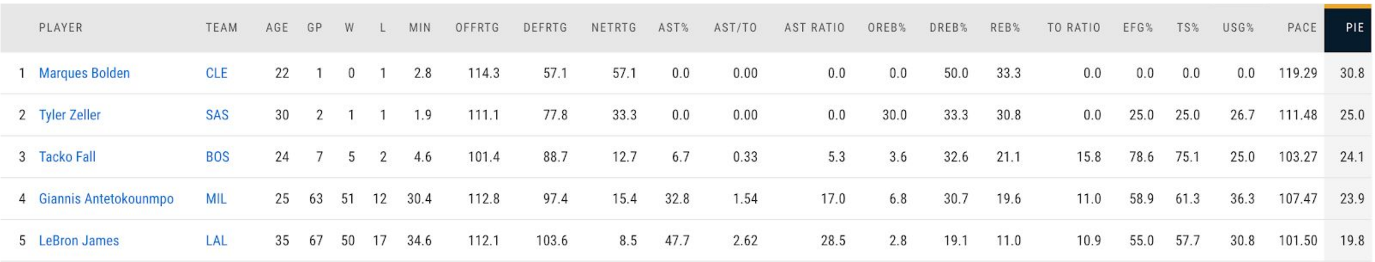
Predicting NBA Player Impact with Advanced NBA Statistics

Problem Statement

Since the 1970's, NBA statistics on player performance had not evolved much, but that changed in 2011. The NBA made a huge leap into the 21st century and joined the data analytics movement started by major league baseball a decade early. The NBA began using state of the art high definition cameras to track player movement to aggregate hundreds of new player metrics that had not been recorded previously. During this same time period, the US Supreme Court lifted a ban on sports gambling that has contributed to a rise in popularity of fantasy sports and sports betting, leading to a greater interest in predicting player performance.

With this project, I was interested to see if there were new insights that could be derived from these new player metrics with the goal of building a linear regression algorithm that could use these advanced NBA statistics to predict a specific player machine the following season. The metric I selected as the target feature was Player Impact Estimate (PIE), which is a statistic that summarizes what proportion of team events (points, rebounds, etc) a player contributes. This is an ideal metric for comparing players among various positions.



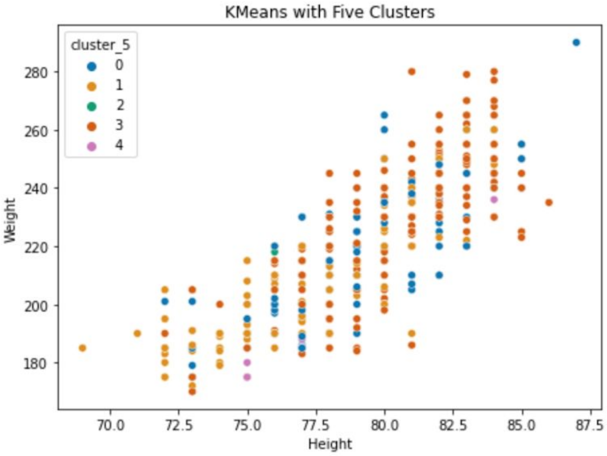
	PLAYER	TEAM	AGE	GP	W	L	MIN	OFFRTG	DEFRTG	NETRTG	AST%	AST/TO	AST RATIO	OREB%	DREB%	REB%	TO RATIO	EFG%	TS%	USG%	PACE	PIE
1	Marques Bolden	CLE	22	1	0	1	2.8	114.3	57.1	57.1	0.0	0.00	0.0	0.0	50.0	33.3	0.0	0.0	0.0	0.0	119.29	30.8
2	Tyler Zeller	SAS	30	2	1	1	1.9	111.1	77.8	33.3	0.0	0.00	0.0	30.0	33.3	30.8	0.0	25.0	25.0	26.7	111.48	25.0
3	Tacko Fall	BOS	24	7	5	2	4.6	101.4	88.7	12.7	6.7	0.33	5.3	3.6	32.6	21.1	15.8	78.6	75.1	25.0	103.27	24.1
4	Giannis Antetokounmpo	MIL	25	63	51	12	30.4	112.8	97.4	15.4	32.8	1.54	17.0	6.8	30.7	19.6	11.0	58.9	61.3	36.3	107.47	23.9
5	LeBron James	LAL	35	67	50	17	34.6	112.1	103.6	8.5	47.7	2.62	28.5	2.8	19.1	11.0	10.9	55.0	57.7	30.8	101.50	19.8

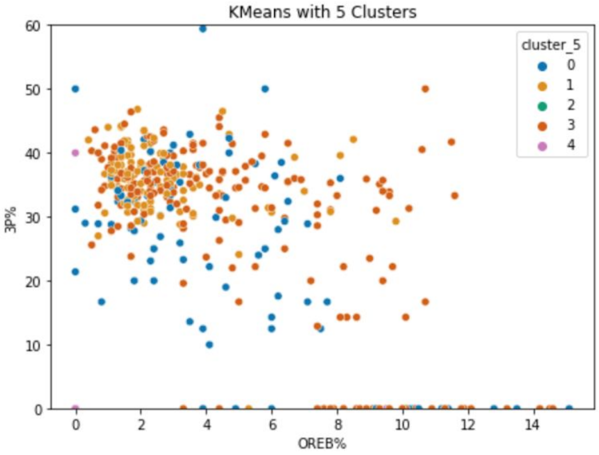
Data Wrangling

The data used for this project was scraped from www.nba.com/stats/ using a combination of Selenium, Beautiful Soup and Pandas. I scraped data from the 2017-2018 season for the explanatory features and the PIE metric from the 2018-2019 season for the response feature. Initial scraping yielded 132 player features for 540 players.

With the granularity of these advanced statistics, many of them showed high collinearity, which would negatively affect our regression model performance. Through an iterative of EDA, collinear variables such as Rebound % and Rebound Chance % were dropped while keeping Adjusted Rebound Chance % which retained much of the information of the dropped statistics.

Curiously, I realized that none of the original 132 features I scraped contained player position. Instead of scraping this feature, I decided to use Scikit-Learn’s K-Means algorithm to cluster players into five groups, based on player performance and not player attributes. Below are two graphs demonstrating those clusters, comparing height vs. weight and Three Point Shooting % vs. Offensive Rebounding %.

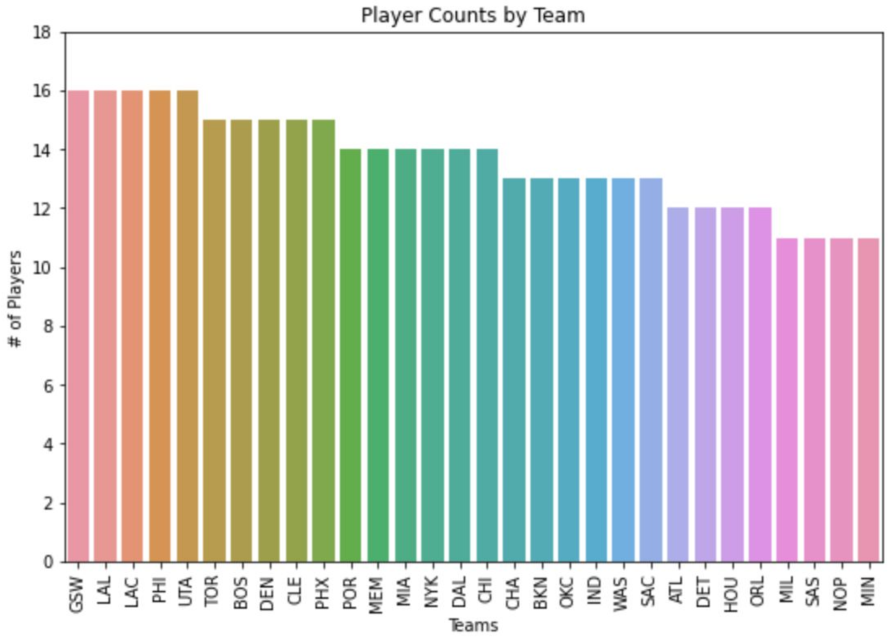




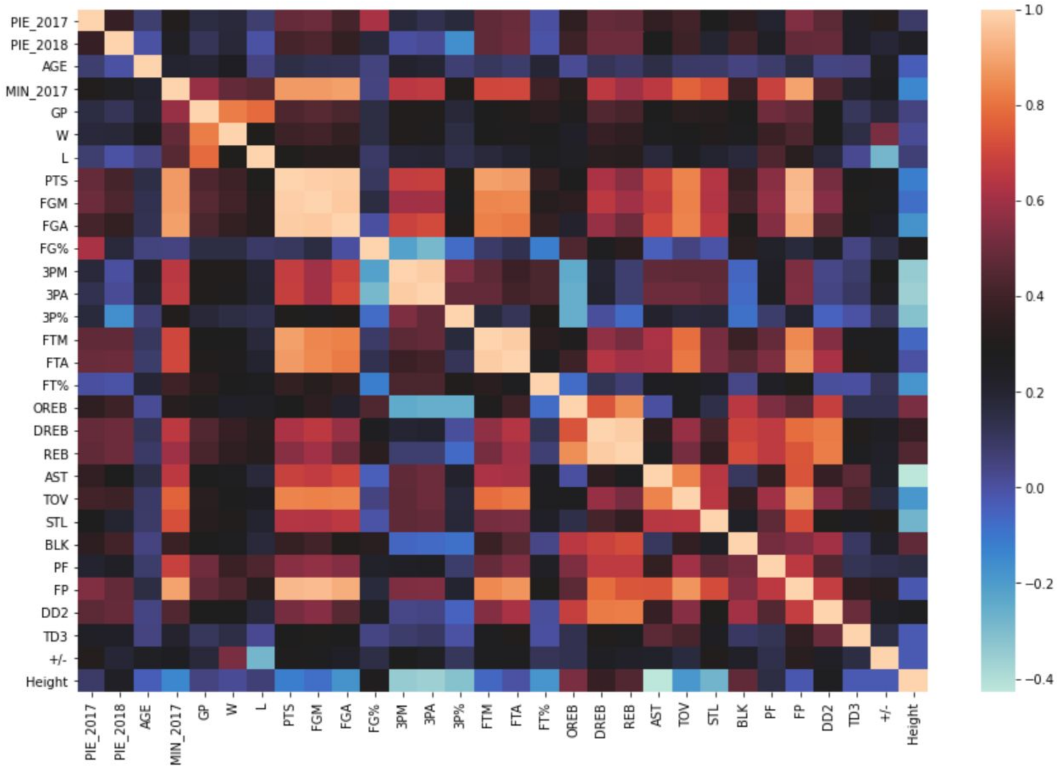
With only 540 players in this dataset, I did not want to remove many players, but some pruning was required. A significant number of players retired after the 2017-2018 season and needed to be removed from the dataset. Additionally, some players left the NBA mid-season to play in international leagues or experience season ending injuries. These exogenous factors were outside of the predictive scope of the regression model. Thus players who played less than three games or had a PIE value of less than one were filtered out of the dataset. After wrangling and pruning, 77 features and 399 players remained to train the regression model.

Exploratory Data Analysis

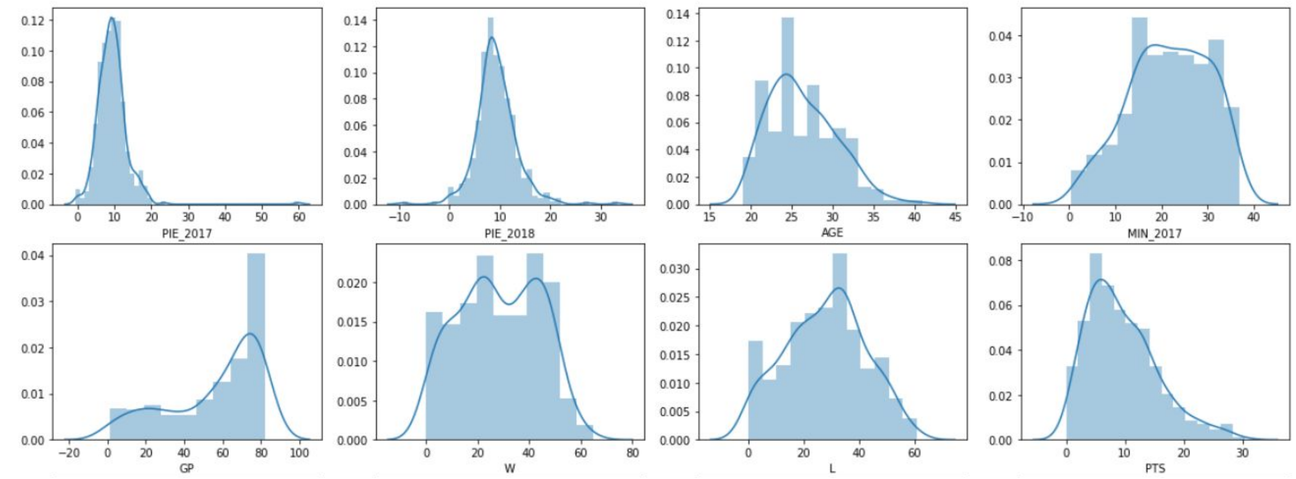
Of all player features, only four of them were categorical variables: Team, Country, College, and the K-Means clusters. Unsurprisingly, players’ country of origin was the USA and more players came straight from high school than from one specific college. More surprisingly, there was a rather large range of roster sizes, even with taking into account the players that had been filtered out from this analysis.



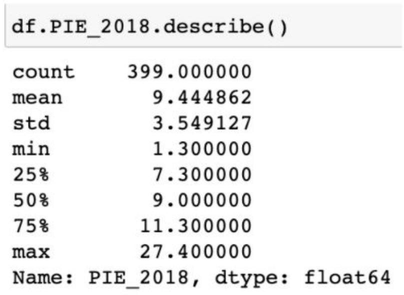
For the remaining numeric variables, I first visually analyzed them with a heatmap of their correlation matrix. Below is a heatmap of the first 30 numeric features. Looking for the light red and light blue colors on this heatmap we can see there are a large number of the features are highly correlated. However, there was too much visual information here to make informed decisions regarding feature pruning, and for that I sorted the values of the correlation matrix to analyze the most highly correlated variables for collinearity.



Next I analyzed the distributions of these numeric variables. Below is a visualization of the probability density functions of the first 8 numeric features. As is evident by looking at these distributions, many of them are skewed and not normally distributed. For better model performance, I transformed all the numeric explanatory features with a Yeo-Johnson transformation before normalizing and standardizing them.

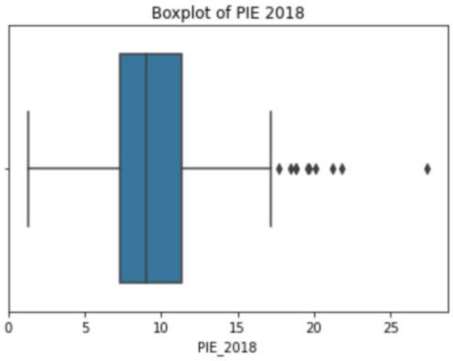


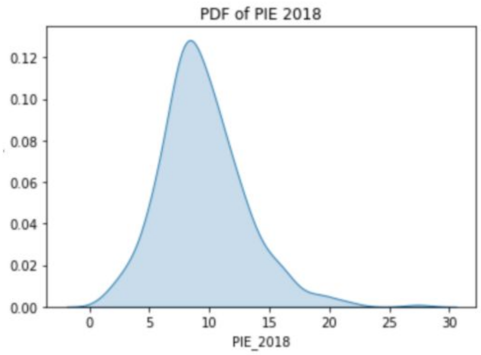
Additionally, I analyzed specifically the target feature Player Impact Estimate(PIE) in the 2018-2019 season. Below are summary statistics, a boxplot, and a probability density function of this variable. It is noteworthy that the mean value of our target variable is 9.44, as we will later see that our regression model has a bias towards predicting a regression towards the mean, which is often accurate but a limitation of the model. It is also notable that the standard deviation is 3.55, which we will later see is significantly higher than our best model’s RMSE score. The PDF on the right visually looks surprisingly normally distributed, however inspecting the boxplot reveals there are a significant number of outliers with high PIE values. These outlier players generally represent all star players, which we will later see that our model had a harder time accurately predicting.



```
df.PIE_2018.describe()

count    399.000000
mean       9.444862
std        3.549127
min        1.300000
25%        7.300000
50%        9.000000
75%       11.300000
max       27.400000
Name: PIE_2018, dtype: float64
```

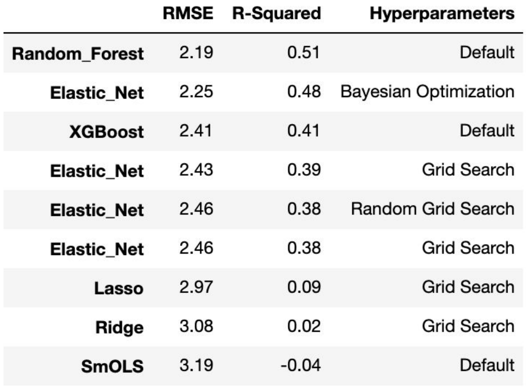




Modeling

In the modeling selection process, I evaluated six different regression algorithms: Statsmodels' OLS, Scikit-Learn's Ridge, Lasso, Elastic Net, Random Forest, and Xgboost. The target metric was minimizing the RMSE score, while taking the R-squared score into consideration, but not of primary importance. For this comparison, I used a basic grid search to tune hyperparameters when applicable.

Random Forest and Xgboost provided predictions with the lowest RMSE, but given the less interpretable nature of these models, I continued tuning hyperparameters of the Elastic Net model. I retrained the Elastic Net model tuning hyperparameters with Random Grid Search and Bayesian Optimization. The best Elastic Net model was tuned with Bayesian Optimization, yielding a RMSE only 0.06 higher than the Random Forest model.

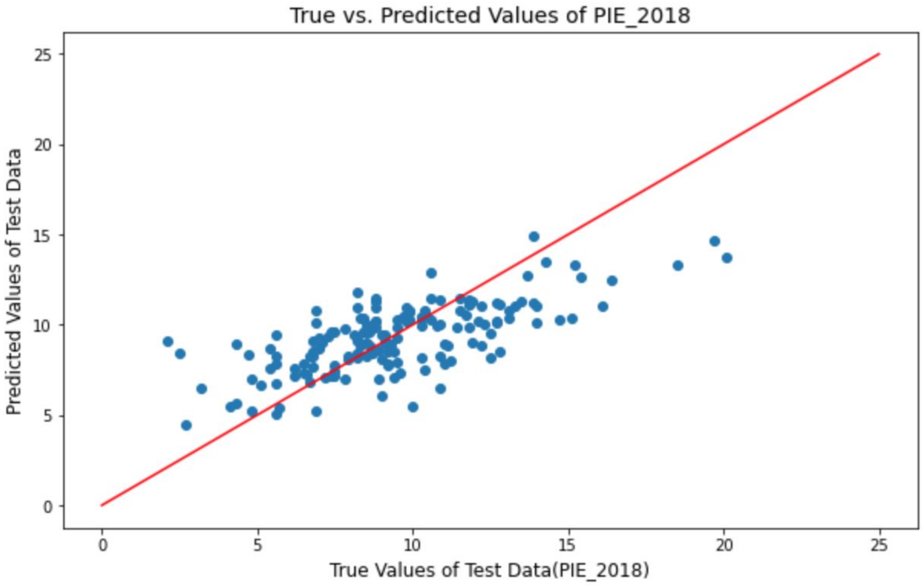


	RMSE	R-Squared	Hyperparameters
Random_Forest	2.19	0.51	Default
Elastic_Net	2.25	0.48	Bayesian Optimization
XGBoost	2.41	0.41	Default
Elastic_Net	2.43	0.39	Grid Search
Elastic_Net	2.46	0.38	Random Grid Search
Elastic_Net	2.46	0.38	Grid Search
Lasso	2.97	0.09	Grid Search
Ridge	3.08	0.02	Grid Search
SmOLS	3.19	-0.04	Default

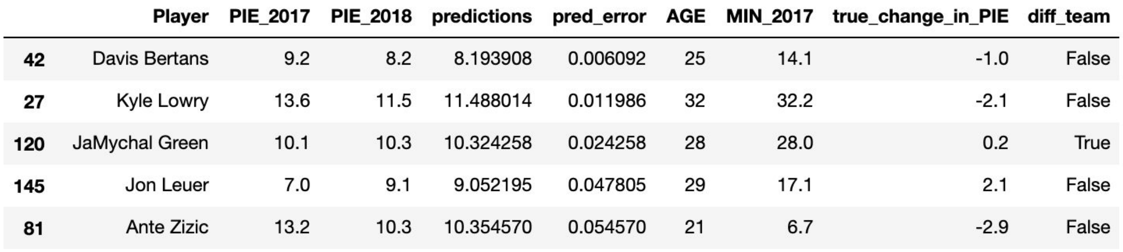
Above is a table of model performance metrics.

Model Analysis

In the graph below of the residuals of the model predictions, we can see that the model was generally overestimating lower true PIE values and underestimating higher true PIE values. This modelling error can be attributed to the model being overly sensitive towards predicting a

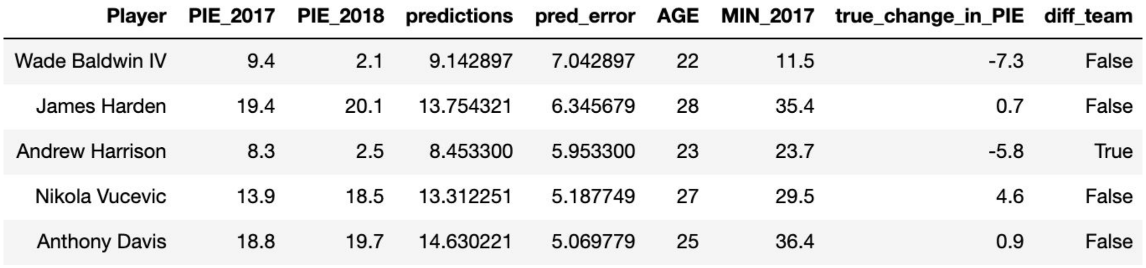


regression towards the mean, and less sensitive to unique player trends, such as an all-star player’s continued growth or an older player’s decline.



	Player	PIE_2017	PIE_2018	predictions	pred_error	AGE	MIN_2017	true_change_in_PIE	diff_team
42	Davis Bertans	9.2	8.2	8.193908	0.006092	25	14.1	-1.0	False
27	Kyle Lowry	13.6	11.5	11.488014	0.011986	32	32.2	-2.1	False
120	JaMychal Green	10.1	10.3	10.324258	0.024258	28	28.0	0.2	True
145	Jon Leuer	7.0	9.1	9.052195	0.047805	29	17.1	2.1	False
81	Ante Zizic	13.2	10.3	10.354570	0.054570	21	6.7	-2.9	False

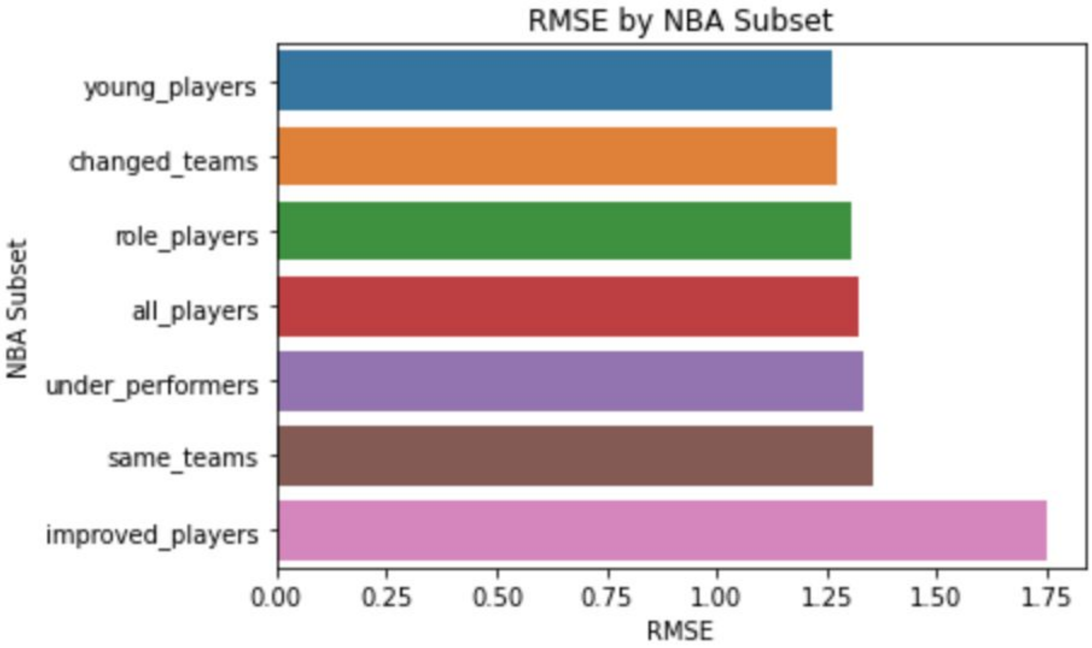
Above is a table of the models most accurate predictions.



Player	PIE_2017	PIE_2018	predictions	pred_error	AGE	MIN_2017	true_change_in_PIE	diff_team
Wade Baldwin IV	9.4	2.1	9.142897	7.042897	22	11.5	-7.3	False
James Harden	19.4	20.1	13.754321	6.345679	28	35.4	0.7	False
Andrew Harrison	8.3	2.5	8.453300	5.953300	23	23.7	-5.8	True
Nikola Vucevic	13.9	18.5	13.312251	5.187749	27	29.5	4.6	False
Anthony Davis	18.8	19.7	14.630221	5.069779	25	36.4	0.9	False

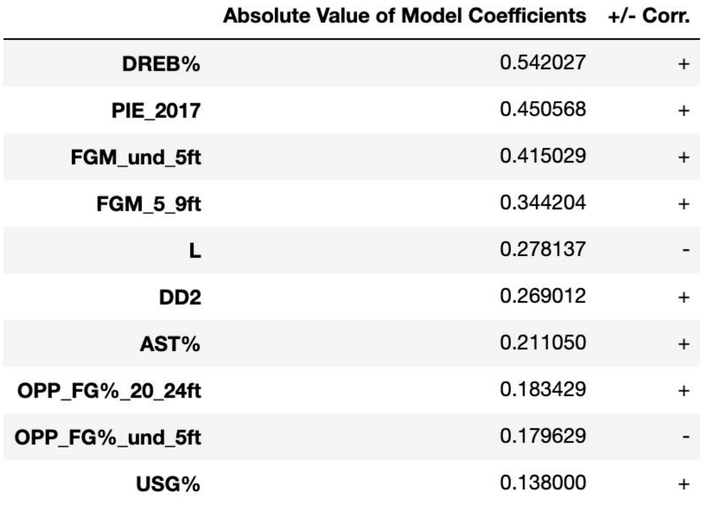
Above is a table of the model’s least accurate predictions.

From these two tables of the most and least accurate predictions, we continue to see this trend of the model predicting a regression towards the mean. Additionally we can see that the model is generally more accurate at predicting players with moderate growth/regression, rather than predicting outlier player’s progress. Looking at different subsets of players, we can



also see the model performed better on young role players who changed teams, and the model performed worse on the improved players who stayed on the same team.

Analyzing the model's coefficients in the table below, we can derive a number of insights about other statistics that are highly correlated with an increase in PIE value in the upcoming season.



	Absolute Value of Model Coefficients	+/- Corr.
DREB%	0.542027	+
PIE_2017	0.450568	+
FGM_und_5ft	0.415029	+
FGM_5_9ft	0.344204	+
L	0.278137	-
DD2	0.269012	+
AST%	0.211050	+
OPP_FG%_20_24ft	0.183429	+
OPP_FG%_und_5ft	0.179629	-
USG%	0.138000	+

Above is a table of coefficient weights of the model.

Surprisingly, Defensive Rebound % was the highest correlated feature with an increase in PIE, which could be attributed to the best rebounders getting more playing time the following season. The third and fourth most correlated features were Field Goals Made under five feet and between five and nine feet. This predictive relevance could be attributed to the ability to high volume short range scorers tended to be more involved in a greater variety of team statistics than long range scorers. Additional insights from these coefficients would suggest players on winning teams with a high Assist % and good short range defense are strong predictors of that player's PIE score the following season.

Business Applications/Recommendations

1. Identify role/bench players projected to show moderate improvement (10-20%) the following season.
2. Inform decision to not resign or offer less money to role/bench players that are projected to regress the following season.
3. Utilize features that were the most impactful coefficients of the regression model to identify growth players.

Future Development

I would like to add more data with which to train this model. This would require scraping data from additional NBA seasons. Ideally this would be done with an automated web scraper. Unfortunately there are enough tabular irregularities in the website that it would prove difficult to automate this cleanly, but it is something I would like to research more.

Additionally, I would like to develop this model into an app that would allow a user to input a specific player and target features and receive an output prediction for the following season.

The repo for this project can be found here on github.