Comparing How NBA Player Value relates to Salary

Corey Becker

Section B

11/7/24

Umut Mete Saka

Introduction

In 2023, The NBA changed the rules on NBA salary caps and introduced new levels of salary caps that apply harsher restrictions on NBA teams that exceed the Soft salary cap (luxury tax line). This means NBA teams need to be very strategic on how they pay players who they give the big contracts to and who they let go. These new changes have disproportionately affected NBA teams who are contending for a championship, causing them to struggle to maintain the rosters that made them so successful. In my opinion, this has harmed 3 NBA teams the most, The Warriors, the Celtics, and my beloved Nuggets, which also happen to be the last 3 NBA champions. Helping NBA teams understand what statistics are worth the most money can help them create better-informed roster decisions, and help players understand their worth and improve their salaries, especially young players working towards their first big contract.

My goal is to create an ML model that can predict the salary an NBA player deserves based on their statistics, which can help teams find high-value players for a lower cost, and pay them more than other teams would. I will also fit a model to predict a player's Value Over Replacement Player (VORP), which is a statistic that attempts to quantify the impact a player has on their team. I will compare the two models to find the differences between how teams allocate salary vs how they are valued according to VORP. This is similar to the baseball concept of MoneyBall, in which the Oakland A's found statistics that greatly contributed to winning but teams didn't value, and they were able to construct a team full of incredibly skilled players for a much lower salary than other teams.

The VORP statistic, according to basketball reference, is " a box score estimate of the points per 100 TEAM possessions that a player contributed above a replacement-level (-2.0) player, translated to an average team and prorated to an 82-game season. Multiply by 2.70 to convert to wins over replacement." This statistics is a valuable tool to quantify the value of an individual player based on their performance. Both salary and VORP are a means of valuing a player, so understanding what statistics are most important to these values can help a team determine what statistics are currently over/under valued. This can help teams better evaluate how players are paid and find undervalued players that can over perform for their salaries.

For this project to be successful, both models need to accurately reflect the current NBA landscape and effectively distinguish between how VORP (Value Over Replacement Player) and salary are calculated. This distinction can help identify which statistics are overvalued or undervalued. The insights gained would be highly valuable for NBA general managers working to stay under the salary cap, as they could use this information to target undervalued players. Additionally, players could leverage these findings to argue that their value is not fully reflected in their current salary. Ultimately, my goal is to re-examine how NBA contracts are structured and challenge traditional perceptions of player value by applying machine learning and advanced statistics.

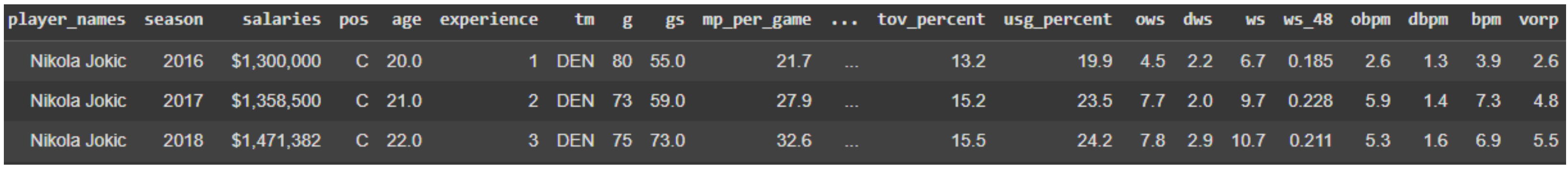
Technical Details

Dataset

I did not choose a Kaggle dataset, rather I am using my own modified data set. The dataset contains NBA player data and player salaries from 1991 to the last NBA season. This dataset is based on the NBA stats (1947-present) database created by Sumitro Datta, which is a web scrape of player statistics from basketball-reference.com. I then expanded on this database by inserting NBA player salaries and adjusted salaries in 2023 dollars, by web scraping these salaries from HoopsHypes, which is a USA Today sports site.

My data had statistics for each NBA player from 1990 to 2024, which includes player information (age, experience, position, etc.), per game statistics, salary, and advanced statistics including VORP. Each datapoint (row) in my dataset is an individual season for a player, so we have all the statistics and the salary for all 21 years of LeBron James's career. Each model used a different subset of features which were determined using a best subsets method.

Example data



player_names	season	salaries	pos	age	experience	tm	g	gs	mp_per_game	...	tov_percent	usg_percent	ows	dws	ws	ws_48	obpm	dbpm	bpm	vorp
Nikola Jokic	2016	$1,300,000	C	20.0	1	DEN	80	55.0	21.7	...	13.2	19.9	4.5	2.2	6.7	0.185	2.6	1.3	3.9	2.6
Nikola Jokic	2017	$1,358,500	C	21.0	2	DEN	73	59.0	27.9	...	15.2	23.5	7.7	2.0	9.7	0.228	5.9	1.4	7.3	4.8
Nikola Jokic	2018	$1,471,382	C	22.0	3	DEN	75	73.0	32.6	...	15.5	24.2	7.8	2.9	10.7	0.211	5.3	1.6	6.9	5.5

Data Wrangling & Cleaning

Player Name Regularization:

In order to merge the different statistical datasets and the salary data on player name and season, I needed to ensure that player names matched across the datasets. The major problem was player names with Diacritics, such as Nikola Jokić, as the salary dataset only used standard characters, so I had to regularize all their names to not include the Diacritics.

Salary regularization:

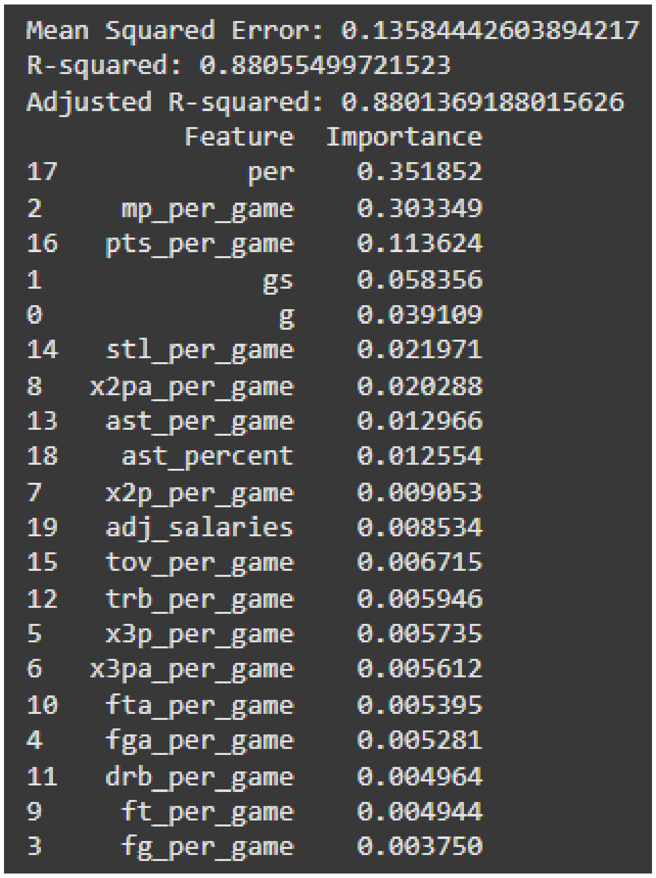
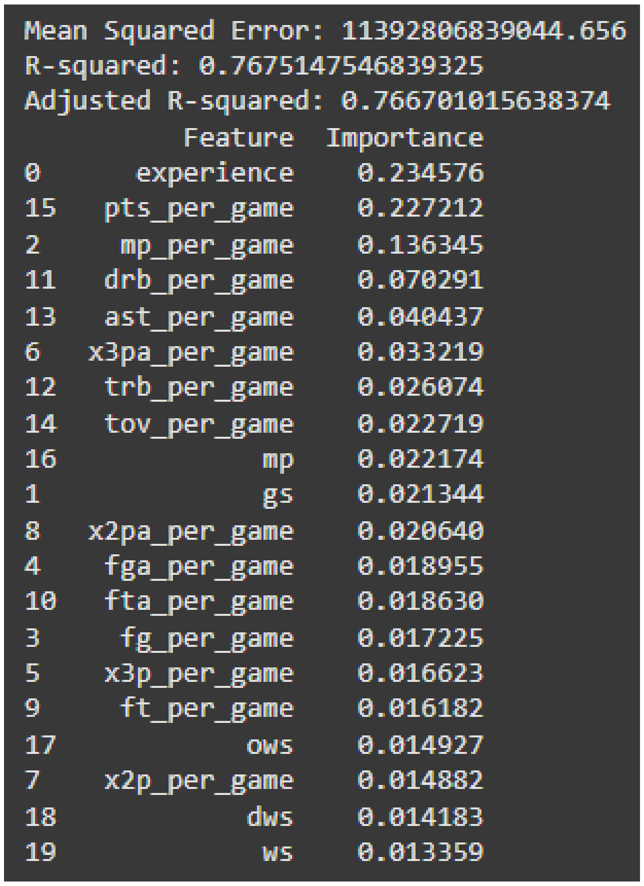
I regularized player's salaries to account for inflation, as a $20 million contract in 1990 is equivalent to a $48 million contract in 2024. To do this, I used the CPI library. This was necessary to ensure that the results of my regression model would be consistent across the years.

Null shooting percentage fixing:

A lot of players had a NaN for their shooting percentage, which was due to the player not attempting any of those shots. I converted all the NaN's into zeros. This isn't perfectly accurate as these players haven’t attempted a shot, but this is the only way to keep their data. This was especially important for older years where three pointers were less common, so players like Shaquille O'Neal would stay in the data.

Models

I created 2 random forest models for both VORP and Salary. For both models, I used 20 features, which was the smallest number of features with a good adjusted R2 for both models. I will judge the models based on which coefficients are most important, using the Gini Importance metric for each feature.

VORP Random Forest Model	**Salary Random Forest Model**
	

```
Mean Squared Error: 0.13584442603894217
R-squared: 0.88055499721523
Adjusted R-squared: 0.8801369188015626
           Feature  Importance
17             per    0.351852
2      mp_per_game    0.303349
16    pts_per_game    0.113624
1               gs    0.058356
0                g    0.039109
14    stl_per_game    0.021971
8    x2pa_per_game    0.020288
13    ast_per_game    0.012966
18     ast_percent    0.012554
7     x2p_per_game    0.009053
19    adj_salaries    0.008534
15    tov_per_game    0.006715
12    trb_per_game    0.005946
5     x3p_per_game    0.005735
6    x3pa_per_game    0.005612
10    fta_per_game    0.005395
4     fga_per_game    0.005281
11    drb_per_game    0.004964
9      ft_per_game    0.004944
3      fg_per_game    0.003750
```

```
Mean Squared Error: 11392806839044.656
R-squared: 0.7675147546839325
Adjusted R-squared: 0.766701015638374
           Feature  Importance
0       experience    0.234576
15    pts_per_game    0.227212
2      mp_per_game    0.136345
11    drb_per_game    0.070291
13    ast_per_game    0.040437
6    x3pa_per_game    0.033219
12    trb_per_game    0.026074
14    tov_per_game    0.022719
16              mp    0.022174
1               gs    0.021344
8    x2pa_per_game    0.020640
4     fga_per_game    0.018955
10    fta_per_game    0.018630
3      fg_per_game    0.017225
5     x3p_per_game    0.016623
9      ft_per_game    0.016182
17             ows    0.014927
7     x2p_per_game    0.014882
18             dws    0.014183
19              ws    0.013359
```

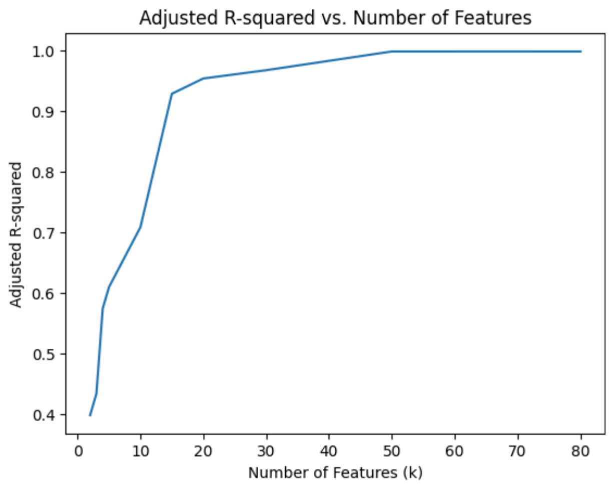
We can see that both models have similar features that are most important, but the most important feature for each is missing in the other.

Training

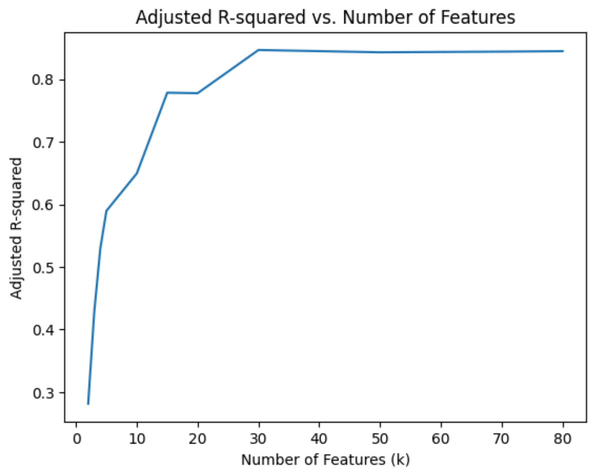
For both models, I did the same training steps. First, I used one-hot encoding to turn categorical variables like team and position into numeric variables that I could use. Second, I used feature selection to reduce my input parameters, for both models I used the Select K best function, and used adjusted R2 as the metric to maximize. I did this to ensure both models have a reasonable amount of parameters, and I can compare the 15 most important parameters between the two models. Lastly, I did a train test split as I had over 20,000 data points to use, I withheld 20% of the data for testing. I used feature selection to enable me to compare the coefficients between the two models and accurately what was most important to predicting VORP/Salary. I also had to get rid of all the win share features for the vorp model as VORP and win shares are all the same statistic with different scalars.

As you can see from the plot below, 20 features is the elbow point for adjusted R2, more features would lead to overfitting and excess computations for minimal results

VORP



Salary

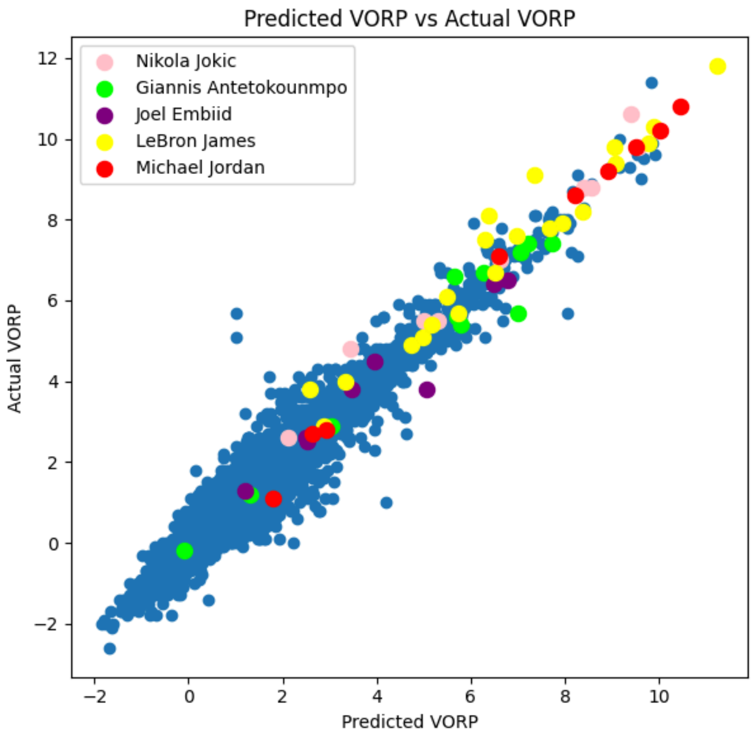


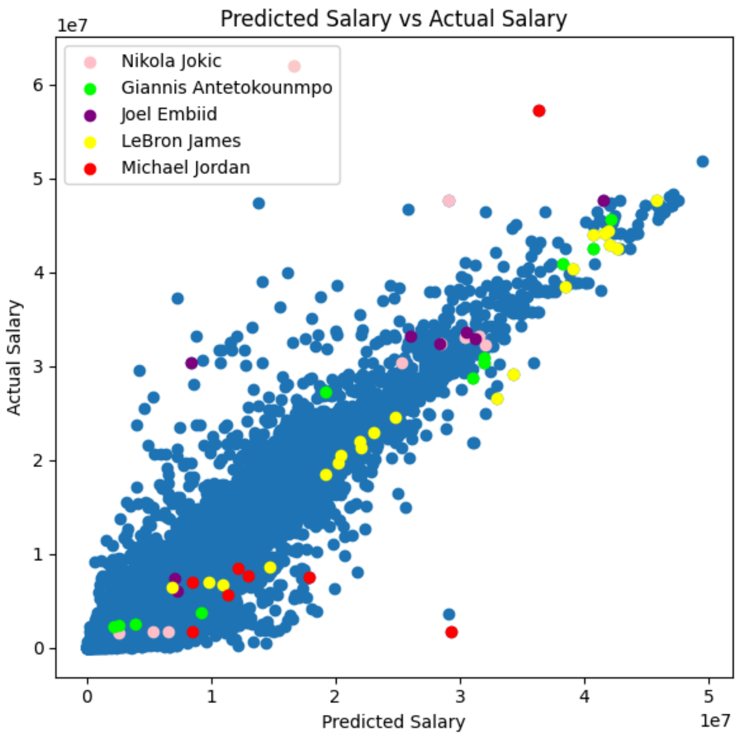
Performance

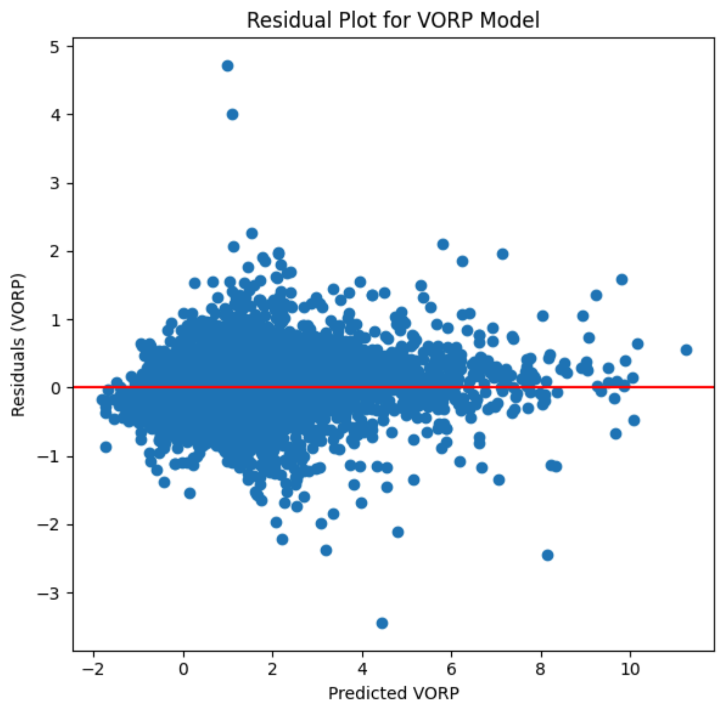
Most metrics are available in the model summaries above but here are a few more!

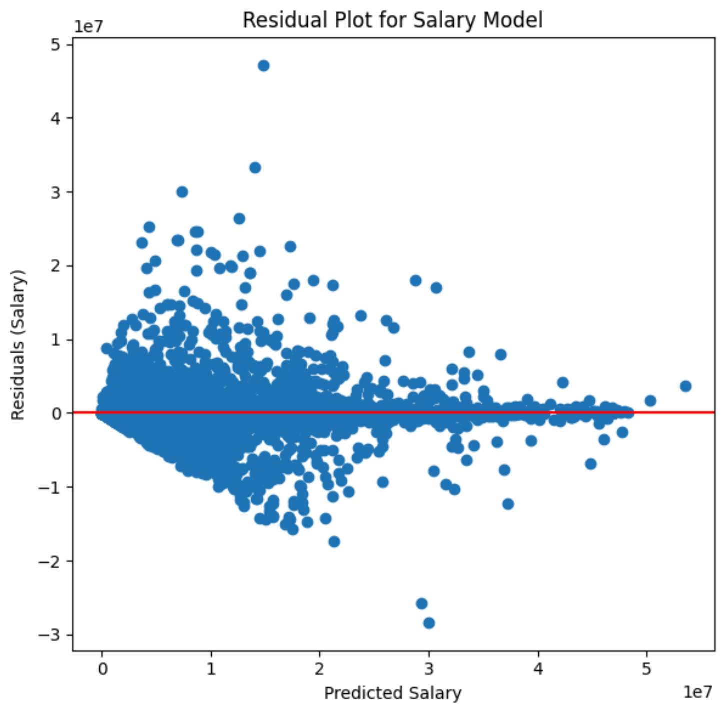
Metrics	VORP	Salary
Out of Bag Error	0.9625	0.9529
sMAPE	0.6069	0.1093

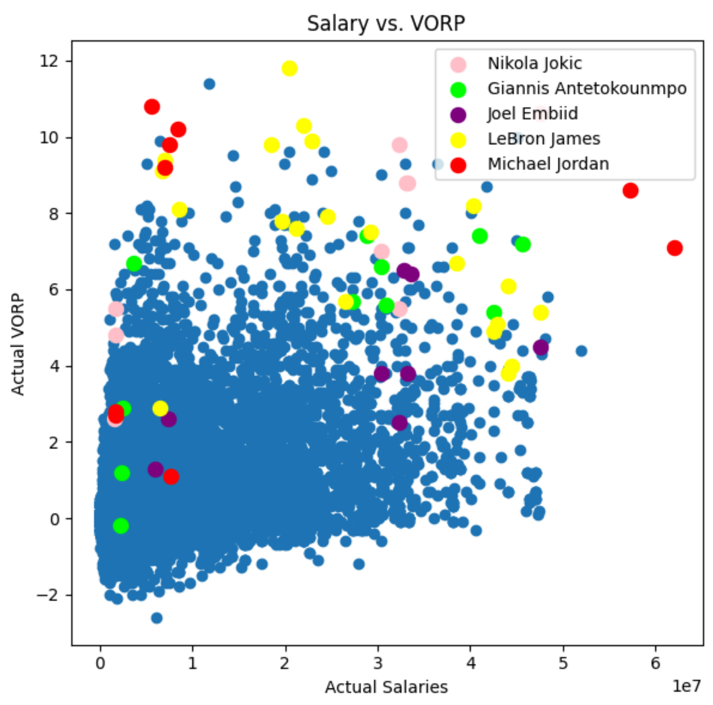
Plots

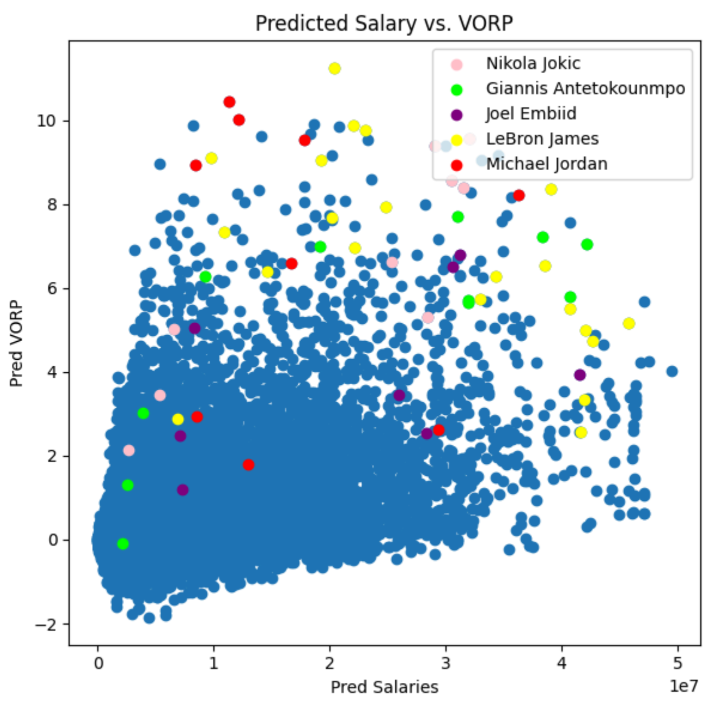

Predicted VORP vs Actual VORP
Nikola Jokic
Giannis Antetokounmpo
Joel Embiid
LeBron James
Michael Jordan
Actual VORP
Predicted VORP

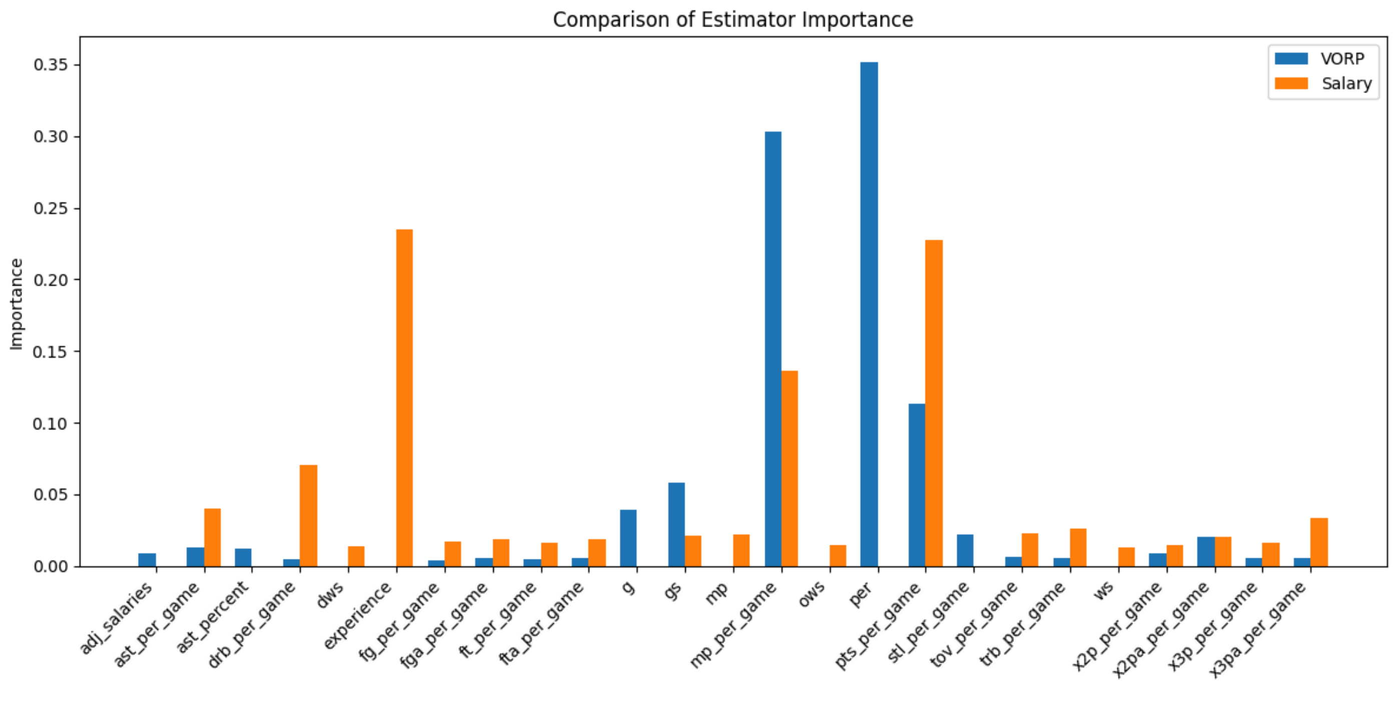

Predicted Salary vs Actual Salary
1e7
Nikola Jokic
Giannis Antetokounmpo
Joel Embiid
LeBron James
Michael Jordan
Actual Salary
Predicted Salary
1e7


Residual Plot for VORP Model
Residuals (VORP)
Predicted VORP

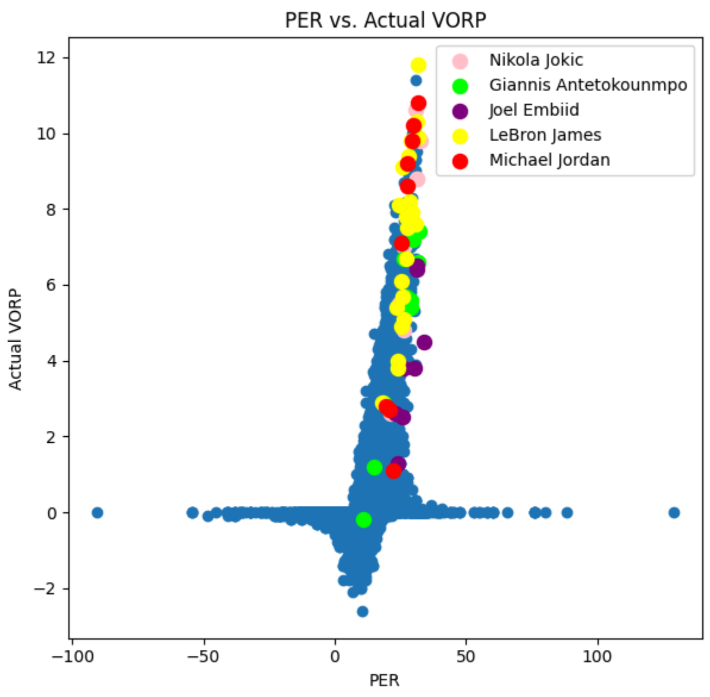

Residual Plot for Salary Model
1e7
Residuals (Salary)
Predicted Salary
1e7

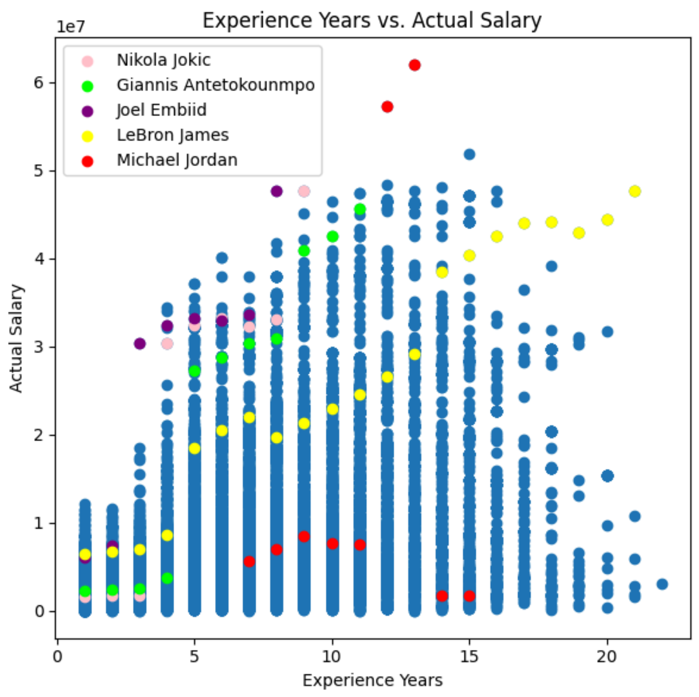






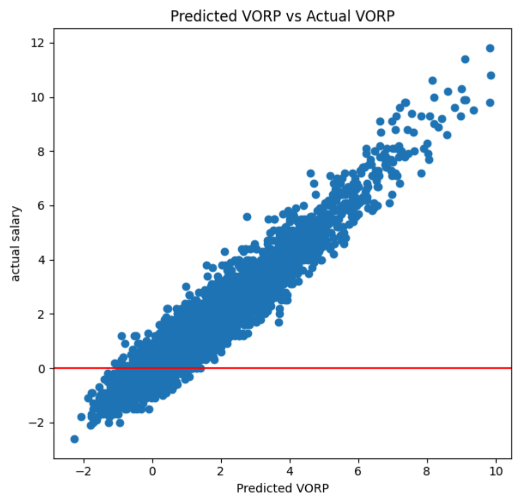
This Chart is super helpful to compare the importance of each estimator for the different models, We can see how PER and experience are only important in one model, but how points per game and minutes per game are important in both

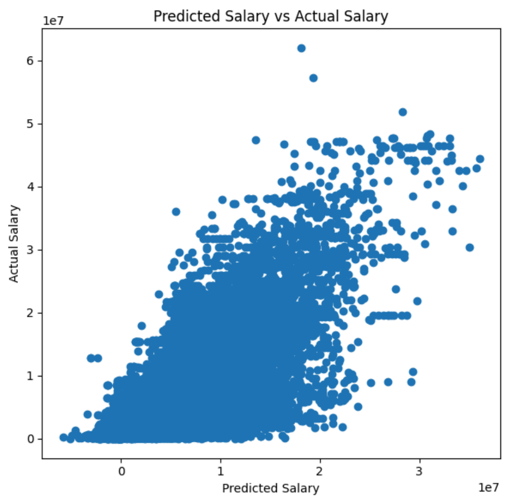




Model iteration

My first iteration of my models used linear regression, which worked great for VORP, but was very lacking for Salary, as seen below. I experimented with other models and found random forest regression to work much better for salary and work at a similar level for VORP. You can see how good VORP was but how bad salary was





Challenges and Conclusion

Challenges

The largest challenge in this project was pivoting from the linear regression models to the random forest models, while my results turned out much better, it took a long time to update everything and redo all my models. I also failed to adjust the salaries initially, which gave me super weird results.

Success Reflection

My models were rather successful, I was able to accurately model both VORP and salary and was able to find differences between the models. Teams should look more into stats like PER when determining a player's contract to get better value, and they should value the minutes a player plays per game over the points they score. The models tell me that a player's availability (games played and minutes per game) are more important to a player's value than stats like points, assists and rebounds. A player who plays every game every year, and who can play lots of minutes (Lebron or Jokic) is more valuable overall than a player who puts up crazy statistics but is often hurt (Embiid). I Think these models and plots could really help players and GM's figure out how to distribute the salary.

Lessons Learned

The biggest lesson I learned was to try different models first before taking the time to improve them. I spent a lot of time trying to perfect linear regression rather than taking the time to explore multiple models and finding what works best. I also learned how to do cool plotting stuff like highlight player names on plots.

Future Work

I wouldn't change much about what I did, but if I had more time, I would dive into the different statistics and try to remove redundant statistics that could mess up my model. For example, 3pt percentage and overall field goal percentage are always related, and only keeping one of the two could make a better model. If I had time to continue on this project, I would look deeper at specific players and find which players had large residuals, showing that they under/over performed. This would allow more specific understanding of players and could help find players that are super under or overvalued in the NBA today.

Acknowledgment and References

My MATH 335 Profesor Ashlyn Munson helped me understand the metric needed to improve my models and helped me understand what was wrong with my old linear regression models.

Glossary. Basketball Reference. (n.d.). https://www.basketball-reference.com/about/glossary.html