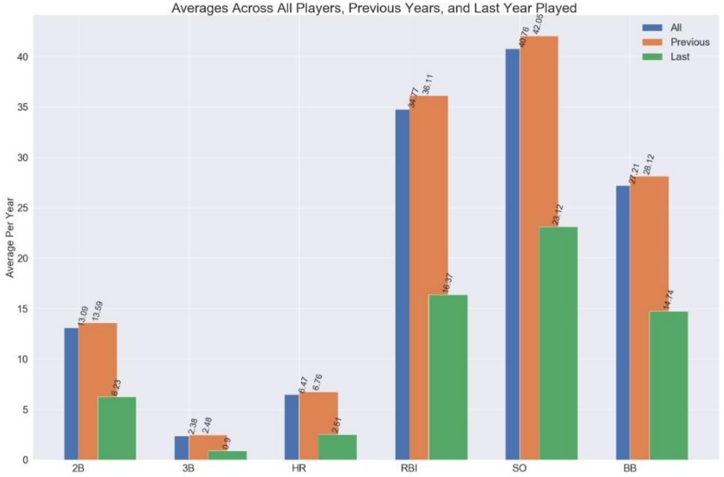
- <u>All Players Between 1900-2018</u>:
  batter_and_change_FINAL DataFrame has 41,745 rows and 84 columns
- <u>All Previous Years Played</u>:
  previous_years_FINAL DataFrame has 38,908 rows and 84 columns
- <u>The Last Year of a Players Career</u>:
  last_year_df_FINAL DataFrame has 2837 rows and 84 columns

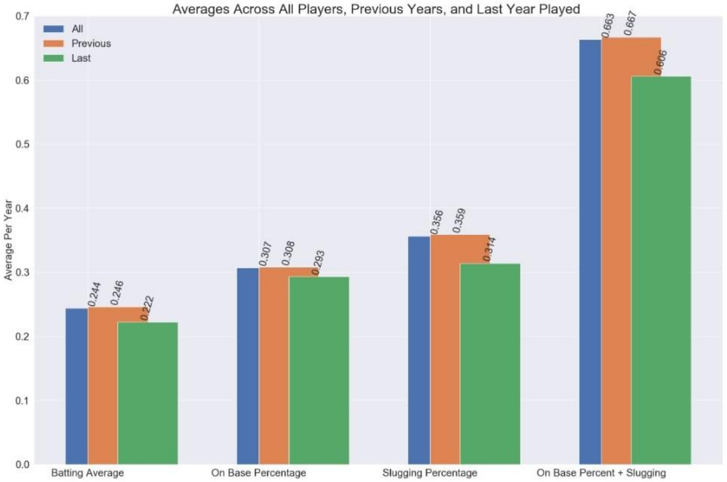






Description of the Data Continued





All Players 1900-2018

	G	AB	AVE	RBI	HR
count	41745.000000	41745.000000	41745.000000	41745.000000	41745.000000
mean	88.017056	281.305258	0.244268	34.771757	6.466427
std	45.520339	192.072007	0.057721	30.504605	8.674031
min	20.000000	20.000000	0.000000	0.000000	0.000000
25%	42.000000	94.000000	0.217252	9.000000	0.000000
50%	88.000000	250.000000	0.253886	27.000000	3.000000
75%	132.000000	459.000000	0.282353	53.000000	9.000000
max	165.000000	716.000000	0.485714	191.000000	73.000000

Previous Years

	G	AB	AVE	RBI	HR
count	38908.000000	38908.000000	38908.000000	38908.000000	38908.000000
mean	90.117097	290.678087	0.245869	36.113858	6.755269
std	45.643573	192.978232	0.057828	30.875618	8.855580
min	20.000000	20.000000	0.000000	0.000000	0.000000
25%	43.000000	98.000000	0.220000	9.000000	0.000000
50%	92.000000	268.000000	0.255735	29.000000	3.000000
75%	134.000000	471.000000	0.283665	55.000000	10.000000
max	165.000000	716.000000	0.485714	191.000000	73.000000

Last Year

	G	AB	AVE	RBI	HR
count	2837.000000	2837.000000	2837.000000	2837.000000	2837.000000
mean	59.216073	152.761720	0.222320	16.365527	2.505111
std	32.068875	119.929253	0.051430	15.963043	3.842233
min	20.000000	20.000000	0.000000	0.000000	0.000000
25%	32.000000	62.000000	0.194118	5.000000	0.000000
50%	51.000000	114.000000	0.226131	11.000000	1.000000
75%	81.000000	206.000000	0.256228	22.000000	3.000000
max	157.000000	629.000000	0.388889	127.000000	38.000000

Feature Engineering

- Wrote a function to assign an era label to each player dependent on when they played the game and then what percentage of their career they played in that era.
- Assigned a year label to each player and created dummy columns from these labels.
- Assigned a decade label to indicate which decades the player had played in.
- The above columns were created to account for the different era's that have occurred over the last 119 years.
- Created a binary column to indicate if a player batted and threw right handed.
- Created AVE, OBP, Slug_Percent, and OPS columns.
- Computed a players experience by subtracting the current year from the debut year.
- Split my final dataframe into previous years and final year so that I can test my models on unseen data and compare their results.

Columns Name	Description
playerID	unique identifier
yearID	year for that row
teamID	team played on
stint	stint
G	games played
AB	at-bats
R	runs
H	hits
2B	doubles
3B	triples
HR	homeruns
RBI	runs batted in
SB	stolen bases
CS	caught stealing
BB	base on balls
SO	strike out
IBB	intentional walk
HBP	hit by pitch
SH	sacrifice hit
SF	sacrifice fly
GIDP	grounded into double plays
nameFirst	first name
nameLast	last name
bats	left or right
throws	left or right
debut	first year played
finalGame	last year played
_percent	percent spent in that era
era	binary era
decade	binary decade
throws_R	1 if throws R
bats_R	1 if bats R
AVE	average
OBP	on base percentage
Slug_Percent	slugging percentage
OPS	on base + slugging
debutYear	first year played
currentYear	year of that row
YRSPRO	experience
_chg	change from previous year
KMeans_label	cluster label

Model Preparation:

- Used previous years to train and test on
- Made sure that columns such as G, H, AB were left out
- Scaled my train, test and unseen data for some of the regression models
- Created polynomial features to my X variable to provide more data to my models
- Grid search over several parameters for each model
- Created a pickle file of each fit and trained model for future evaluation.

Models

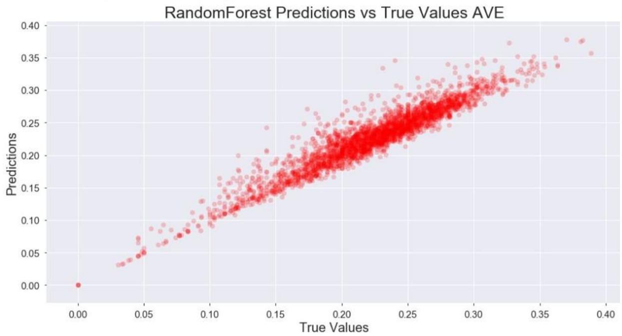
For each target, HR's, RBI's, and AVE, I fit each of the following models:

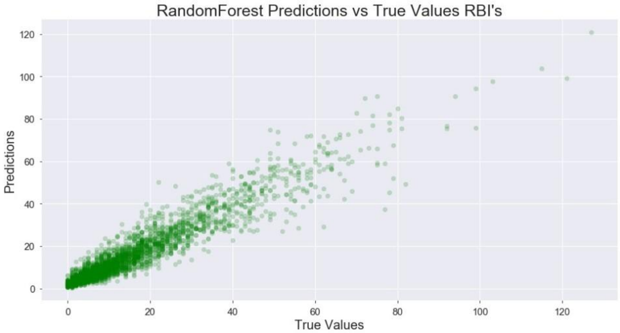
- Linear Regression
- Ridge
- Lasso
- ElasticNet
- RandomForest

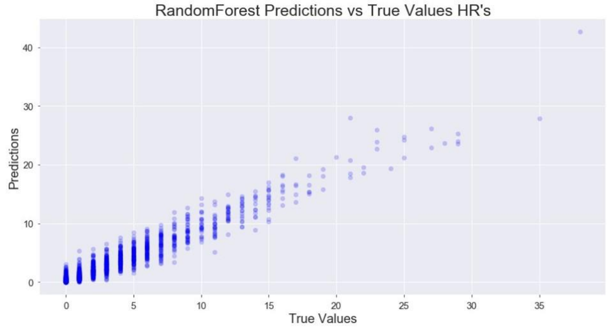
RandomForest Scores

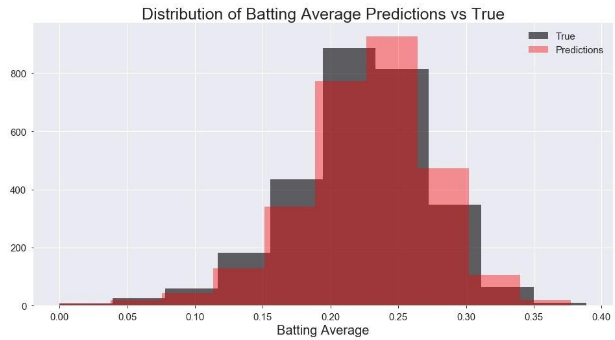
	AVE	RBI	HR
Train R2	0.9936	0.9887	0.9914
Test R2	0.9543	0.9362	0.9566
New Data R2	0.8939	0.8933	0.9323
New Data RMSE	0.0167	5.21	0.9998

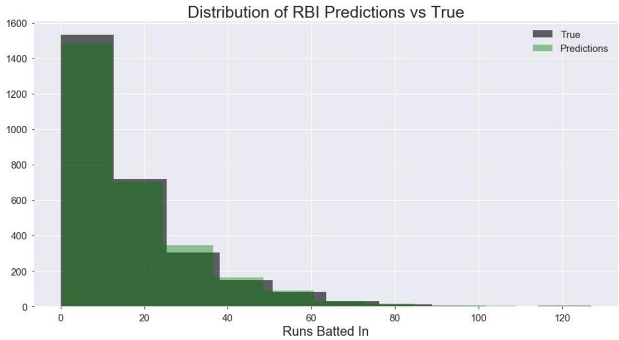
Best Model Results

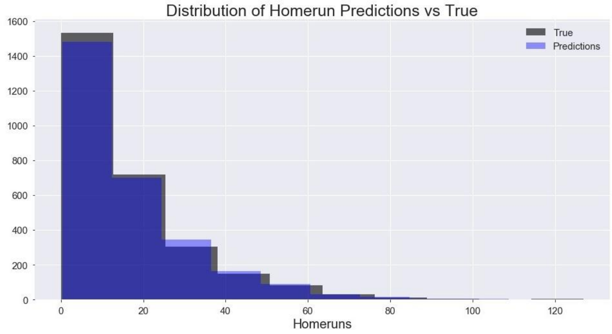












Feature Importance

RandomForest AVE

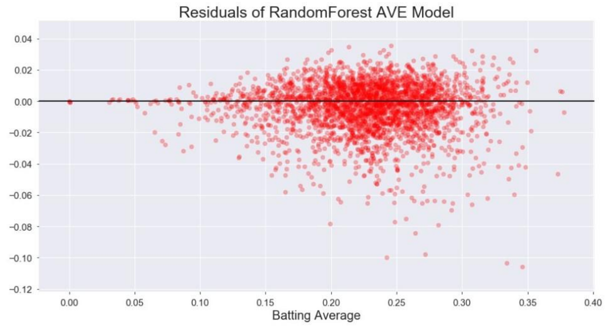
features	importance
OBP Slug_Percent	0.672240
OPS OBP	0.180931
OPS Slug_Percent	0.015534
BB Slug_Percent	0.012084
BB HR	0.010909
BB SO	0.007599
2B OBP	0.004799
OPS BB	0.003709
2B RBI	0.003132
BB GIDP	0.002926
2B SH	0.002197
2B 3B	0.002035
Slug_Percent^2	0.001979
OBP^2	0.001820
OBP	0.001776

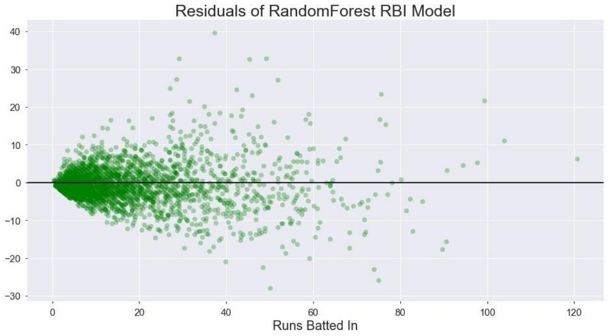
RandomForest RBI

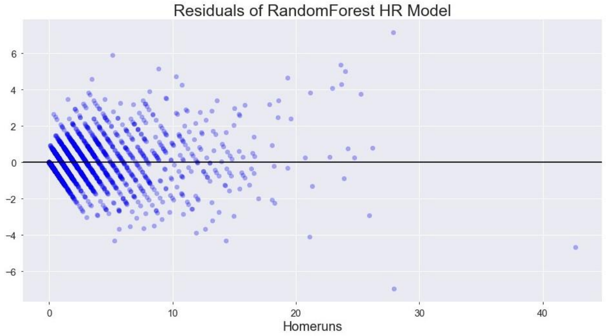
features	importance
2B Slug_Percent	0.600547
2B HR	0.165273
2B BB	0.080705
HR AVE	0.029625
2B 3B	0.012446
3B HR	0.003488
SF GIDP	0.003295
HR SF	0.003104
3B 1920-41_percent	0.002910
2B AVE	0.002892
HR GIDP	0.002810
3B SH	0.002161
2B SH	0.001597
HR SH	0.001431
BB GIDP	0.001331

RandomForest HR

features	importance
RBI SO	0.593223
RBI Slug_Percent	0.155370
Slug_Percent	0.065035
Slug_Percent^2	0.062440
SO Slug_Percent	0.017111
AVE^2	0.011973
AVE	0.011102
SO GIDP	0.009086
3B AVE	0.003270
2B 3B	0.002999
2B AVE	0.001984
RBI GIDP	0.001636
3B KMeans_label	0.001598
OBP AVE	0.001483
throws_R Slug_Percent	0.001343







Conclusions

Primary Findings:

- The RandomForest Regression model performed the best on predicting all metrics.
- Many of the models performed well on test data but poorly on unseen data.
- This indicated to me that using an ensemble model was a better approach at more accurately predicting my targets.
- I believe that I can achieve even better results if I log transform my target variable since there was a skewed distribution for two of the three targets.
- The information obtained from the models such as the most important features can then be leveraged to direct scouting reports and help teams better evaluate their players performance. These models allow for a players past history as well as others from around the league to determine what type of batter they are.
- This can then allow for a better forecast of a team's performance broken down by player.

Limitations and Assumptions:

- More feature engineering can improve the models.
- It is possible that there may have been some data leakage, further investigation is needed.
- Computing power becomes an issue when fitting certain models. This greatly influences how much tuning can go into each model.
- The results are encouraging and I believe they can by extended to predict many other offensive metrics.

Future Analysis:

- Explore other models such as ExtraTreesRegressor, AdaBoostRegressor and BaggingRegressor.
- Test models on other offensive metrics such as on-base percentage (OBP), or Slugging Percent and see if they can generalize well to these metrics.
- Test the models on select subsets of players such as by position or by era. This may reveal very interesting findings.
- There is truly a mountain of available data for baseball as well as other sports and am excited to apply what I have learned in this project to future projects.