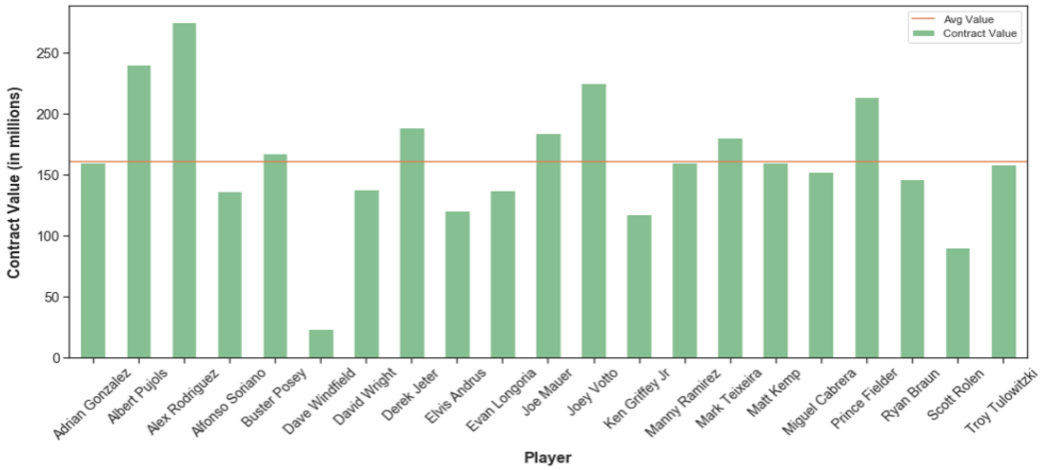
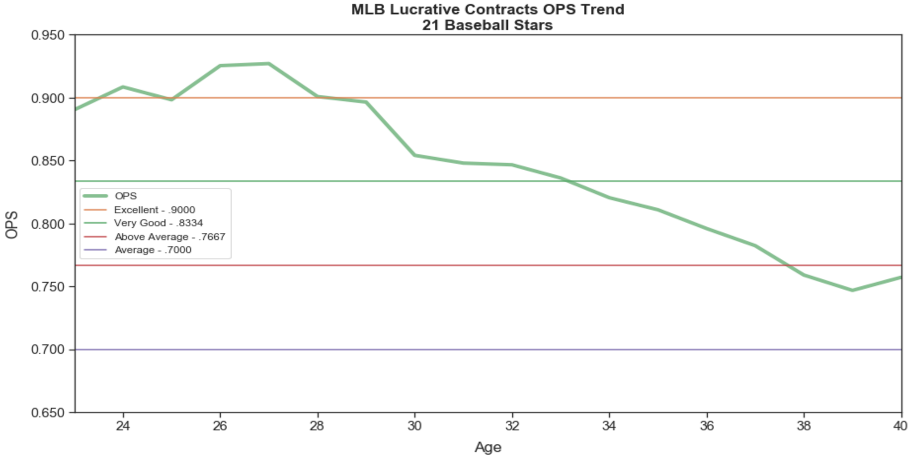
Executive Summary

Baseball contracts involve intense negotiations and millions of dollars are at stake. Recently, the St. Louis Cardinals signed Paul Goldschmidt, agreeing to a five-year, $130 million contract. Paul Goldschmidt is 32 years old. Age plays a big role in a baseball players performance. At some point as players get older, their performance on the field inevitably starts to decline. I have wondered whether the Paul Goldschmidt deal was good for the St. Louis Cardinals, and some have said, according to Forbes Magazine, there is reason to believe he could be entering the decline phase of his career. The Cardinals are betting that he will produce through the age of 37 years old. Will Paul Goldschmidt continue to perform through the age of 37? In more general terms, I would like to do analysis on MLB hitters and look at \`m\` years of past performance and predict the next \`n\` number of years of a major league baseball batter. Machine learning algorithms and predictive models will be used to facilitate the predictions. My customer in this analysis is baseball teams, analysts and fans.

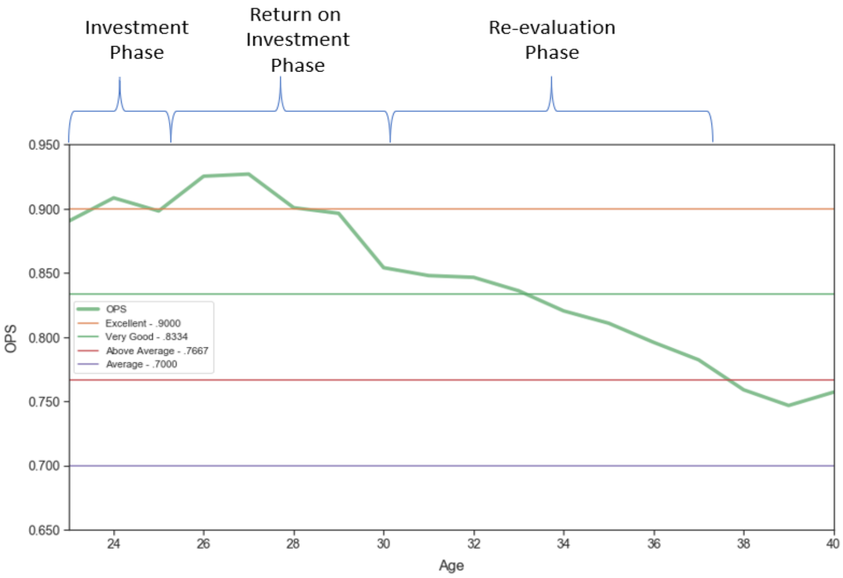
What do future contracts look like for the next superstar position player of the game? Why a new model? Because the existing model is great for superstar players, but not so much for the owners. Fox Sports provided an article of 21 top contracts in baseball (Top 21). The average contract value was about 170 million dollars with an average contract duration of about nine years. The figure below shows the 21 players and their contract value.



Using this information and baseball statistics from Sean Lahman\`s database, the 21 player\`s combined career OPS performance was charted. The below chart plots the age of the 21 players vs. combined OPS performance. The average age at signing of the lucrative contracts was approximately 28 years of age and ended at about the age of 36. Note the trend after the contracts were signed. The owners who signed these players will likely not get a reasonable return on investment.



As shown above the contracts were signed at the peak of the players performance curve at around 28 years of age. As the player aged their performance gradually decreased at a rate of about 15 OPS points per year. Owners need a better model for signing players as the below figure proposes.



In this model there are three stages: investment stage, return on investment stage and re-evaluation stage. The concept is to initially sign potential stars to a `fair` initial contract for 3 to 5 years. Advanced analytics and machine learning models will show that player batting performance can be predicted with reasonable accuracy given 2 to 4 years of performance data which will be shown as part of this project. Nothing is guaranteed, but there should be a specified certainty or confidence level. Next, sign the player to a 5 to 6 years lucrative contract from age 26 to 31, and then re-evaluate the player after their lucrative contract is over. In this manner, owners get a return on investment on the millions invested in this player. That`s a tall order given player and agents may not agree with the approach. But it should be the ultimate goal. The goal of this project is to show that predicting a batter`s performance with some amount of certainty is possible given 2 to 4 years of major league performance statistics. Performance prediction in sports is challenging. There are so many variables in a player`s performance such as early aging, injuries, family problems, drug dependency problems, etc. Think about it, a 95 mile per hour fastball from 60 feet 6 inches (pitcher`s mound to home plate distance) arrives across the plate in .434 seconds. In that time, the player needs to decide whether to swing or not and make solid contact with a baseball that is about 3 inches in diameter with a bat that is about 3 inches in diameter. Crazy!

The remainder of this paper goes into the details on how the data was acquired, transformed and validated making it usable data, the process of exploratory data analysis (EDA) and finally statistical analysis and machine learning modelling and results. The initial model is built predicting the next year of each player in order to get a baseline. For example, given a player`s first year, their second year OPS is predicted. Given a player`s 1st and 2nd year, their third year is predicted and so on.

The steroid era was a controversial era in baseball. Agreement on the era`s timeframe isn`t clear cut. Some say it started in the late 1980`s and others say it started in mid-1990`s. In 2004, major league baseball introduced mandatory PED testing which was the start of the end of that era. A lot of analysis has been done on PED usage during the `steroid era` in baseball. Did the players of this era really have an advantage over others and is it justified to keep them out of baseball`s Hall of Fame? This project uses the 1993 to 2003 timeframe for the steroid era.

Introduction

Major League Baseball has been America`s sports pastime for over 100 years and was first founded in 1871 for the National League and 1901 for the American League. Today, there are 15 teams in each league. In the year 2000, the two leagues merged into what is now known as Major League Baseball (MLB). Since the very beginning, statistics in baseball has played a major role in the game. In today`s baseball, advanced metrics are being used by every major league team in order to gain advantage over their competition. MLB organizations employ data science teams to collect this information for executives, general managers and coaches. But, statistics in baseball have always been polarizing. Some managers have lost their jobs recently because they could not adopt and did not believe in advanced metrics.

Quote from Bobby Bragan (baseball manager – 1940`s) – `Say you were standing with one foot in the oven and one foot in the ice bucket. According to the percentage people, you should be perfectly comfortable.`

Quote from Leonard Koppett (A Thinking Man`s Guide to Baseball – 1967) – `Statistics are the lifeblood of baseball. In no other sport are so many available and studied so assiduously by participants and fans. Much of the game`s appeal, as a conversation piece, lies in the opportunity the fans get to backup up opinions and arguments with convincing figures, and it is entirely possible that more American boys have mastered long division by dealing with batting averages than in any other way.`

As ESPN Analyst Harold Reynolds said, `All of the sudden, it`s not just BA and Runs Scored, it`s OBA. And what is O-P-S?` Certainly, the `old standard` hitting metrics like batting average and runs scored have given way to more advanced metrics such as OPS (on-base plus slugging) which is a more meaningful metric on how well a player is performing at the plate. There are many other advance metrics today in baseball as well. I will be using OPS for this project. OPS is calculated by adding a player`s on-base percentage with their slugging percentage.

The details of the equations are as follows:

OPS = OBP + SLG

OBP = (H + BB + HBP) / (AB + BB + SF + HBP)

TB = (nSingles * 1) + (nDoubles * 2) + (nTriples * 3) + (nHomeRuns * 4)

SLG = TB / AB

Where:

H – total number of hits of a player

BB – total number of walks (base on balls) of a player

HBP – total number of times the player was hit by a pitch

AB – total number of plate appearances (times at bat) by the player

SF – total number of sacrifice flies of a player

TB – is the total bases and is a weighted sum (1 for single, 2 for double, 3 for triple, 4 for HR).

So, TB is (nSingles * 1) + (nDoubles * 2) + (nTriples * 3) + (nHomeRuns * 4)) where nSingles is the number of singles, nDoubles is the number of doubles, nTriples is the number of triples and nHomeRuns is the number of home runs.

OPS – on-base plus slugging

OBP – on-base percentage

SLG – slugging percentage

NOTE: all statistics are taken over a period of time (typically a year)

(source : Wikipedia)

I would like to use OPS for both parts (performance/age prediction and PED usage) of analysis. In order to do this analysis, I have done some internet research and found raw baseball data collected from 1871 to 2018 of major league baseball games. All the above atomic data elements such as hits, at bats, etc are available and therefore OPS, OBP and SLG can be computed. Thanks to Sean Lahman and others, they have created a database with yearly baseball statistics from 1871 to 2018. The database has copyright 1996-2018 by Sean Lahman. I have read the license agreement which is licensed under Creative Commons Attribution and will not restrict me from using this data. The raw data needed will be from the 1954 to 2018. Up until 1954, sacrifice fly statistics were not consistently collected and they are needed for the OPS calculation. So, I am using data from 1954 onward as my population data.

Tony La Russa (ex St. Louis Cardinal Manager) has been quoted as saying (paraphrased) `you may not agree with me, but you don`t have all of the information that I have`. Now we do.

Batting Performance Predictive Model – One Year Look Ahead

Data for this project comes from Sean Lahman`s baseball data sets with batting performance collected from 1871 to 2018. Models were built using a set of features without the use of regression towards the mean (RTM) and with the use of RTM. Various regression machine learning algorithms were implemented. The algorithms were Linear Regression, Ridge Regression, Lasso Regression, Non-Linear Regression, Random Forests, Support Vector Machines and XGBoost algorithms. There were two models built, one for predicting one year look ahead OPS and one for predicting one year look ahead career OPS. Features for the model were as follows:

Features for our machine learning model were created along with the y value (OPS and career OPS). Lag1 values were used as features. For 2017, the 2016 OPS, SLG, OBP, etc actuals and rtm values were used as feagures. After a substantial effort, the following are the features that were used for all model runs.

`ndecade` – current decade for the year in which the player participated (zero mean normalized).
`nage` – age of the player for each year played normalized (zero mean normalized).
`nheight` – height of the player (zero mean normalized). It turns out weight doesn`t affect the model.
`POS_1B` – bit map of 1 if player is a first baseman, 0 otherwise.
`POS_2B` – bit map of 1 if player is a second baseman, 0 otherwise.
`POS_3B` – bit map of 1 if player is a third baseman, 0 otherwise.
`POS_SS` – bit map of 1 if player is a short stop baseman, 0 otherwise.
`POS_OF` – bit map of 1 if player is an outfielder, 0 otherwise.
`lag1_nSLG` – previous year slugging with regression towards the mean (rtm) applied (zero mean normalized).
`lag1_ncSLG` – previous year slugging without rtm applied (zero mean normalized).
`lag1_nOBP` – previous year on base percentage without rtm applied (zero mean normalized).
`lag1_ncOBP` – previous year career on base percentage without rtm applied (zero mean normalized).
`lag1_nOPS` – previous year on base plus slugging without rtm applied (zero mean normalized).
`lag1_ncOPS` – previous year career on base plus slugging without rtm applied (zero mean normalized).
`lag1_nHR` – previous year Home Runs without rtm applied (zero mean normalized).
`lag1_ncHR` – previous year career Home Runs without rtm applied (zero mean normalized).

The models were run with a training set and a testing set with 80% training and 20% testing. A custom training / test split algorithm was used in favor of the standard train_test_split() function. Given a player`s career, there should be no player that is split across the training / testing sets. That is, given a player, all yearly career statistics for this player should either be in the training set or the test set but not both. In this way, the training set does not see any years of a player who is being predicted in the testing set.

The training set input to the model consisted of all data from 1954 to 2018 excluding players who had less than 300 at bats (AB) in a given year. The purpose of this was to exclude pitchers who were included in the data as well as utility players and other players who did not have much playing time. The segment of the population of my interest is players who are starters and play full time. This is a bias but an intended one.

The testing set also excluded players with less than 300 at bats. In addition, analysis was performed and any player with an OPS of .300 or less or an OPS of 1.2 or greater was excluded as they full outside two standard deviations of OPS values and viewed as outliers. Players with age of 19 or under and players 38 and over were also excluded as they fell well outside two standard deviations for player ages.

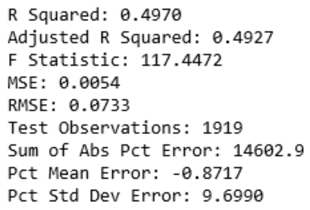
See the results of all the runs by clicking on link below.

GitHub Machine Learning Jupyter Notebook - Yearly OPS

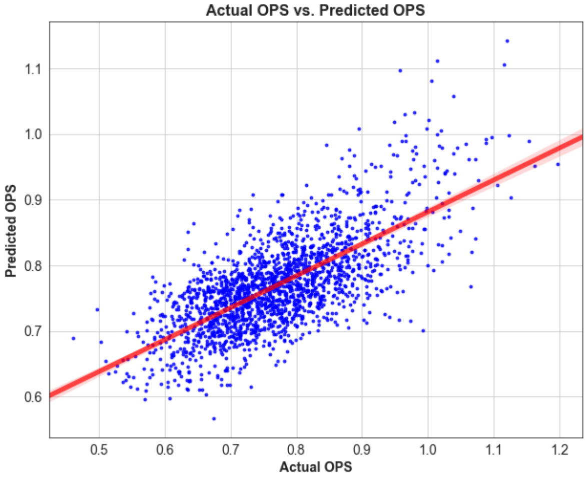
GitHub Machine Learning Jupyter Notebooks - Career OPS

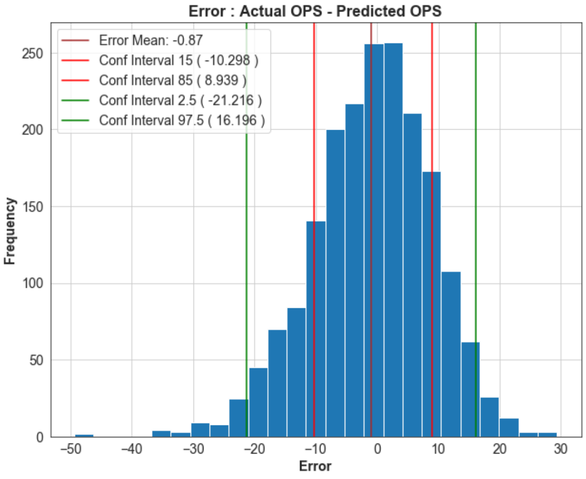
I have included the algorithm results which performed the best. XGBoost algorithm which stands for Extreme Gradient Boosting is a boosted tree algorithm and had the best scores.

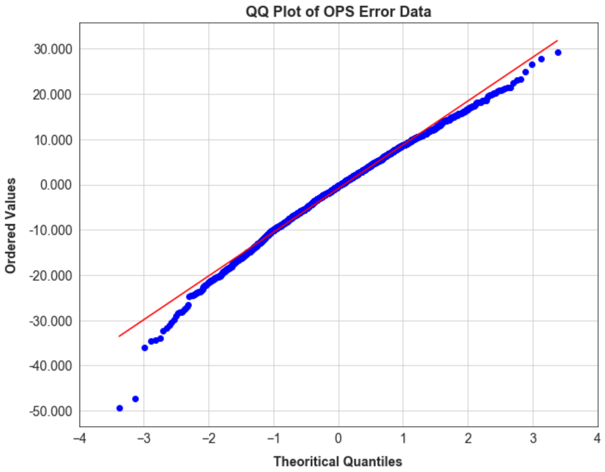
Here are the results from the run for predicted yearly OPS:

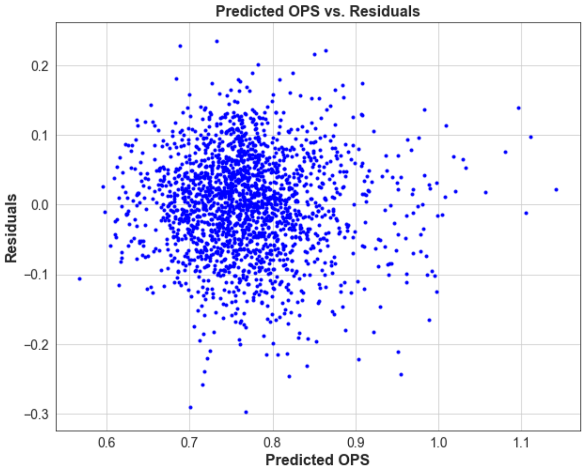


```
R Squared: 0.4970
Adjusted R Squared: 0.4927
F Statistic: 117.4472
MSE: 0.0054
RMSE: 0.0733
Test Observations: 1919
Sum of Abs Pct Error: 14602.9
Pct Mean Error: -0.8717
Pct Std Dev Error: 9.6990
```

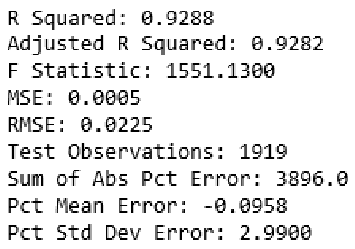

Actual OPS vs. Predicted OPS


Error : Actual OPS - Predicted OPS

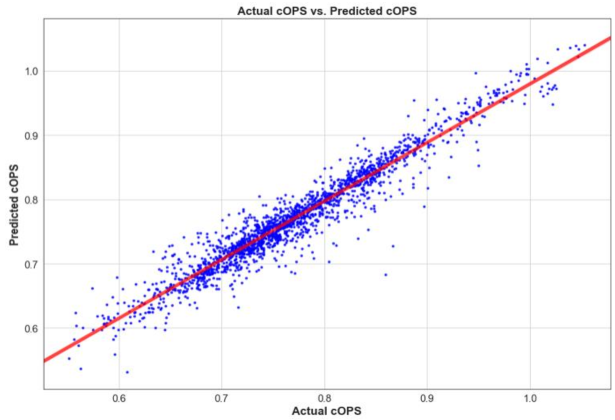

QQ Plot of OPS Error Data

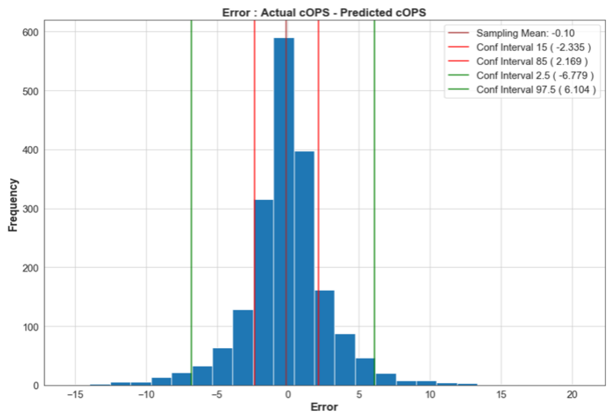

Predicted OPS vs. Residuals

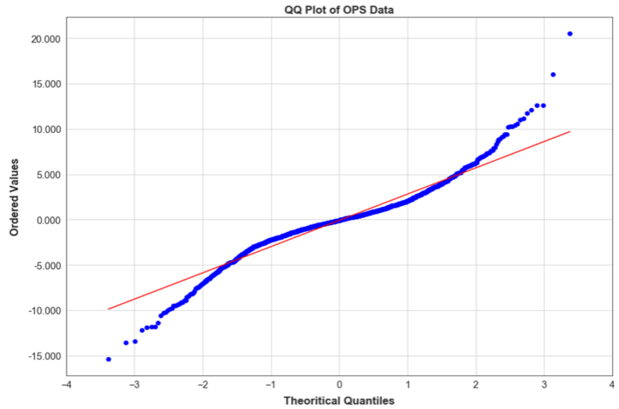
In addition to prediction of yearly OPS, career OPS was predicted as well. Career OPS is just a cumulative OPS of a player`s career. For year 1 in the league, it would be just year 1 OPS. For year 2, it would be year 1 and year2 statistics combined and so on. The results of career OPS were much better from a modeling perspective. The following represents the machine learning results of career OPS using XGBoost.

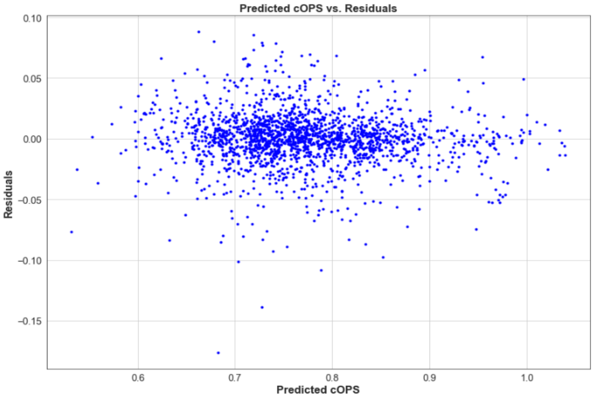


```
R Squared: 0.9288
Adjusted R Squared: 0.9282
F Statistic: 1551.1300
MSE: 0.0005
RMSE: 0.0225
Test Observations: 1919
Sum of Abs Pct Error: 3896.0
Pct Mean Error: -0.0958
Pct Std Dev Error: 2.9900
```

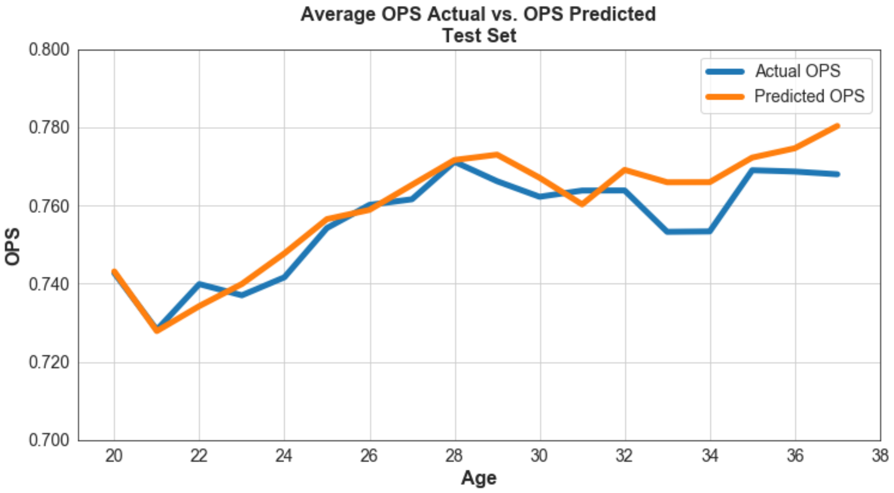




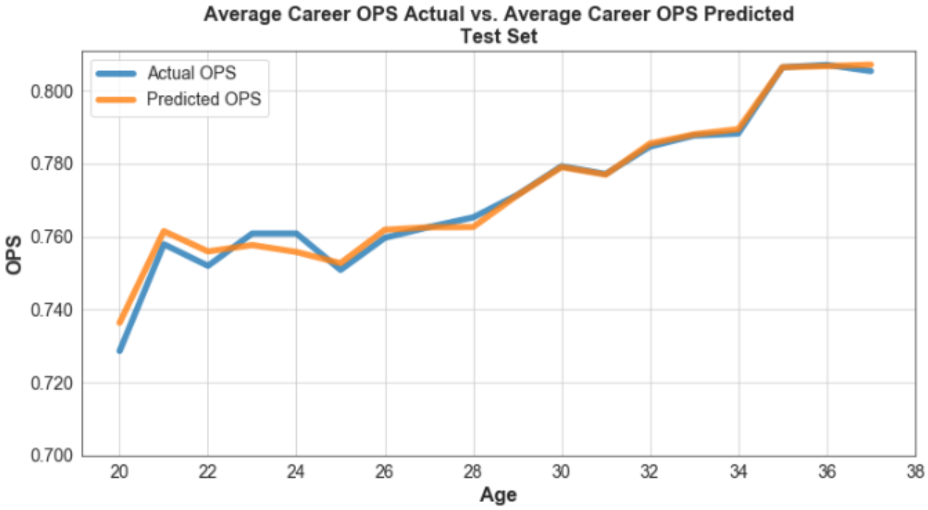




The average actual OPS vs. average predicted OPS was plotted by age as well as career actual OPS vs. career predicted OPS. The next two plots show the results.



Below are the results of the average actual career OPS vs. average predicted career OPS.



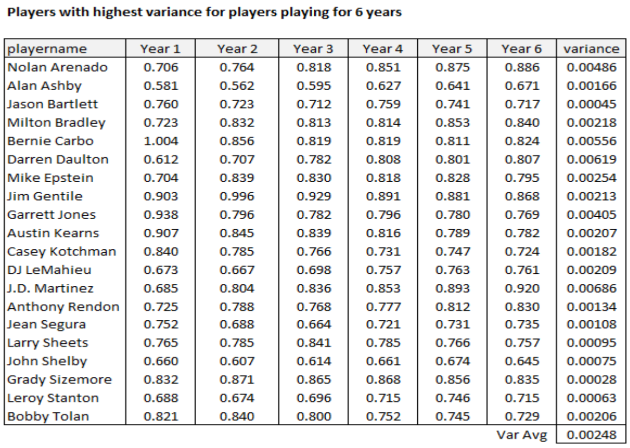
Note that at about the age of 25 for career OPS, the actual vs predicted are very close. Individual performance predictions were made as well. See the Jupyter Notebook link for individual player performance comparisons.

OPS Jupyter Notebook

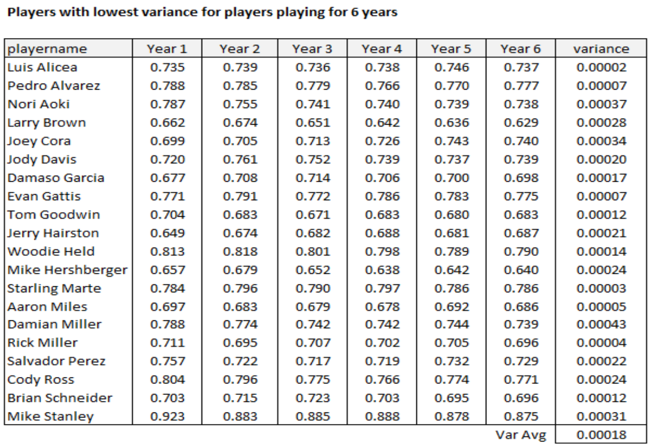
Career OPS Jupyter Notebook

Batting Performance : Five Year Forecast

The last part of this project is to try to predict 5 year projections of players. For example, if a player has played for n years in major league baseball, can the next five years be predicted with reasonable accuracy. The first question to be answered is `n` years. To help answer this, the following experiment was performed. All players who had a career for 6 years were selected from the population of players from 1960 to 2018. The players with the highest OPS variance over the 6 years (top 20) were selected, and the players with the lowest OPS variance over the 6 years (top 20) were selected. The following are the two tables of the players

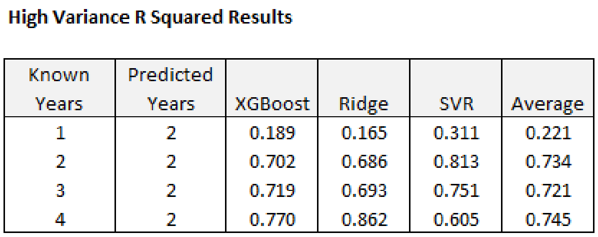

Players with highest variance for players playing for 6 years

playername	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6	variance
Nolan Arenado	0.706	0.764	0.818	0.851	0.875	0.886	0.00486
Alan Ashby	0.581	0.562	0.595	0.627	0.641	0.671	0.00166
Jason Bartlett	0.760	0.723	0.712	0.759	0.741	0.717	0.00045
Milton Bradley	0.723	0.832	0.813	0.814	0.853	0.840	0.00218
Bernie Carbo	1.004	0.856	0.819	0.819	0.811	0.824	0.00556
Darren Daulton	0.612	0.707	0.782	0.808	0.801	0.807	0.00619
Mike Epstein	0.704	0.839	0.830	0.818	0.828	0.795	0.00254
Jim Gentile	0.903	0.996	0.929	0.891	0.881	0.868	0.00213
Garrett Jones	0.938	0.796	0.782	0.796	0.780	0.769	0.00405
Austin Kearns	0.907	0.845	0.839	0.816	0.789	0.782	0.00207
Casey Kotchman	0.840	0.785	0.766	0.731	0.747	0.724	0.00182
DJ LeMahieu	0.673	0.667	0.698	0.757	0.763	0.761	0.00209
J.D. Martinez	0.685	0.804	0.836	0.853	0.893	0.920	0.00686
Anthony Rendon	0.725	0.788	0.768	0.777	0.812	0.830	0.00134
Jean Segura	0.752	0.688	0.664	0.721	0.731	0.735	0.00108
Larry Sheets	0.765	0.785	0.841	0.785	0.766	0.757	0.00095
John Shelby	0.660	0.607	0.614	0.661	0.674	0.645	0.00075
Grady Sizemore	0.832	0.871	0.865	0.868	0.856	0.835	0.00028
Leroy Stanton	0.688	0.674	0.696	0.715	0.746	0.715	0.00063
Bobby Tolan	0.821	0.840	0.800	0.752	0.745	0.729	0.00206
						Var Avg	0.00248

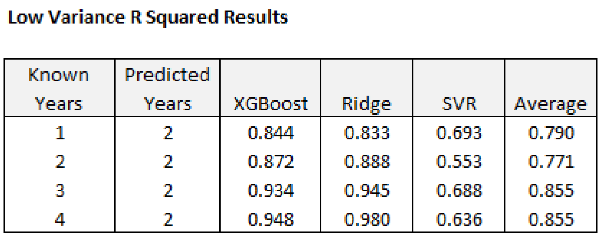

Players with lowest variance for players playing for 6 years

playername	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6	variance
Luis Alicea	0.735	0.739	0.736	0.738	0.746	0.737	0.00002
Pedro Alvarez	0.788	0.785	0.779	0.766	0.770	0.777	0.00007
Nori Aoki	0.787	0.755	0.741	0.740	0.739	0.738	0.00037
Larry Brown	0.662	0.674	0.651	0.642	0.636	0.629	0.00028
Joey Cora	0.699	0.705	0.713	0.726	0.743	0.740	0.00034
Jody Davis	0.720	0.761	0.752	0.739	0.737	0.739	0.00020
Damaso Garcia	0.677	0.708	0.714	0.706	0.700	0.698	0.00017
Evan Gattis	0.771	0.791	0.772	0.786	0.783	0.775	0.00007
Tom Goodwin	0.704	0.683	0.671	0.683	0.680	0.683	0.00012
Jerry Hairston	0.649	0.674	0.682	0.688	0.681	0.687	0.00021
Woodie Held	0.813	0.818	0.801	0.798	0.789	0.790	0.00014
Mike Hershberger	0.657	0.679	0.652	0.638	0.642	0.640	0.00024
Starling Marte	0.784	0.796	0.790	0.797	0.786	0.786	0.00003
Aaron Miles	0.697	0.683	0.679	0.678	0.692	0.686	0.00005
Damian Miller	0.788	0.774	0.742	0.742	0.744	0.739	0.00043
Rick Miller	0.711	0.695	0.707	0.702	0.705	0.696	0.00004
Salvador Perez	0.757	0.722	0.717	0.719	0.732	0.729	0.00022
Cody Ross	0.804	0.796	0.775	0.766	0.774	0.771	0.00024
Brian Schneider	0.703	0.715	0.723	0.703	0.695	0.696	0.00012
Mike Stanley	0.923	0.883	0.885	0.888	0.878	0.875	0.00031
						Var Avg	0.00018

The two sets of data were run through the machine learning model starting with a known year 1 and predicting two years out. Then, a known 2 years predicting 2 years out. Next, a known 3 years predicting 2 years out. And finally, a known 4 years and predicting 2 years out. The idea is to better understand how variability of OPS effects the predictive model and is there a number of years of known performance that needs to be known to predict future years.

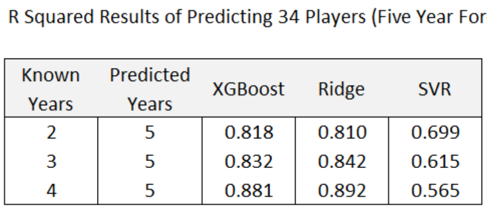

High Variance R Squared Results

Known Years	Predicted Years	XGBoost	Ridge	SVR	Average
1	2	0.189	0.165	0.311	0.221
2	2	0.702	0.686	0.813	0.734
3	2	0.719	0.693	0.751	0.721
4	2	0.770	0.862	0.605	0.745


Low Variance R Squared Results

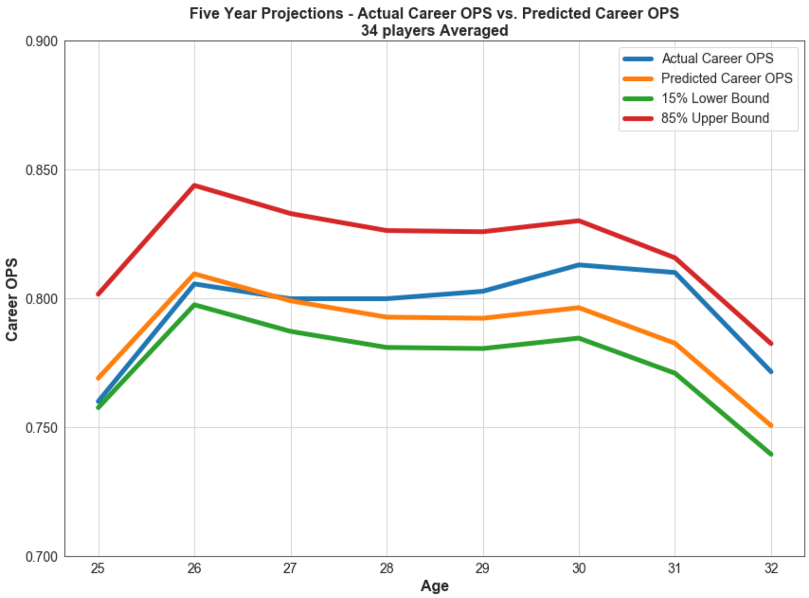
Known Years	Predicted Years	XGBoost	Ridge	SVR	Average
1	2	0.844	0.833	0.693	0.790
2	2	0.872	0.888	0.553	0.771
3	2	0.934	0.945	0.688	0.855
4	2	0.948	0.980	0.636	0.855

Note the players who had a very low OPS variance over their 6-year career had very high R Squared values and happened quickly. Players who had high OPS variability over their 6-year career were inconsistent initially but got better over known years. It looks like as little as two years of known performance helps predict future performance. As the saying goes `the best predictor of future behavior is past behavior`. Below are the R squared results of predicting career OPS five years out. It shows that after as little as two years of play, predicted career OPS metrics are a good indicator of future batting performance.

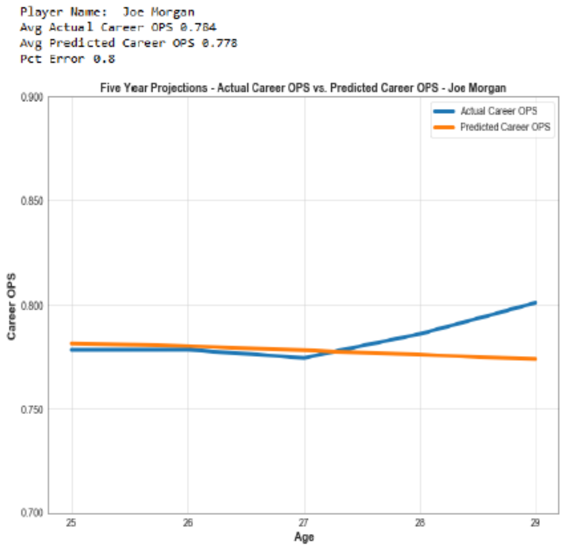

R Squared Results of Predicting 34 Players (Five Year Forecast)

Known Years	Predicted Years	XGBoost	Ridge	SVR
2	5	0.818	0.810	0.699
3	5	0.832	0.842	0.615
4	5	0.881	0.892	0.565

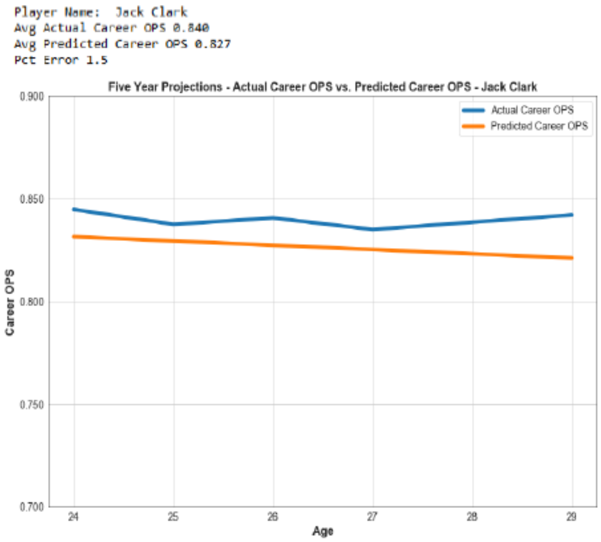
The above results consist of 34 players who played at least 10 years. Their first year in the majors was between 22 and 24 years of age for each of the 32 players. The below chart below shows the predicted five years of career OPS. Note the lower bound is the important one, and it is very close to the actuals line plot. It shows that the actual career OPS metric is above this curve with 70% certainty. It is not a perfect indicator but gives a good idea of how well this player will do over the next five years. Typically, a 95% control interval would be used, but because batting performance is so difficult to predict, it does not mean much because the range would be too great for my predictions.



Here are the results of using career OPS to predict Joe Morgan and Jack Clark (one of my favorites).



```
Player Name:  Joe Morgan
Avg Actual Career OPS 0.784
Avg Predicted Career OPS 0.778
Pct Error 0.8
```



```
Player Name:  Jack Clark
Avg Actual Career OPS 0.840
Avg Predicted Career OPS 0.827
Pct Error 1.5
```

Not that the predicted career OPS provides a good lower bound for the player. This was true for almost all of the 34 players.

Was the signing of Paul Goldschmidt a good deal for the Cardinals? Predictions using career OPS are really only useful for the first few years of a career which is very powerful. In Paul Goldschmidt`s case, he has played for nine years including 2019. His career OPS numbers will tend to dominate any predictions and will not be useful. And predicting OPS for five years is not practical (too much variation). Looking at Paul`s last 5 years including 2019, his OPS numbers have dropped an average of 37 points per year. But last year he signed with the Cardinals with a 2019 OPS of .821 which probably does not match up to his abilities. Looking at the 21 lucrative contracts and tracking their OPS decline, it is about .015 OPS points per year. Using this information, my prediction for the next four years is as follows:

Year	Age	OPS
2020	32	0.857
2021	33	0.842
2022	34	0.827
2023	35	0.812
	Avg OPS	0.835

Adding in the .821 in 2019, his average predicted OPS is .832 which puts him on the borderline of very good and above average OPS category.

Regression Towards the Mean

The discovery of Stein`s Paradox in 1955 by Charles Stein of Stanford University undermined a century and a half of work on estimation theory (Morris, 1977). Stein`s paradox concerns the use of averages to estimate unobservable quantities. From a baseball perspective, if you have 10 players and want to predict future batting averages of the 10 players, it is better to look at the 10 players as a whole, then to try to predict each person`s batting average individually. And the future averages can be predicted no matter what the batting abilities of the players actually are. The process in Stein`s method is to first calculate the average of the averages (grand average). Then shrink the individual averages towards the grand average. In other words, regression towards the mean (RTM). So, if one of the 10 players has a batting average above the overall league average then this player`s average must be reduced. Alternatively, if one of the player`s average is below the overall league average, then it must be increased. This shrunken value `z` is the James-Stein estimator. See the following link for additional information (Stein`s Paradox). This project will apply regression towards the mean to baseball batting performance statistic OPS (On Base Plus Slugging). In 1970, Bradley Efron and Carl Morris applied Stein`s methods to 18 major league baseball players. The figure below shows that the players true averages are clustered more closely around the grand average as predicted than was shown earlier in the season. The James Stein Estimator was used to predict the future state baseball batting averages of the 18 players. Even through the players initial averages were widely spread, in the end they clustered around the grand average as Stein`s Paradox suggests.

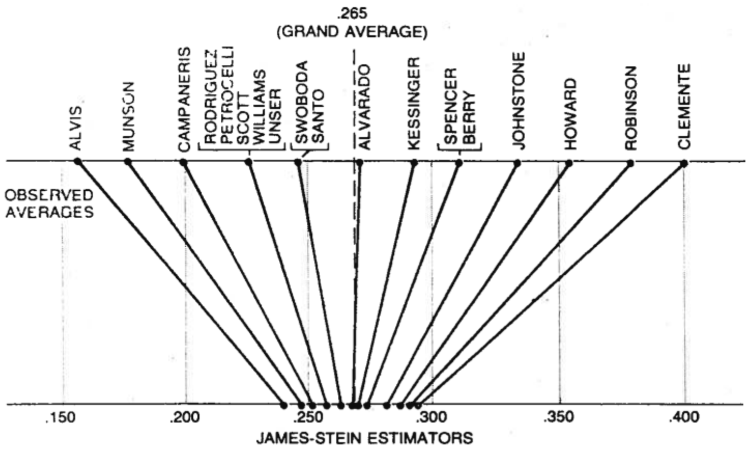


Figure was extracted from article by Efron and Morris of Stanford University

There were three techniques that were considered for implementing the regression towards the mean which are 1) James Stein Estimator 2) Pearson`s Correlation Coefficient 3) Bernoulli Trials estimator. The Bernoulli Trials approach appealed to me because of its simplicity and flexibility. It seemed to fit well within the feature frameworks of my machine learning models and algorithms. Information on each of the estimators can be found as follows: James Stein Estimator (Stein`s Paradox), Binomial Estimator 3-D Baseball and Pearson Correlation Coefficient Estimator (Pomona College).

Regression towards the mean was applied to On Base Percentage (OBP) and to Slugging (SLG) individually. Then the two were added together to get to the regression to the mean OPS. The approach was to calculate cumulative mean values OBP and SLG percentages. For example, the cumulative mean value of SLG was .419 (1954 to 1960) with a variance of .004643. **NOTE: only data from 1954 onward was collected due to lack of consistency of the treatment of Sacrifice Flies prior to 1954 even though data was available from 1871.** The cumulative mean value of OBP was .340 during this period with a variance of .001074. So, the cumulative mean value or grand mean of OPS was .759 in 1960. In 1970, cumulative SLG mean value was .409 (1954 to 1970), and the cumulative OBP was .334 resulting in a cumulative grand mean OPS of .743. These calculations were performed from 1960 through 2018 with data from 1954 to 2018.

To regress SLG to the mean, we need to calculate the number of at bats (AB) for which the binomial variance is the same as the variance of the true talent in the population sample.

Observed Variance = True Talent Variance + Variance Due to Binomial Distribution

or

True Talent Variance = Observed Variance - Variance Due to Binomial Distribution

The variance of the Binomial Distribution is so small due to the number of at bats (57,188) from 1954 to 1960. The variance due to binomial distribution with 57,888 at bats is .00000426 for SLG. True Talent Variance of SLG across all players from 1954 to 1960 would be .004643 - .00000426 = .004639. As the number of `at bats` increases, the variance due to binomial distributions gets very small and becomes negligible. For example, in 2018, the cumulative SLG variance is .005457 and the True Talent is .005457 because it is based upon a little over six million at bats from 1954 to 2018.

Now that we have the True Talent variance, we can find the number of at bats (n) which represents the True Talent Variance and plug it into the following equation noting that $\hat{p} \sim [p, p(1-p)/n]$

TrueTalentVariance = (p * (1-p)) / n or n = (p * (1-p)) / TrueTalentVariance

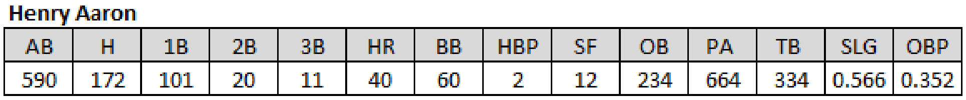
n = (.419 * (1-.419))/.004639 = **52.5** at bats

Next, the rtmTB needs to be calculated. To get this value multiply the SLG by the number of rtm at bats.

.419 * 52.5 = **22**

For the rtmOBP, rtmPA = n = (.340 * (1-.340)) / .001074 = **209.8255** plate appearances; rtmOB = 209.8255 * .340396 = **71.4** on base events.

Next, these values need to be applied to all players in 1960. Let`s take an example from 1960 for Henry Aaron. The following table represents Henry Aaron`s statistics around SLG and OBP for the year 1960.



Henry Aaron

AB	H	1B	2B	3B	HR	BB	HBP	SF	OB	PA	TB	SLG	OBP
590	172	101	20	11	40	60	2	12	234	664	334	0.566	0.352

SLG = TB / AB = (1B + 2 * 2B + 3 * 3B + 4 * HR) / AB

ActualSLG = 334 / 590 = .566, rtmSLG = (334 + 22) / (590 + 52.5) = .554

Performing the same calculation for OBP for 1960 and knowing that

OBP = OB / PA = (H + BB + HBP) / (AB + BB + HBP + SF)

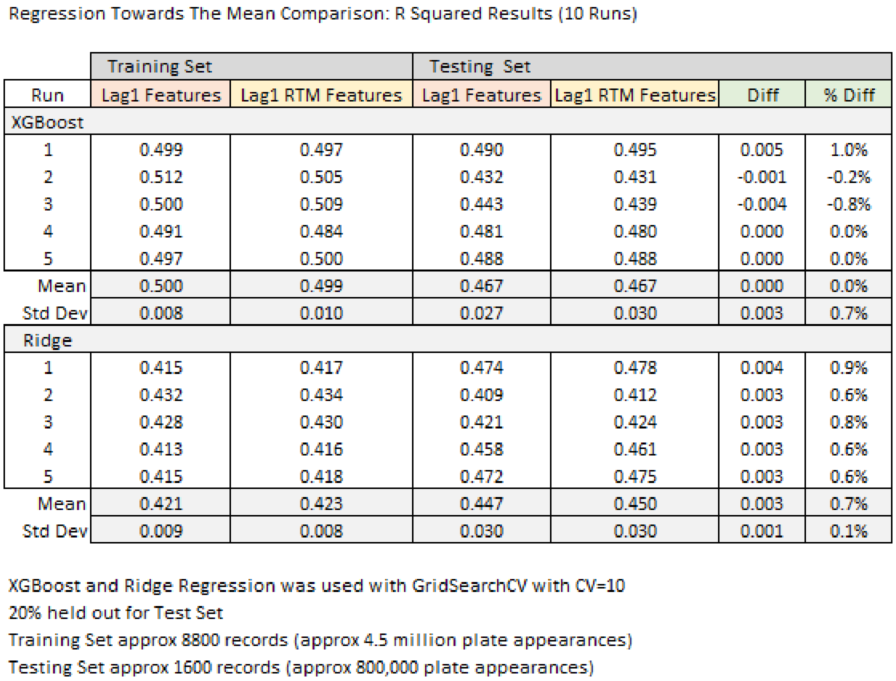
ActualOBP = 234 / 664 = 352, rtmOBP = (234 + 71.4) / (664 + 209.8) = .350

Since Henry Aaron`s stats are higher than the league average (AVG SLG = .419 and AVG OBP = .340), the regression towards the predicted SLG and OBP values are reduced.

We do this for all players for all years from 1960 to 2018 using the regression towards the mean values for each year.

The machine learning algorithms run against the above model were: Linear Regression, Non-Linear Regression, XGBoost, Random Forests, Ridge Regression, Lasso Regression and SVM using regression towards the mean features. After each machine learning run, the results of the run by player by year are written to an excel spreadsheet for comparison purposes. Here are the results of the average actual OPS vs. Predicted OPS of the entire test set.

In addition, a comparison was performed with non-regression toward the mean features (same features as above without rtm applied) and rtm features with the results of the comparisons below. For the comparisons, the two machines learning algorithms used were XGBoost and Ridge regression. The following are the R-Squared results of 5 runs using the XGBoost algorithm and 5 runs using Ridge regression algorithm.

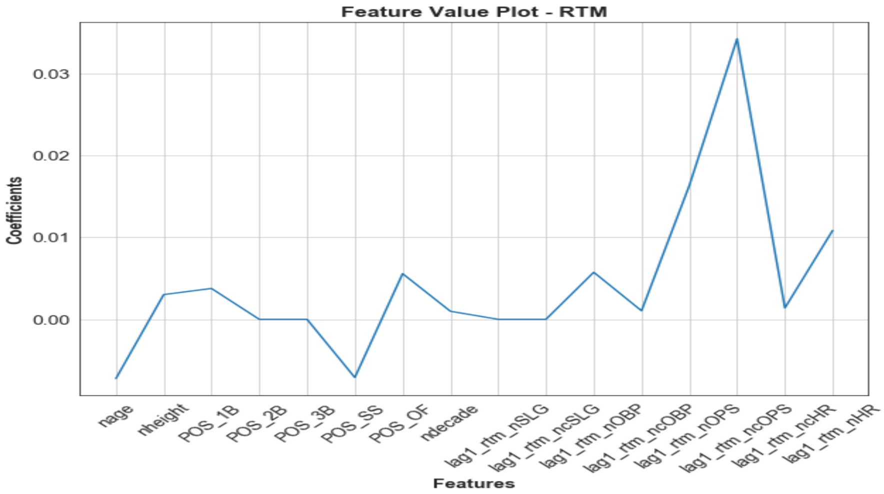


Regression Towards The Mean Comparison: R Squared Results (10 Runs)

	Training Set		Testing Set			
Run	Lag1 Features	Lag1 RTM Features	Lag1 Features	Lag1 RTM Features	Diff	% Diff
XGBoost						
1	0.499	0.497	0.490	0.495	0.005	1.0%
2	0.512	0.505	0.432	0.431	-0.001	-0.2%
3	0.500	0.509	0.443	0.439	-0.004	-0.8%
4	0.491	0.484	0.481	0.480	0.000	0.0%
5	0.497	0.500	0.488	0.488	0.000	0.0%
Mean	0.500	0.499	0.467	0.467	0.000	0.0%
Std Dev	0.008	0.010	0.027	0.030	0.003	0.7%
Ridge						
1	0.415	0.417	0.474	0.478	0.004	0.9%
2	0.432	0.434	0.409	0.412	0.003	0.6%
3	0.428	0.430	0.421	0.424	0.003	0.8%
4	0.413	0.416	0.458	0.461	0.003	0.6%
5	0.415	0.418	0.472	0.475	0.003	0.6%
Mean	0.421	0.423	0.447	0.450	0.003	0.7%
Std Dev	0.009	0.008	0.030	0.030	0.001	0.1%

XGBoost and Ridge Regression was used with GridSearchCV with CV=10
20% held out for Test Set
Training Set approx 8800 records (approx 4.5 million plate appearances)
Testing Set approx 1600 records (approx 800,000 plate appearances)

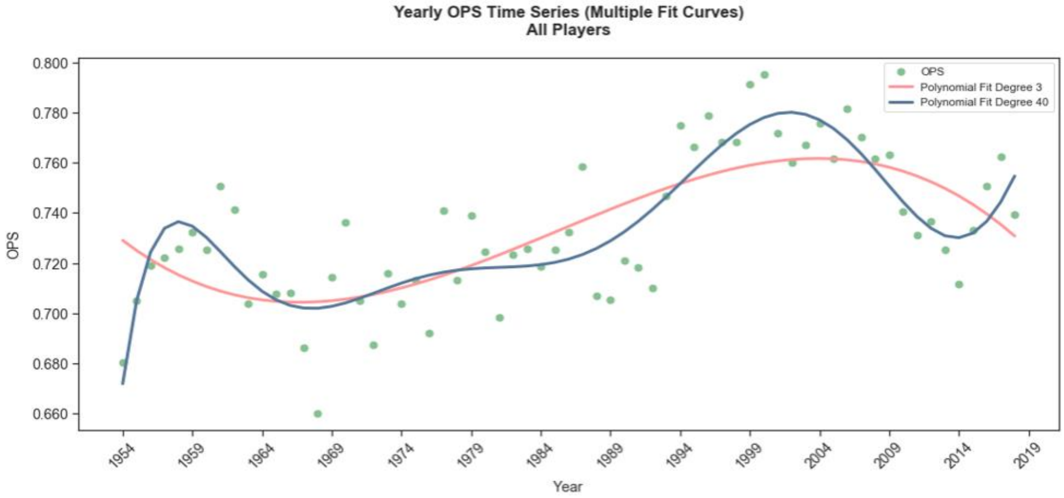
As you can see the regression towards the mean had no real effect on the results for XGBoost and may have had a slight improvement on Ridge Regression. This can be explained by looking at the Lasso feature usage plot below. Note that career OPS has a major effect on the weighting of importance of features.



The results of rtm on career OPS have very little effect due to the size of the numbers. For example, Henry Aaron in year 6 had a career number of plate appearances of 3,988 and got on base a total of 1,519 times. The rtm for year 6 in his career would be (1519 + 71.4) / (3988 + 209.8) which is .378 vs his actual OBP of .381, for SLG total bases (TB) was 2040 and at bats was 3556 for a SLG of .574. The rtm for SLG is .571. Actual career OPS for year 6 is .955 vs .949 which is less than a percent difference. The more years played, the less impact rtm has.

`Steroid Era` Analysis

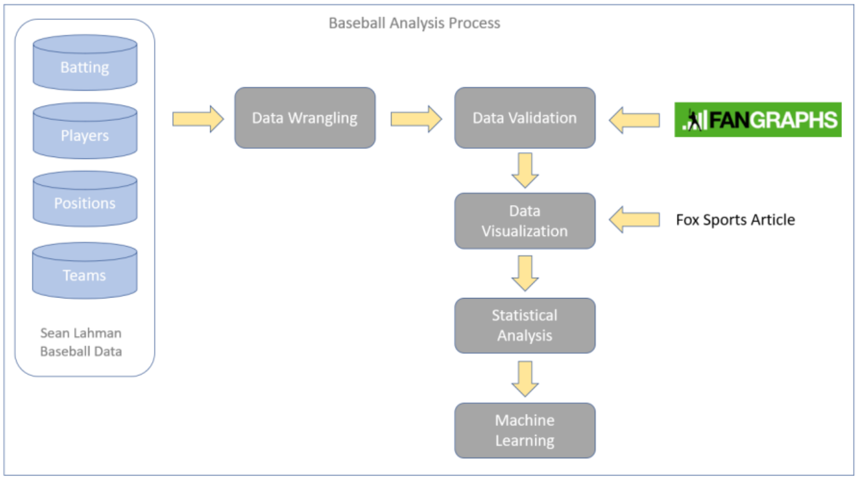
The `steroid era` in baseball was a dark time for MLB baseball. Even today some of the greatest players of all time are not in MLB`s Hall of Fame because of suspected steroid use. Mark McGuire is one such example. There is no exact start and end date to the steroid era. ESPN has defined it as starting in the later 1980`s and ending in the late 2000`s. Others have defined it to be from 1993 to 2003. The latter definition will be used for this project. In 2003, MLB introduced performance enhancing drug testing. Analysis of hitting performance of the pre-steroid era (1982 to 1992), the steroid era and post-steroid era (2004 to 2014) will be performed. I would like to know if the steroid era conclusively showed a significant advantage to player`s performance using OPS as a metric. The figure below shows a times series plot with two fitted curves, one overfitted with polynomial degrees 40 and one properly fitted with polynomial degree 3.



My customer is analysts who refuse to let players who took (or suspected of taking) performance enhancing drugs (PEDs) into the hall of fame because of unfair advantage. It is also for fans who want more information on the subject. Did they really have an advantage? A lot of work in this area has already been done, but I wanted to know for myself. The above figure shows an interesting trend. From about 1970, the OPS performance numbers gradually increased up until about 2009. This trend did not start in the steroid era, it started well before that. You could argue that starting in the early 1970`s, the performance numbers gradually started to increase. In early 1990`s it looks as if the performance numbers accelerated a bit. However, from my research, there isn`t enough evidence to conclusively say the steroid era players had an advantage. The OPS performance numbers started increasing well before that era.

Stages of the End-To-End Process

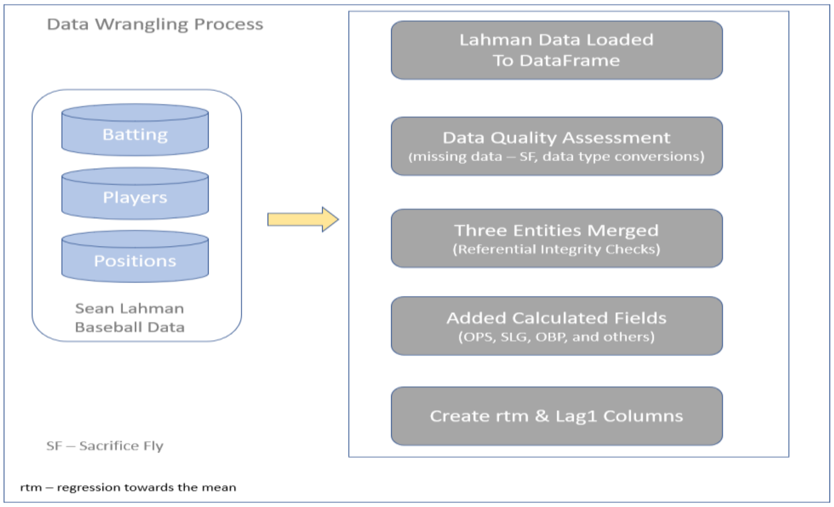
There are five main components of the baseball analysis process:



The five steps are Data Wrangling, Data Validation, Data Visualization, Statistical Analysis and Machine Learning.

Data Wrangling

The first step is Data Wrangling. Data was downloaded from the Sean Lahman site and staged for loading. There were three main data entities that were loaded: Batting, Player and Position data. Why only use data from 1954 when data was available from 1871 onwards? The sacrifice fly was not tracked consistently until 1954. According to Wikipedia, `batters have not been charged with a time at-bat for a sacrifice hit since 1893, but baseball has changed the sacrifice fly rule multiple times. The sacrifice fly as a statistical category was instituted in 1908, only to be discontinued in 1931. The rule was again adopted in 1939, only to be eliminated again in 1940, before being adopted for the last time in 1954`. Sacrifice flies are required for OPS calculation, and for this reason only data from 1954 onward is used. Here is the process for Data Wrangling:


Data Wrangling Process
Batting
Players
Positions
Sean Lahman Baseball Data
Lahman Data Loaded To DataFrame
Data Quality Assessment (missing data – SF, data type conversions)
Three Entities Merged (Referential Integrity Checks)
Added Calculated Fields (OPS, SLG, OBP, and others)
Create rtm & Lag1 Columns
SF – Sacrifice Fly
rtm – regression towards the mean

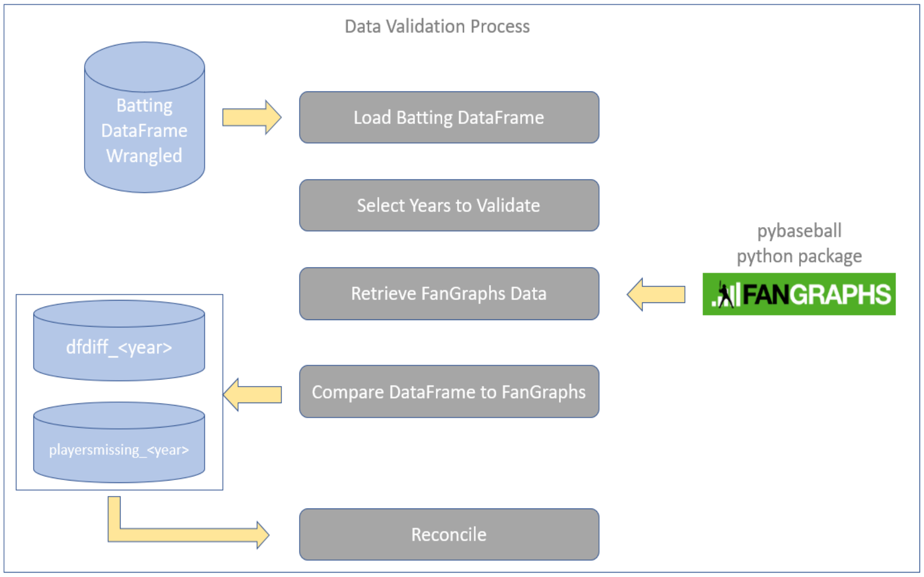
A data quality assessment was performed, the three entities were joined, additional columns were added and the DataFrame was written to a CSV file to be used in downstream processes. Here is a link to the code bases:

GitHub Data Wrangling Code , Data Wrangling rtm Code , Data Wrangling Lag Code

If you get an error, it is likely caused by `GitHub API rate limit exceeded`. You`ll have to try later.

Data Validation

The next step in the process is Data Validation. The following diagram defines the data validation process. In order to independently validate the Lahman data after all the data wrangling was performed, the FanGraphs API was used. The pybaseball package integrated the FanGraph API. All that was needed was a function call which implemented the API which was provided by the pybaseball package.

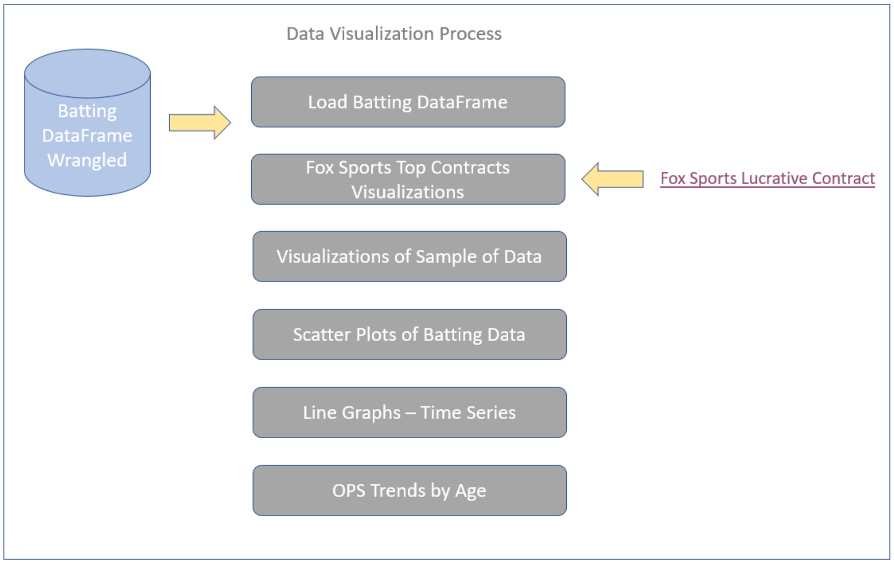


All the wrangled Lahman data was successfully reconciled using FanGraphs. Here is a link to the Data Validation code:

GitHub Data Validation Code

Data Visualization

The third step in the process was Data Visualization. During this step, EDA was performed. To make it interesting, a Fox Sports article listed the top contracts in MLB which listed the dollar amount of the contract, the duration of the contract and when it was signed. This data was manually entered into a spreadsheet, loaded and integrated with the wrangled Lahman data. The following flows the overall data visualizations performed:


Data Visualization Process
Batting DataFrame Wrangled
Load Batting DataFrame
Fox Sports Top Contracts Visualizations
Fox Sports Lucrative Contract
Visualizations of Sample of Data
Scatter Plots of Batting Data
Line Graphs – Time Series
OPS Trends by Age

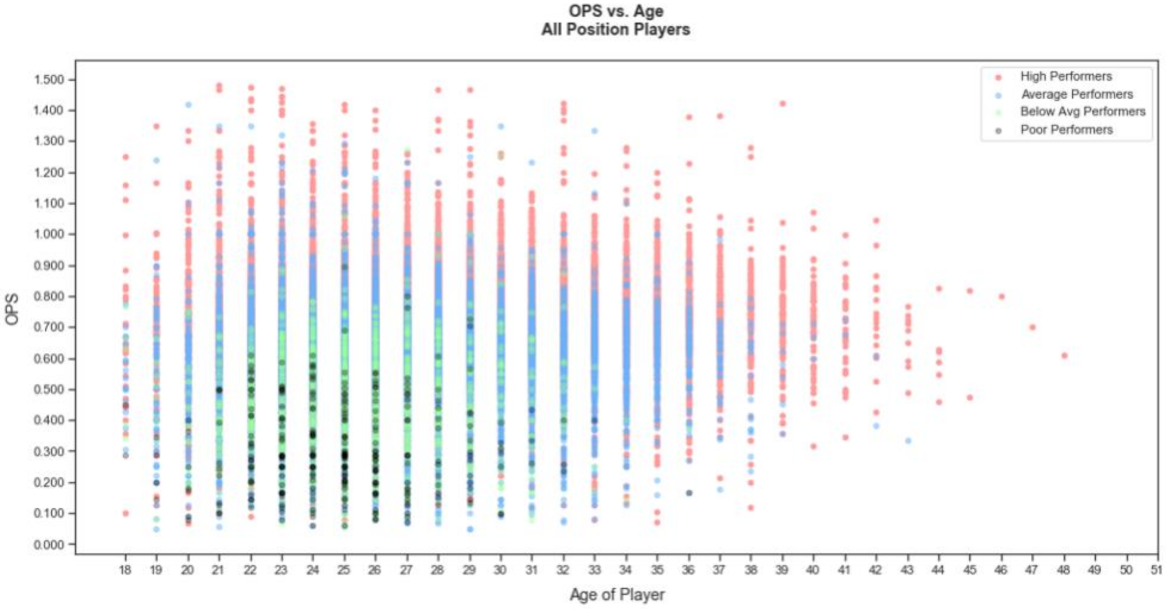
Here are the links to the Fox Sports article and the link to the data visualization Jupyter Notebook.

Fox Sports Article:
Fox Sports MLB`s Most Lucrative Contracts

Jupyter Notebook – Data Visualizations

GitHub Exploratory Data Analysis

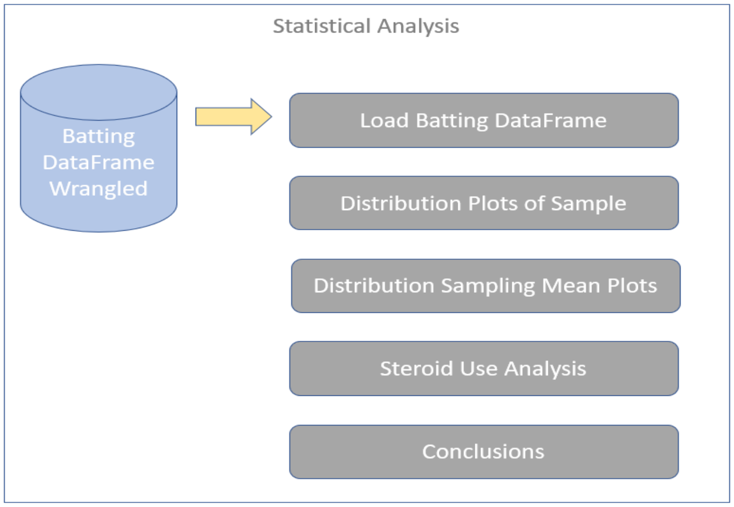
Below is the scatter plot of all MLB players from 1954 to 2018. Note the bands of colors representing different categories of players.



Initially, with one single color it was very difficult to visualize anything. You can already observe in the color bands that there is an upswing in performance of each category of player and then a downswing of performance over their career.

Statistical Analysis

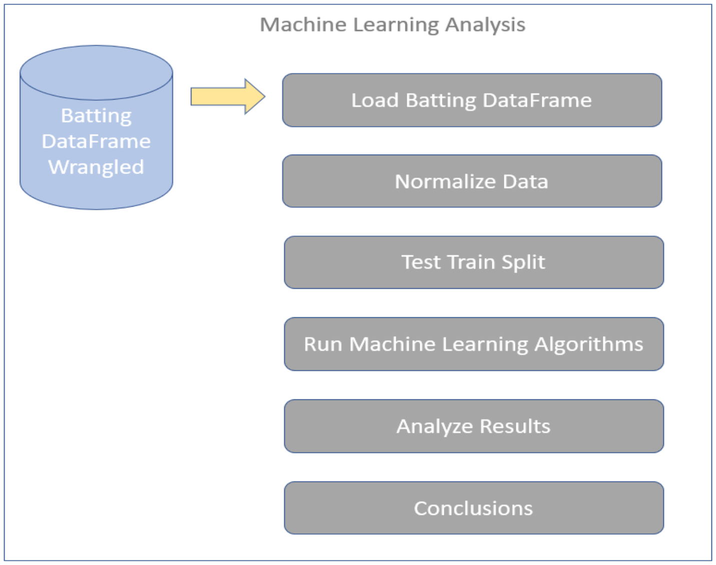
To further my understanding and answer my questions, statistical analysis was performed. The following summarizes the analysis steps as follow:


Statistical Analysis
Batting DataFrame Wrangled
Load Batting DataFrame
Distribution Plots of Sample
Distribution Sampling Mean Plots
Steroid Use Analysis
Conclusions

The Jupyter Notebook with the full analysis can be found by clicking on the following link: GitHub Statistical Analysis

Machine Learning Analysis

The machine learning process is as follows:

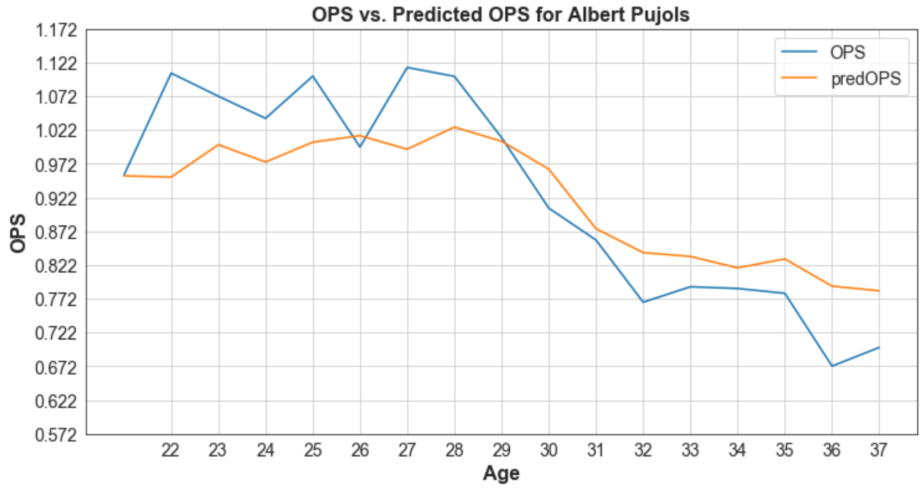

Machine Learning Analysis
Batting DataFrame Wrangled
Load Batting DataFrame
Normalize Data
Test Train Split
Run Machine Learning Algorithms
Analyze Results
Conclusions

The initial process starts the loading of the wrangled batting data in csv format into a dataframe. After that features are normalized, and the test train sets are created from a custom test train split function. After that the machine learning algorithms are run and analysis is provided as to the success of each run. The following link takes you to the GitHub Jupyter Notebook for all of the machine learning runs.

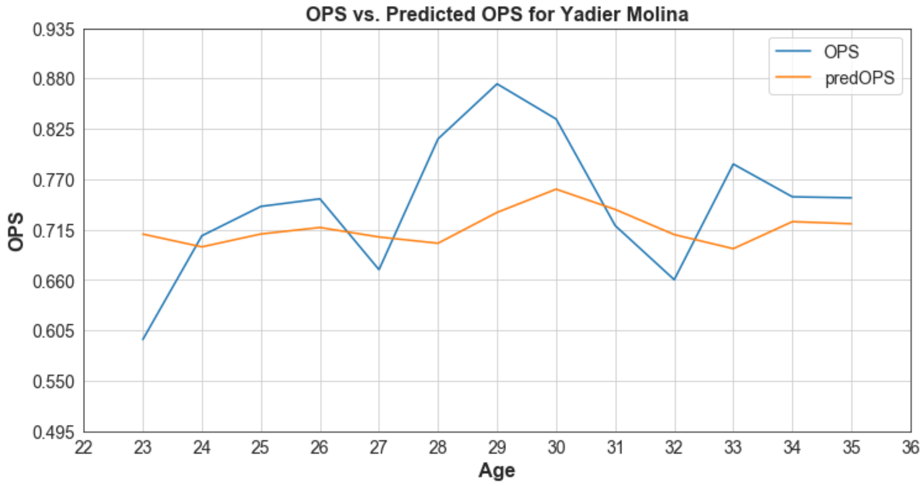
GitHub Machine Learning Jupyter Notebook

There are three charts that I would like to highlight as part of the statistical analysis: Yadier Molina, Albert Pujols and Paul Goldschmidt performance plots.

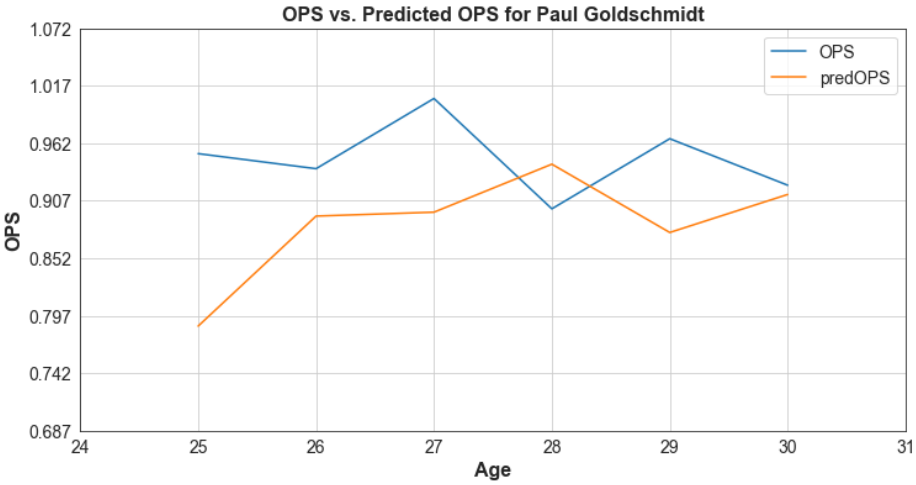
```
Player Name: Albert Pujols
Avg Actual OPS 0.927
Avg Predicted OPS 0.921
Pct Error 0.6
```



```
Player Name: Yadier Molina
Avg Actual OPS 0.743
Avg Predicted OPS 0.717
Pct Error 3.5
```



```
Player Name: Paul Goldschmidt
Avg Actual OPS 0.947
Avg Predicted OPS 0.885
Pct Error 6.6
```



The project showed that you can predict future batting performance using OPS. But there are limitations to how accurate you can be. There are too many variables that we have no control over which leads to somewhat varied performance from year to year. The regression towards the mean proved to be insignificant for XGBoost algorithm, but for Ridge Regression showed minor improvement. Career OPS prediction proved to be very accurate and can be used as a means to predict a player`s performance over time.