Predicting NBA Players' Salaries from Their Statistics

Jordan Rivera

Abstract

The goal of this project was to examine the relationship between National Basketball Association (NBA) players' salaries and their statistics. We aimed to achieve this objective through the construction of a multiple linear regression model that best predicts salaries from player statistics. Using strategies and techniques learned from UCLA's Statistics 101A course, I underwent the process of building, testing, and adjusting a multiple linear regression model. My final model, which was the 42nd iteration, achieved an R^2 value of 0.77793 on Kaggle, scoring first place. However, I decided to select the 39th iteration of my model to be graded and discussed in this paper, as this version had fewer betas and achieved a higher complexity grade while still having an R^2 value higher than the 2nd place model in the class. This model had 16 predictors along with an R^2 value of 0.76852 on Kaggle, which still qualified for 1st place on the leaderboard.

Introduction

The National Basketball Association (NBA) is a professional basketball league in North America that is comprised of 30 teams split into two conferences, the Eastern and Western Conference. Each conference is split into three divisions, each with five teams. The NBA features some of the most talented and highest paid athletes in the world. The purpose of this project was to examine the relationship between a player's salary and his statistics. To do this, we were given a dataset that included 68 columns and 420 observations. One of the column variables was the response variable Salary, leaving 67 variables as potential predictors.

These variables included simple traditional statistics such as games played (G), points per game (PTS), assists per game (AST), and rebounds per game (TRB). There were also advanced statistics that were included, such as Value Over Replacement Player (VORP), Wins Shares (WS), and Player Efficiency Rating (PER). There were several variables that included a period at the end of the variable name - these periods represented the percentage symbol %, and thus allowed for shooting percentage statistics such as Field Goal Percentage (FG.), Three Point Field Goal Percentage (X3P.), and Free Throw Percentage (FT.). Other percentage statistics are slightly more complicated - Assist Percentage (AST.) measures the percentage of teammate field goals a player assisted while he was on the floor, Steal Percentage (STL.) measures the percentage of opponents' possessions that resulted in a steal by the player, Block Percentage (BLK.) measures the percentage of opponents' field goal attempts that were blocked by the player, and so on. The dataset also included information such as the player's age, nationality, and position, along with team information and team statistics such as team wins (T.W) and losses (T.L), along with more advanced metrics such as team offensive rating (T.Ortg) and team defensive rating (T.DRtg). Here is a comprehensive list of all initial predictors provided along with their variable type.

Table 1: Type of Predictors Provided in Training Dataset

Variable	Type
NBA_Country	Categorical
Age	Numerical
TM	Categorical
G	Numerical

Variable	Type
MP	Numerical
PER	Numerical
TS.	Numerical
X3PAr	Numerical
FTr	Numerical
ORB.	Numerical
DRB.	Numerical
TRB.	Numerical
AST.	Numerical
STL.	Numerical
BLK.	Numerical
TOV.	Numerical
USG.	Numerical
OWS	Numerical
DWS	Numerical
WS	Numerical
WS.48	Numerical
OBPM	Numerical
DBPM	Numerical
BPM	Numerical
VORP	Numerical
Rk	Numerical
Pos	Categorical
GS	Numerical
FG	Numerical
FGA	Numerical
FG.	Numerical
X3P	Numerical
X3PA	Numerical
X3P.	Numerical
X2P	Numerical
X2PA	Numerical
X2P.	Numerical
FT	Numerical
FTA	Numerical
FT.	Numerical
ORB	Numerical
DRB	Numerical
TRB	Numerical
AST	Numerical
STL	Numerical
BLK	Numerical
TOV	Numerical
PF	Numerical
PTS	Numerical
Ortg	Numerical
DRtg	Numerical
Team.Rk	Numerical
Team	Categorical
T.Conf	Categorical
T.Div	Categorical
T.W	Numerical

Variable	Type
T.L	Numerical
T.W.L.PERC	Numerical
T.MOV	Numerical
T.Ortg	Numerical
T.DRtg	Numerical
NRtg	Numerical
MOV.A	Numerical
Ortg.A	Numerical
DRtg.A	Numerical
NRtg.A	Numerical

Methodology

In total, there were 42 iterations of my model, with the 42nd and final version achieving the highest R^2 score on Kaggle, and the 39th version being the model to be graded and the topic of this paper. For the purpose of brevity, I will not go through all 42 attempts and each of the minor adjustments and modifications made in between each, but rather highlight some of the most important versions that were key to leading to the final product. I will also explain the strategies and techniques used along the way.

New Variables

One of the most important strategies in building my model was the creation of new variables from existing variables. While many of these new variables were not useful and therefore discarded, there were others that proved to be quite useful, even if they weren't used in the final model. I will now highlight the most important new variables that I created.

New Numerical Predictors

The first two predictors I created were numeric predictors: Percentage of Games Started and Minutes Per Game. As a fan of the NBA, I leveraged my knowledge of the sport to think of these variables. I knew that starting players are likely to be paid more than bench players, and that teams will want their highest paid players (presumably their best players) to play more minutes per game. Therefore, the two numerical predictors I created are Percentage of Games Started (GS.) and Minutes Per Game (MPG), which were calculated by simply dividing GS by G, and MP by G, respectively. Below we see that both of these variables are positively related with Salary, although there is a lot of error.

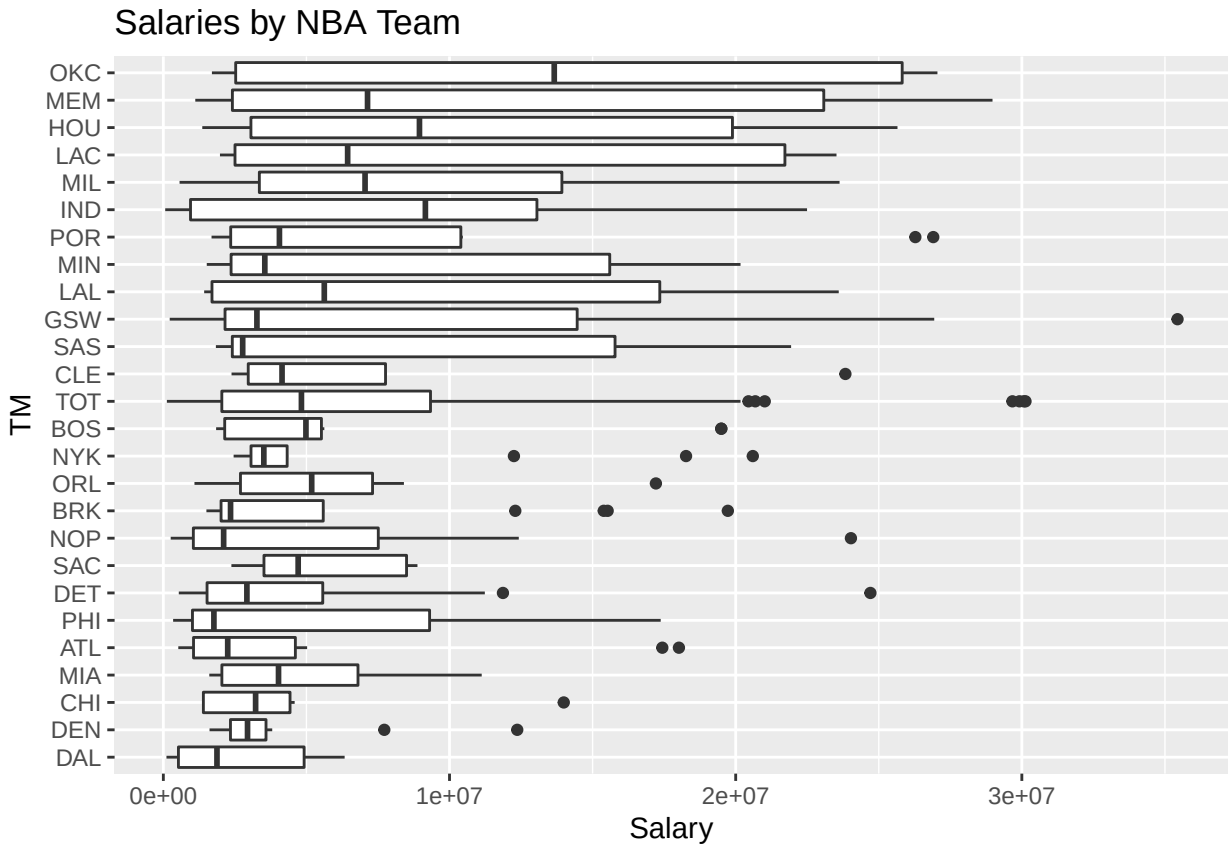
Both of these predictor variables were integral to my model, as they were added in within the first two versions of the model and were present in my final model.

New Categorical Predictors

Team Payroll I also created several categorical predictors. The first was TeamPayroll, which categorized teams into three different levels: High, Medium, and Low. In order to sort the teams, I looked at many different factors. The first was the mean salary for each team, according to the training dataset.

TM	mean_salary	number
OKC	14079327	10
MEM	12092245	14
HOU	11548655	14
LAC	10912244	8
MIL	9675639	14
IND	9333285	8
POR	9072419	10
MIN	8777972	12
LAL	8676504	14
GSW	8371636	22
SAS	8263279	8
CLE	8219170	5
TOT	7019450	119
BOS	6811163	10
NYK	6515755	13
ORL	5803428	12
BRK	5610724	16
NOP	5575611	10

TM	mean_salary	number
SAC	5460469	11
DET	5310799	15
PHI	5134628	19
ATL	5042549	11
MIA	4921230	7
CHI	4192996	8
DEN	3889827	12
DAL	2603234	18



I also looked up the annual payrolls for the teams during the 2015-2016, 2016-2017, and 2017-2018 seasons to help inform my decisions. These numbers were found on Basketball Reference. Furthermore, a good amount of trial and error was used, as I often changed the brackets and then used cross validation to see how it would affect the R^2. In the end, after redefining the levels many times, the teams were categorized into High, Medium, and Low as such:

Table 3: Team Payroll Levels: High, Medium, and Low

High	Medium	Low
OKC	HOU	IND
MEM	LAC	POR
SAS	MIL	MIN
BOS	GSW	LAL
	CLE	NYK
	DAL	ORL

High	Medium	Low
	TOT	BRK
		NOP
		SAC
		DET
		PHI
		ATL
		MIA
		CHI
		DEN

Position Another categorical predictor that I created was simplifying the Pos variable, which gave the player's position and had seven levels. The original variable was distributed as such:

Table 4: Distribution of the Pos variable

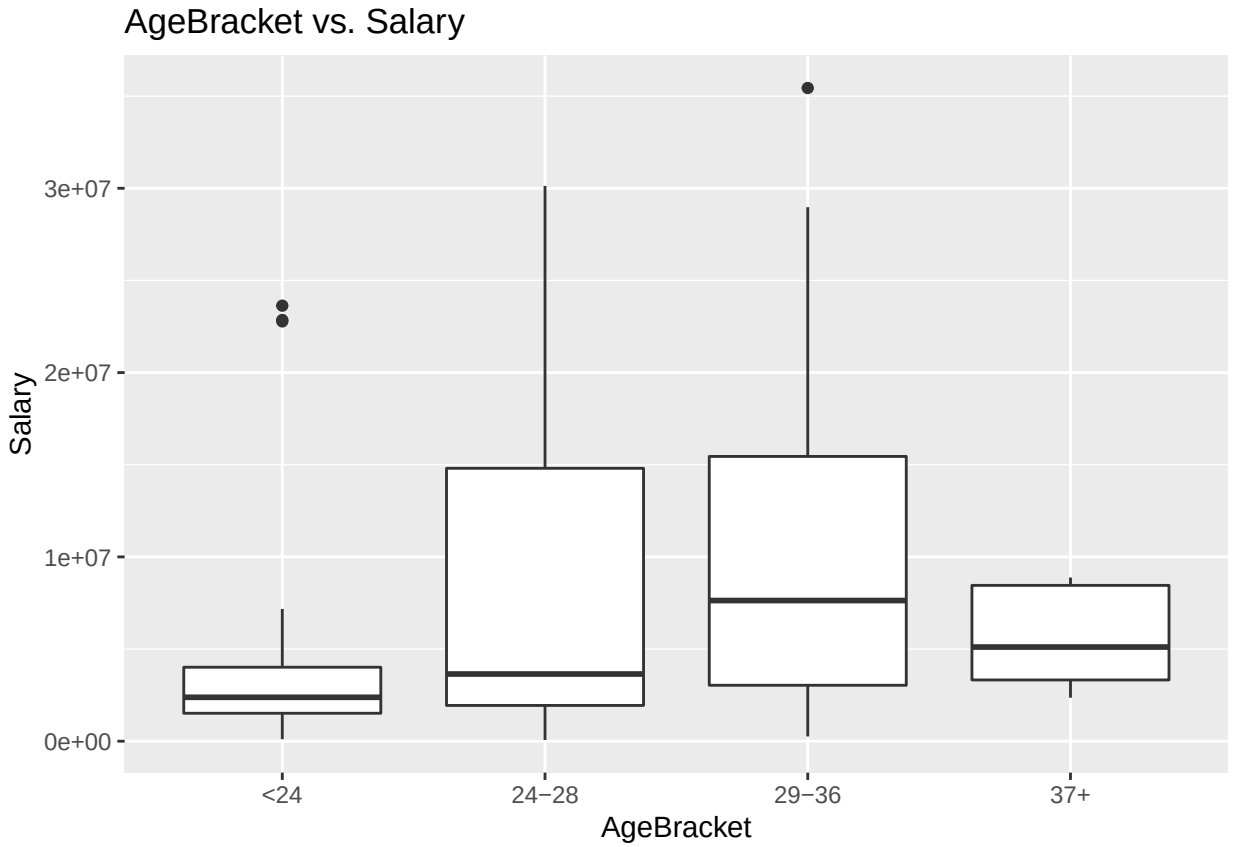
PG	PG-SG	SG	SF-SG	SF	PF	C
100	1	92	4	71	77	75

As you can see, there are some positions, such as PG-SG and SF-SG, that have very few observations. Furthermore, there was incentive to limit the number of levels of this variable so as to reduce the number of betas in the final model. I simplified the Pos variable into a new Position variable, a categorical predictor with three levels. The level "Guard" included PG, SG, and PG-SG. The level "Forward" included SF and SF-SG. The level "Big" included PF and C. It is distributed as such:

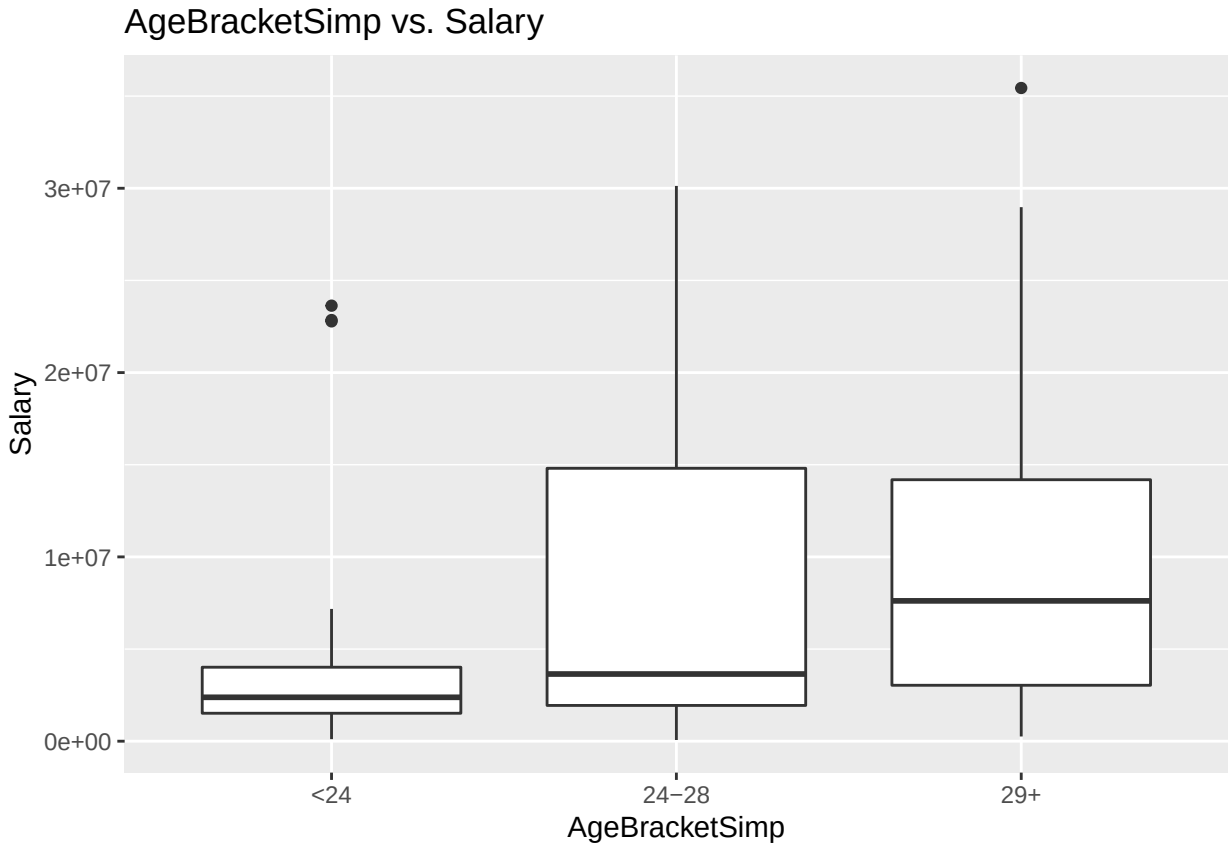
Table 5: Distribution of the Position variable

Guard	Forward	Big
193	75	152

Age Bracket Another categorical predictor that I created transformed the Age variable from a numerical predictor to a categorical predictor, a variable I called AgeBracket. This variable had four levels: <24, 24-28, 29-36, and 37+. Like the TeamPayroll variable, I made a lot of minor adjustments to these levels throughout the many versions of my model, slightly changing the age boundaries to maximize the fit.



Simplified Age Bracket I also created a simplified version of AgeBracket, which I called AgeBracketSimp. The difference is simple: I combined 29-36 and 37+ into one level, 29+. The reason for doing this was that both AgeBracket and AgeBracketSimp contribute to the model when they interact with different variables. For example, the AgeBracket:PTS interaction maximized fit, while the AgeBracketSimp:TeamPayroll also maximized fit. Using only one variable or the other led to increased error and a lower R^2. A lot of trial and error with cross validation was used to determine this. In fact, I created a third Age Bracket variable called AgeBracketAlt that I used in Models 41 and 42, as it proved to be beneficial when interacting with certain variables. However, since this write-up is on Model 39, that is not relevant here. I will discuss the interactions later on in this paper. Also, creating this variable helped to simplify my model so as to reduce the number of betas.

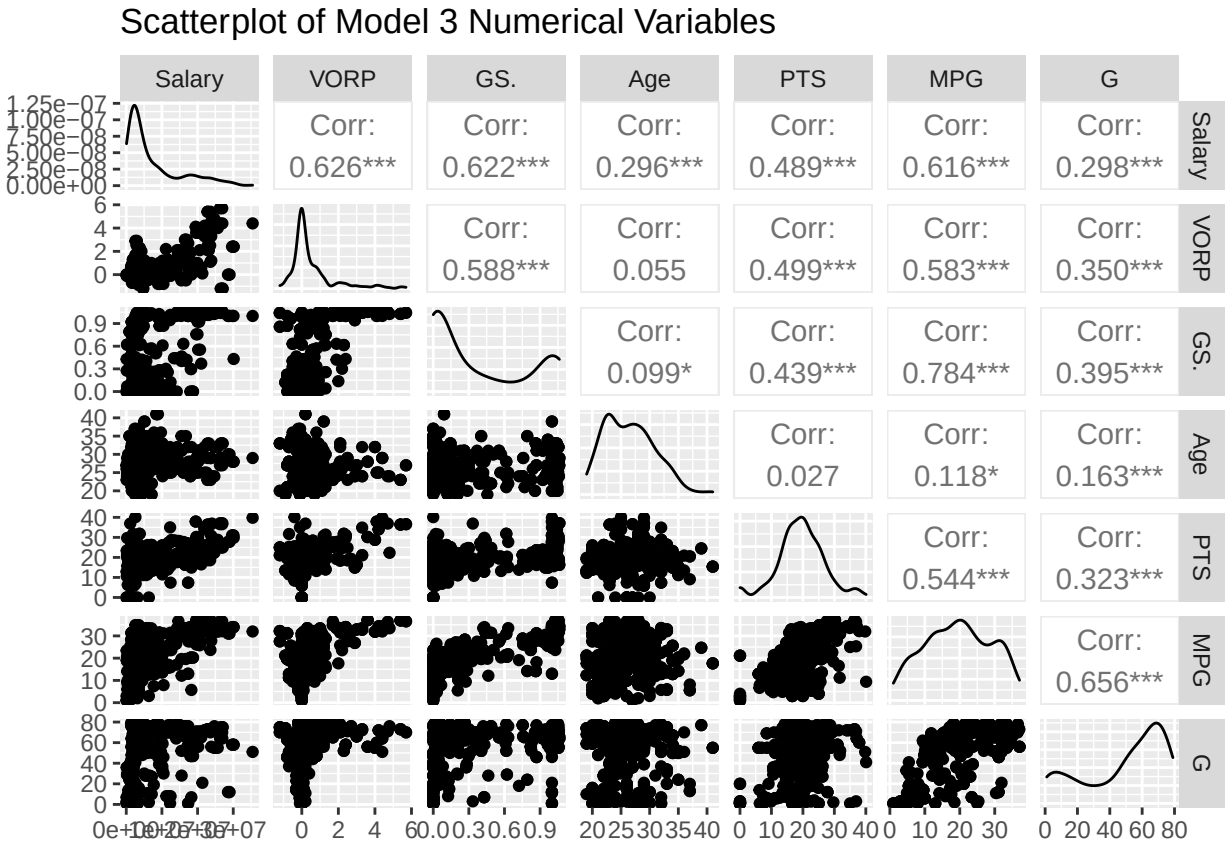


Building the Model

As mentioned earlier, there were 42 different iterations of my model. In order to be concise, yet also sufficiently capture the strategies and thought process used in developing the model, I will highlight the most important versions of the model in this section. All versions of my model can be found in the accompanying R file.

Model Version 3

Selecting the predictor variables for my first few versions of my model was done using my knowledge of the NBA. For example, I included the aforementioned GS. (percentage of games started) and MPG (minutes per game) variables that I had created. I also included VORP (Value Over Replacement Player), a statistic that I knew was a good measure for the overall value of a player. Here is a scatterplot matrix of the numerical predictors used in this model.

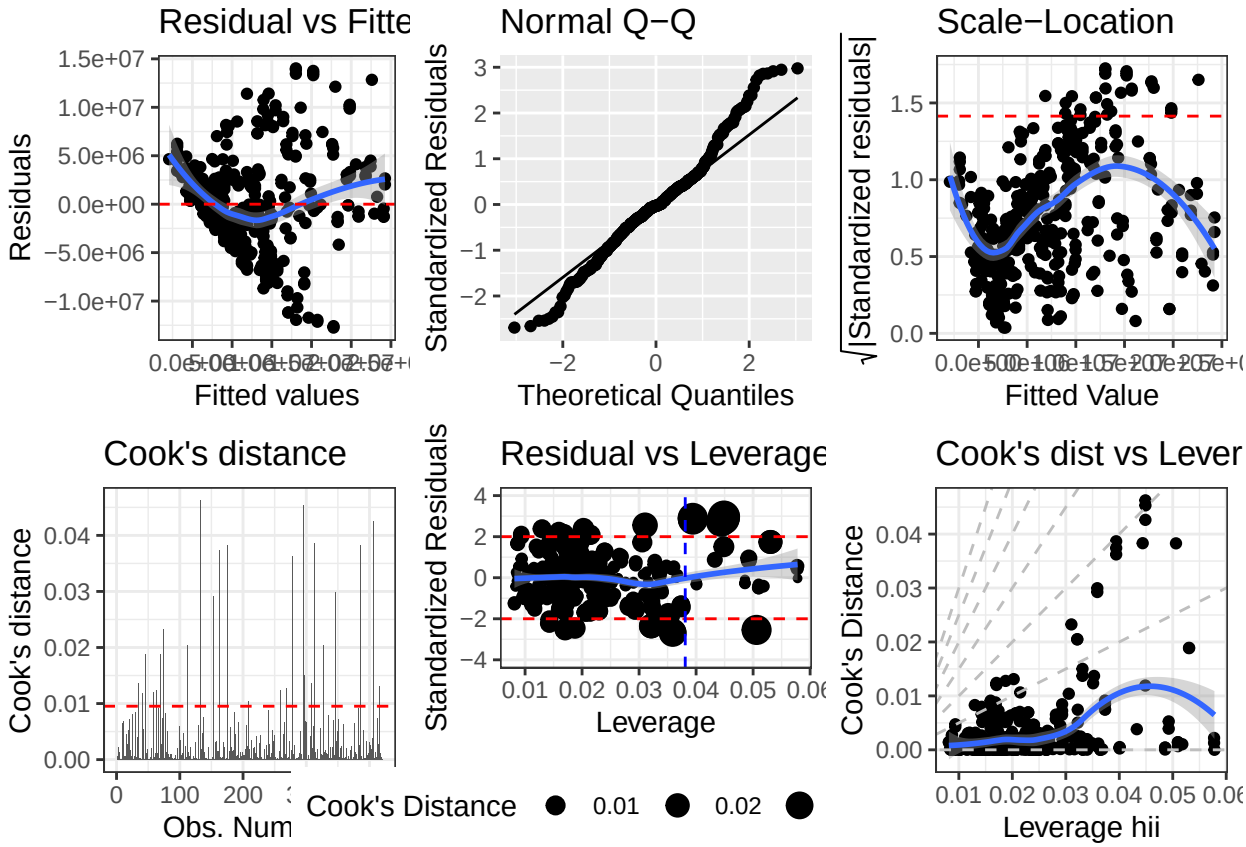


In addition to these numerical predictors, I also included one categorical predictor: Position.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-10004109.02	1634311.15	-6.121300	0.0000000
VORP	1919065.48	262855.96	7.300826	0.0000000
GS.	2935973.11	1004189.37	2.923725	0.0036500
Age	435620.35	54129.54	8.047738	0.0000000
PTS	111504.87	41397.11	2.693543	0.0073591
PositionForward	-2262505.43	707609.94	-3.197391	0.0014940
PositionGuard	-2280832.34	552564.94	-4.127718	0.0000444
MPG	292179.00	56848.89	5.139572	0.0000004
G	-57051.13	13309.57	-4.286475	0.0000226

Table 7: Model Version 3

Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
4820000	0.6049017	0.5972112	0.49226



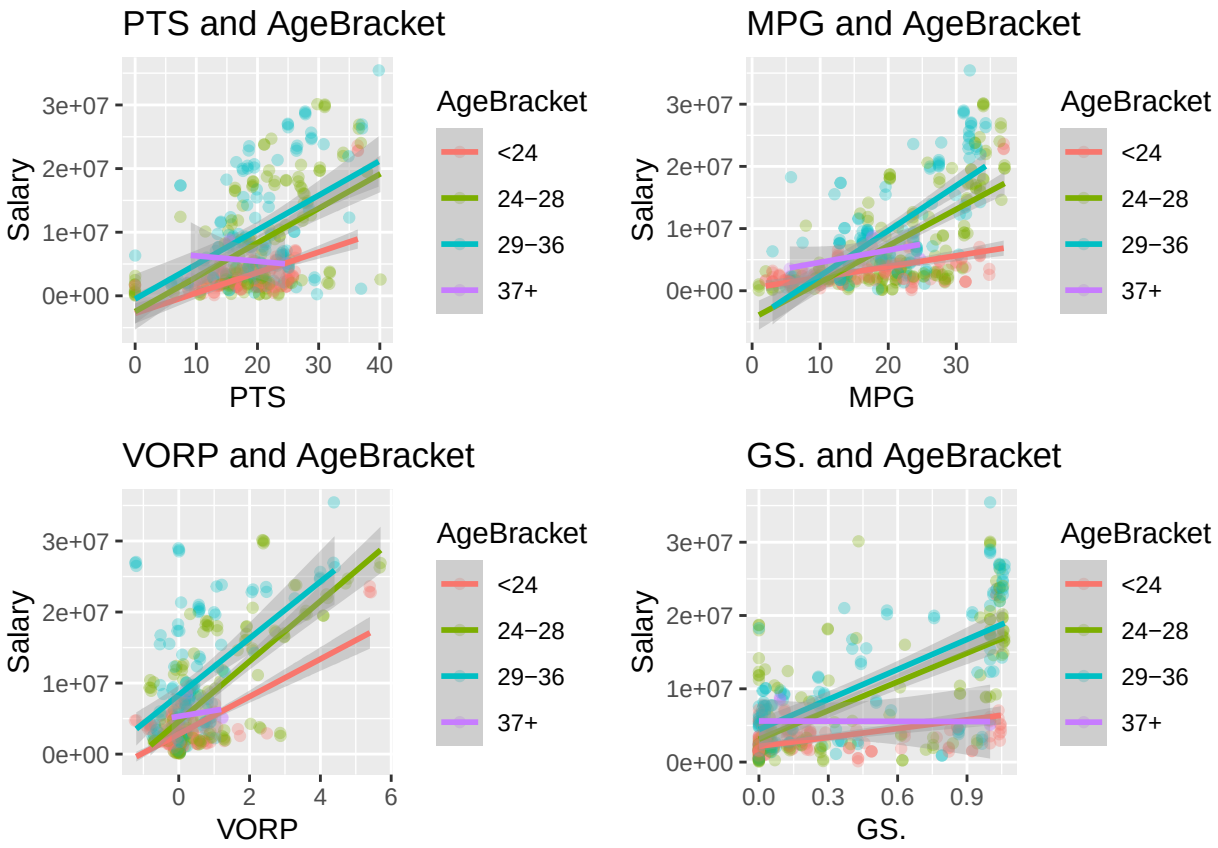
These diagnostic plots reveal several issues. In the first graph, we see a non-random pattern, a fan shape, suggesting that there is a trend that is not being explained by the model. The Scale-Location plot reveals that there is non-constant variance. There also appear to be a good amount of bad leverage points. I attempted to address these issues in future models.

Model Version 12

Between Model Version 3 and Model Version 12, I attempted several strategies such as performing a y-variable log transformation and predictor variable transformations using the Box-Cox procedure, and finding all predictors with a correlation with Salary that was greater than 0.2. I also used the backwards AIC step function to try and eliminate unneccesary variables. All attempts proved to be fruitless in improving my model, until I reached Model Version 10, where I started testing out interactions.

The idea behind interactions was that I wanted a way to reflect the fact that experienced players are paid more than inexperienced players. While Age can reflect this, it does not completely capture this idea, as players can break into the NBA at different ages. For example, I knew that in order to differentiate between a 30-year-old veteran player who scores 30 points per game (likely gets paid a high salary) and a 20-year-old rookie who scores the same amount (likely gets paid a lower salary), I would have to implement an interaction.

Model Version 12 involves the following four interactions. As a note, only one of these interactions was included in the final model, Model Version 39, but I am displaying all four used here to illustrate the process. In general, very young players and very old players might have lower salaries compared to average-aged players with comparable statistics.

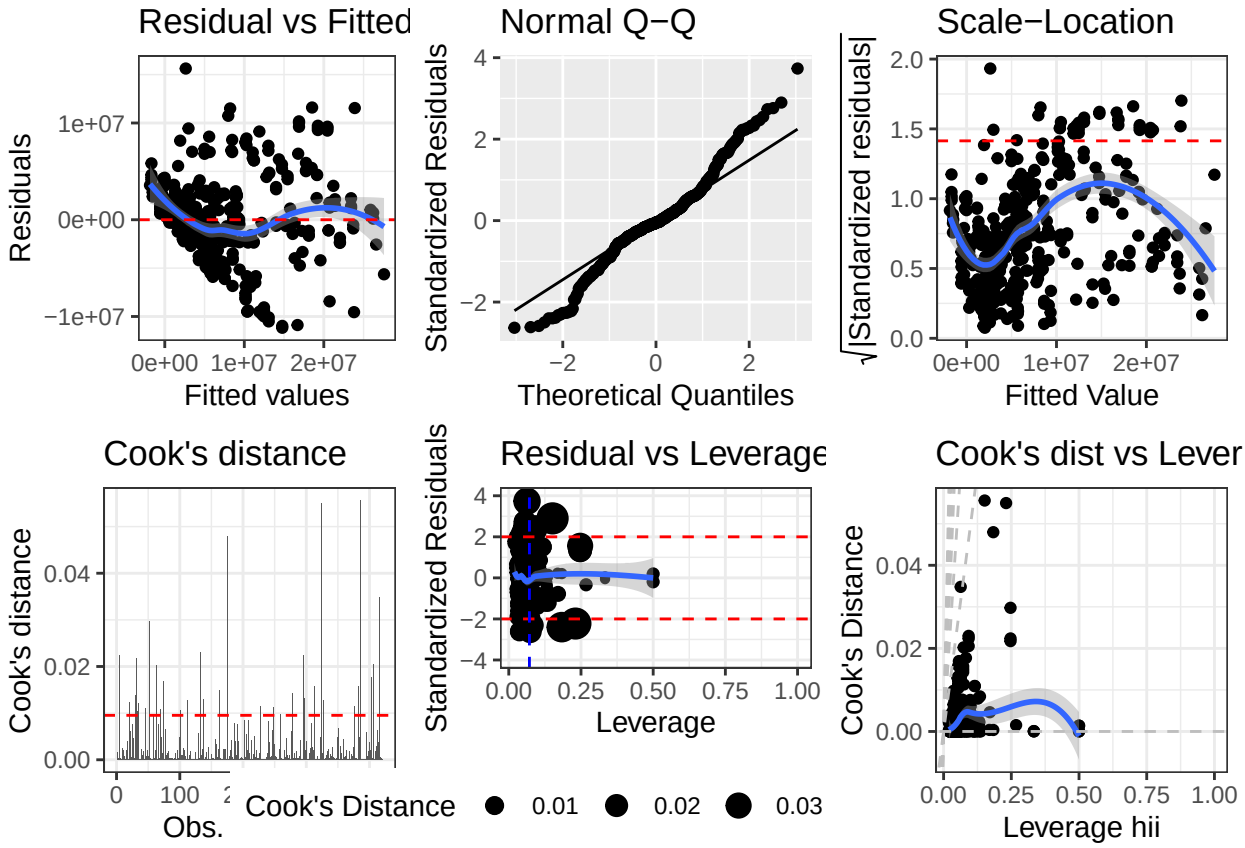


	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-6568328.288	10718647.36	-0.6127945	0.5403667
VORP	1478808.467	505084.02	2.9278465	0.0036118
GS.	-2941605.207	2094489.80	-1.4044495	0.1609749
AgeT	2610749.404	3237249.93	0.8064714	0.4204589
PTS	467697.151	187193.51	2.4984688	0.0128807
MPG	298273.242	93417.27	3.1929132	0.0015219
G	-39805.737	13372.57	-2.9766712	0.0030943
TRB.	-159794.769	75619.46	-2.1131434	0.0352184
OBPM	-79711.714	244786.29	-0.3256380	0.7448716
WS.48	3280421.449	7587972.16	0.4323186	0.6657470
X3P	-807239.090	309372.24	-2.6092809	0.0094197
PER	-125280.705	178846.78	-0.7004918	0.4840347
FG	-571497.363	377509.11	-1.5138638	0.1308642
PositionForward	-2900334.823	754503.35	-3.8440318	0.0001411
PositionGuard	-3334079.726	720113.25	-4.6299380	0.0000050
PTS:AgeBracket24-28	27423.272	83493.29	0.3284488	0.7427473
PTS:AgeBracket29-36	-79382.934	91288.66	-0.8695815	0.3850599
PTS:AgeBracket37+	-204858.200	274560.38	-0.7461317	0.4560339
MPG:AgeBracket24-28	8972.203	101721.41	0.0882037	0.9297597
MPG:AgeBracket29-36	197259.304	107060.55	1.8425023	0.0661548
MPG:AgeBracket37+	519294.252	366635.36	1.4163780	0.1574570
VORP:AgeBracket24-28	675459.438	611339.59	1.1048842	0.2698858
VORP:AgeBracket29-36	389926.548	668916.43	0.5829227	0.5602797
VORP:AgeBracket37+	6956320.932	19792808.07	0.3514570	0.7254339
GS.:AgeBracket24-28	5288223.075	2525056.14	2.0942992	0.0368733
GS.:AgeBracket29-36	7591447.654	2446477.72	3.1030112	0.0020541

	Estimate	Std. Error	t value	Pr(>|t|)
GS.:AgeBracket37+	-16766489.869	23152791.76	-0.7241671	0.4693940

Table 9: Model Version 12

Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
4324000	0.6959732	0.6758595	0.54393



There were still several issues at this point in the model development. The diagnostic plots showed the same problems involving non-random residuals and non-constant variance. Furthermore, I was getting rather frustrated that my Kaggle R^2 was always significantly lower than my training data R^2.

New Strategies

While creating the next iterations of my model, I attempted to solve many of the issues I was having. To solve the problem of the Kaggle R^2 being much lower than my training data R^2, I decided to implement cross validation by splitting the training data into my own training and testing data. However, I wanted the relationship between my training and testing data to be as close as possible to the relationship between the actual training and testing, so as to output an R^2 that was as accurate as possible.

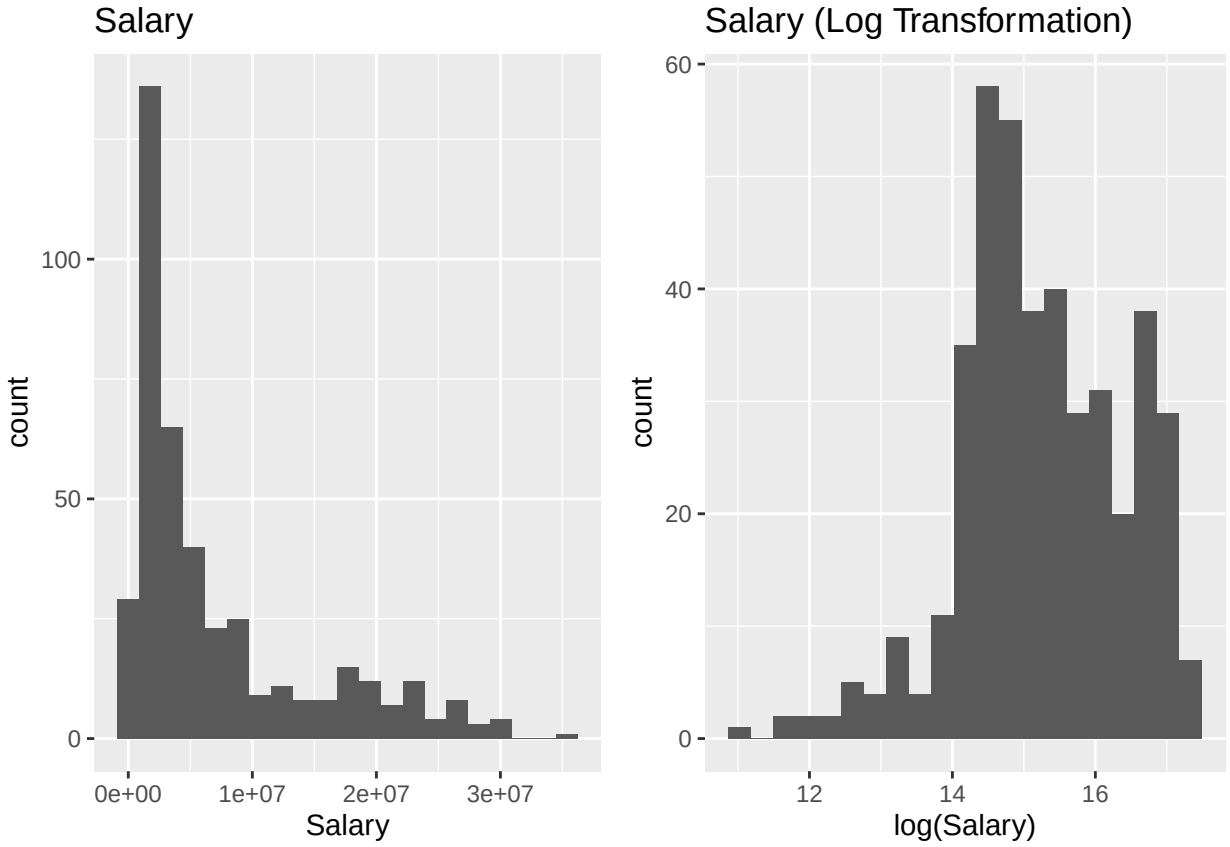
To do this, I wrote a function in R called optimal_seed(), which was very simple - it set the seed to every number between 1 and 1000 and found the seed that resulted in an R^2 closest to the one given by Kaggle. For example, if Model Version 15 had an R^2 of 0.55, I would use this function to find the seed that would split the training data in such a way that the cross validation would have an R^2 as close to 0.55 as possible. I would then use this seed to test the next model, Model Version 16 before submitting on Kaggle. This

function, although simple, was instrumental to my success, as it allowed me to test many different models in R before submitting to Kaggle, basically ensuring that any submission would have a positive result. It also proved to predict the Kaggle R^2 quite well. No extra packages or functions were used in creating this function - the code can be found in the accompanying R file.

This function allowed me to create another simple function that I called comp_search(), which searched through all the predictor variables and added the one that increased the R^2 the most as given by the cross validation. I was inspired by the step() function to write this function, as it continues to add variables to the model until the R^2 given by the cross validation doesn't increase anymore. I also modified it from time to time to see what variables interacted best with other variables to increase R^2 the most. Like the optimal_seed() function, I used only basic R commands and no other packages to write this simple function, and the code can be found in the R file.

Model Version 23

Between Model Version 12 and Model Version 23, not only did I implement these new strategies, but I also tried simplifying models using the step function, added new created variables such as TeamPayroll, and adjusted the AgeBracket levels. Adjusting the AgeBracket levels slightly raised the Kaggle R^2 by 4%, the largest leap so far. However, the model still had violations of assumptions, so for Model Versions 22 and 23, I went back to the log transformation of the y variable, as suggested by the Box-Cox procedure. As we can see on the graphs below, the transformation helped improve normality.

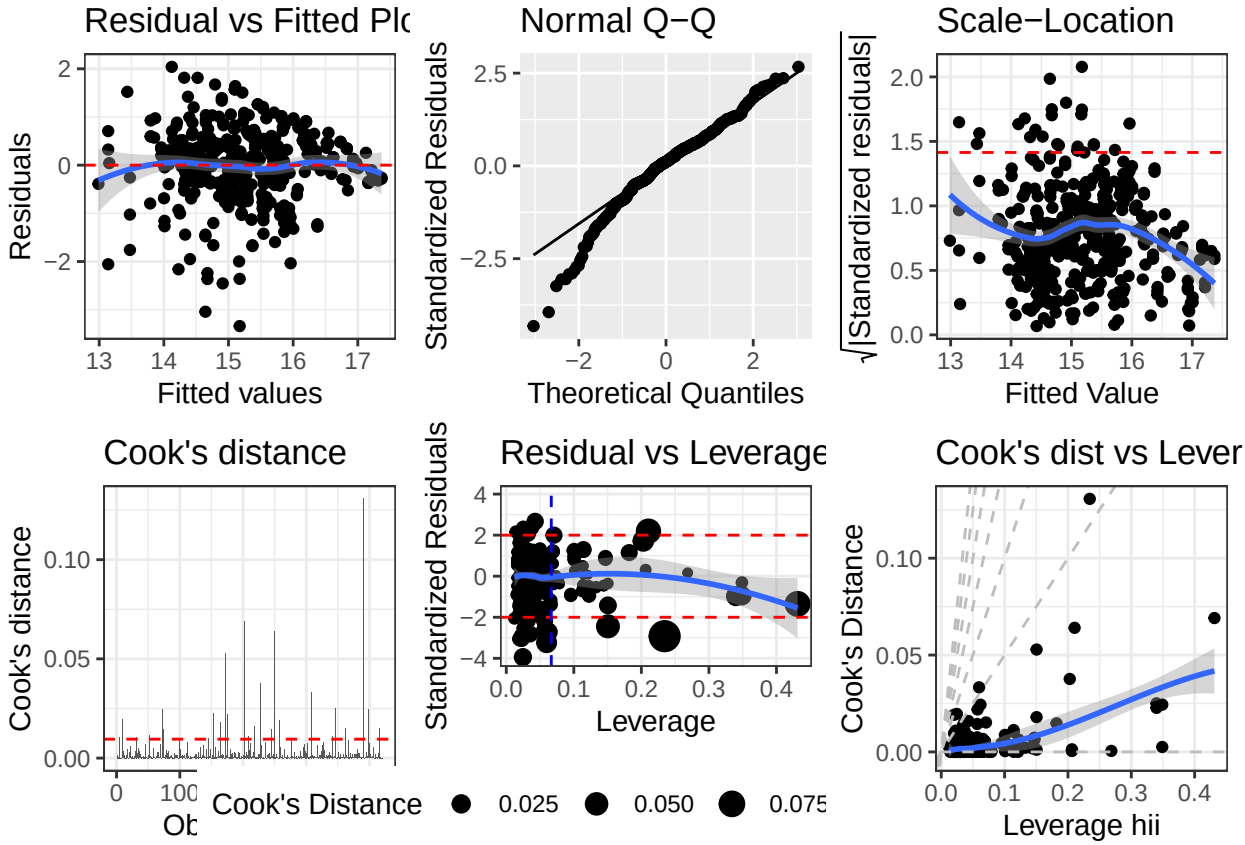


	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	13.3610209	0.4845452	27.5743516	0.0000000
VORP	0.1013369	0.0786556	1.2883611	0.1983650
AgeBracket24-28	-0.8598952	0.2251813	-3.8186800	0.0001555
AgeBracket29-36	0.2110296	0.2515593	0.8388863	0.4020342
AgeBracket37+	-0.1589837	0.6893460	-0.2306298	0.8177203

	Estimate	Std. Error	t value	Pr(>|t|)
MPG	0.0265300	0.0084402	3.1433033	0.0017950
FTA	0.1148627	0.0272148	4.2206005	0.0000302
Rk	0.0008489	0.0003542	2.3967555	0.0169991
TeamPayrollLow	0.0343190	0.1442643	0.2378899	0.8120883
TeamPayrollMedium	-0.0697274	0.1403601	-0.4967754	0.6196204
WS.48	-0.8260808	0.5022532	-1.6447497	0.1008070
X2P.	1.4190814	0.4768727	2.9758074	0.0030996
FT.	-0.5062060	0.3196831	-1.5834620	0.1141065
STL.	0.0403334	0.0350202	1.1517199	0.2501241
FTr	-0.4603800	0.3676058	-1.2523743	0.2111652
TS.	0.0382143	0.6977731	0.0547661	0.9563521
OWS	-0.0313971	0.0516160	-0.6082821	0.5433456
AgeBracket24-28:MPG	0.0606179	0.0105983	5.7195918	0.0000000
AgeBracket29-36:MPG	0.0411301	0.0119728	3.4352852	0.0006539
AgeBracket37+:MPG	0.0582247	0.0405028	1.4375476	0.1513443

Table 11: Model Version 23

Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
0.7815	0.5541773	0.5330007	0.56893

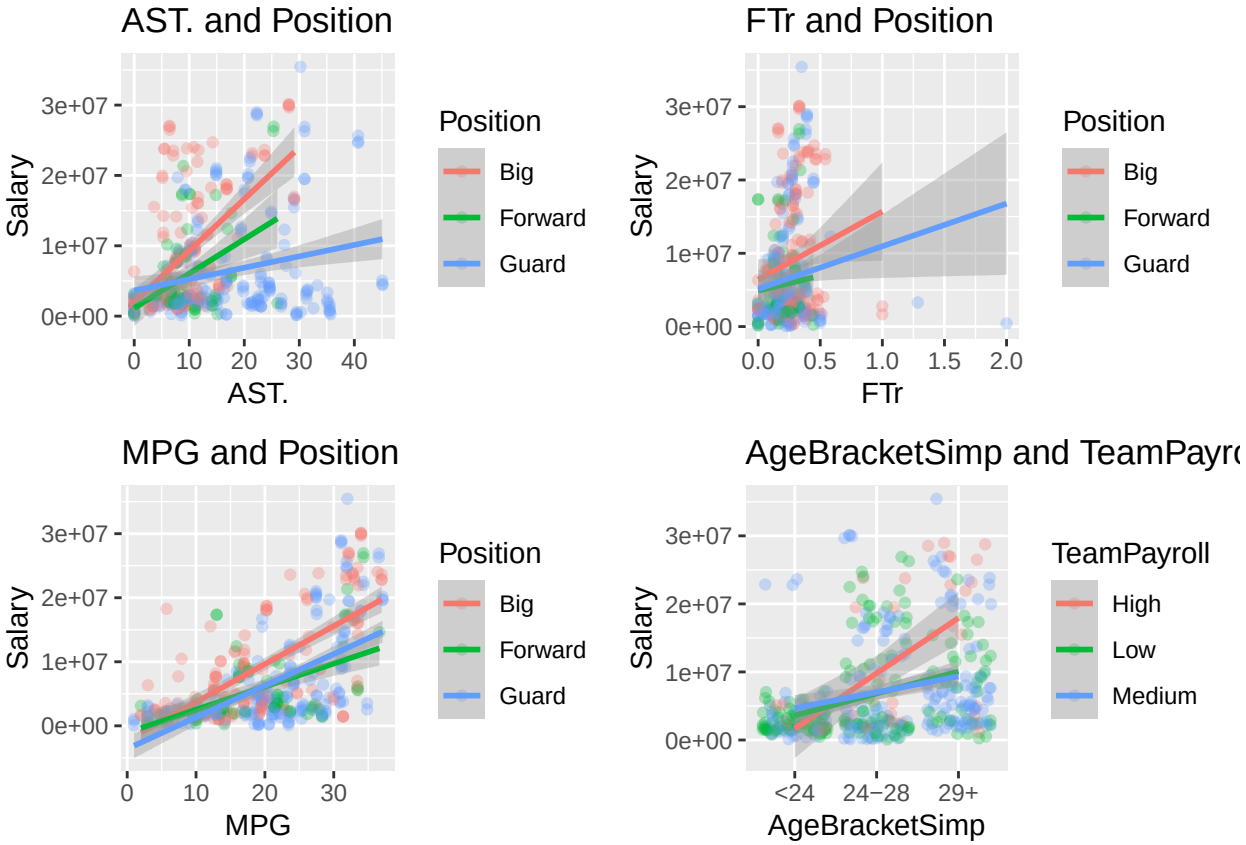


Although performing the transformation seemed to improve normality and the randomness of the residuals, there still appeared to be a fan shaped pattern and non-constant variance. Considering I had been able to

achieve an R^2 10 points higher in Model Version 21 compared to this model, and that the log transformation did not drastically improve the validity of the model, I opted to go back to the untransformed y-variable for subsequent versions of the model.

Model Version 27

Between Model Version 23 and Model Version 27, I explored many new interactions, which helped boost my R^2 value another five points. From a logical standpoint, these interactions make sense. Guards are expected to record a lot of assists, so guards with high AST. values are more common than centers with high AST. values. Therefore, centers with high AST. values would likely command a higher salary. Similar comparisons can be made for FTr (Free Throw Rate) and MPG (Minutes Per Game). An interaction between two categorical variables - AgeBracketSimp and TeamPayroll, also proved to be very beneficial, as it raised the Kaggle R^2 by another three points.

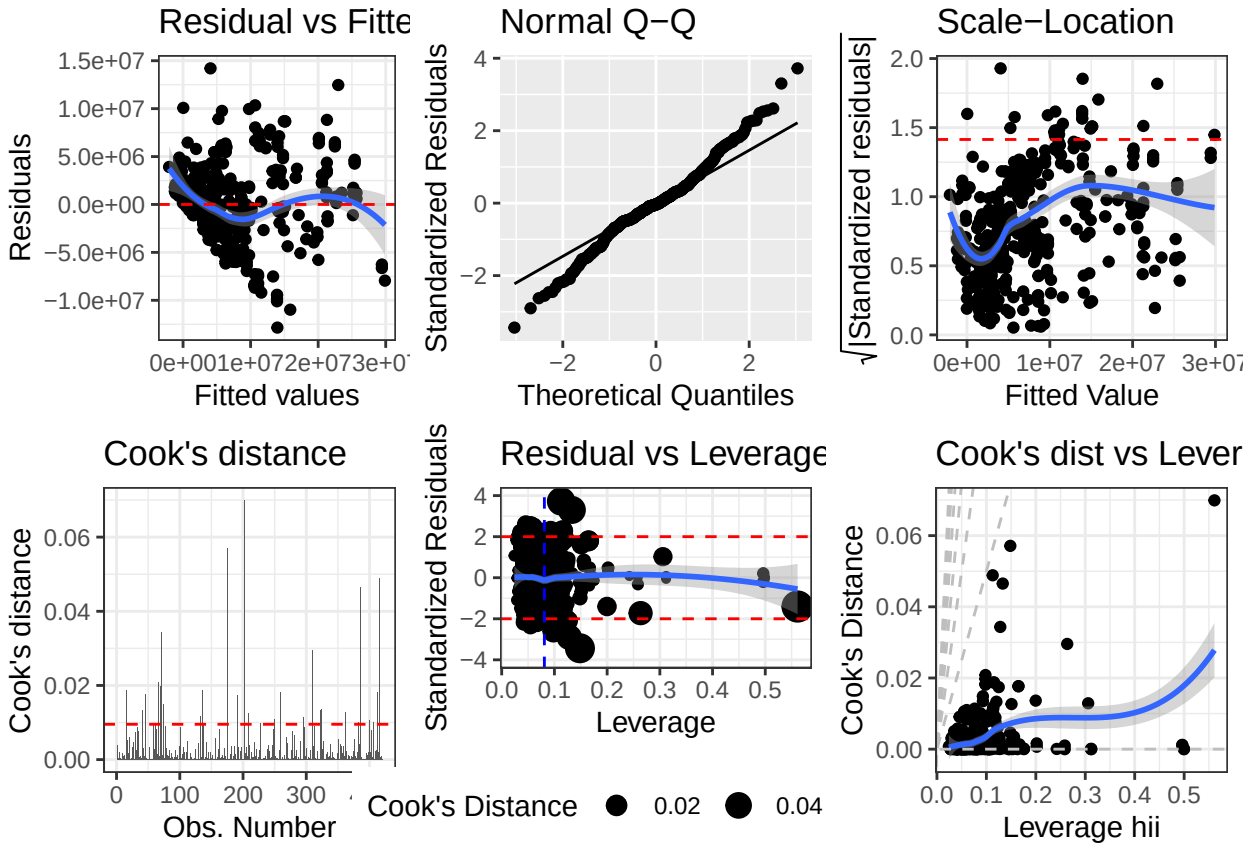


	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2868663.442	2257837.607	1.2705358	0.2046631
FTA	886672.823	227578.219	3.8961234	0.0001153
VORP	1691253.567	275333.528	6.1425631	0.0000000
TeamPayrollLow	220717.889	1406588.472	0.1569172	0.8753926
TeamPayrollMedium	1113904.887	1499804.925	0.7426998	0.4581175
BLK	-1184908.870	317709.370	-3.7295371	0.0002207
Rk	3578.650	2007.165	1.7829376	0.0753863
USG.	71394.403	49219.494	1.4505310	0.1477268
FT.	-2128077.989	1817355.118	-1.1709753	0.2423348
AST	-266542.585	157927.975	-1.6877478	0.0922716
PTS:AgeBracket<24	-109319.978	87290.952	-1.2523632	0.2111997
PTS:AgeBracket24-28	-17453.134	69025.701	-0.2528498	0.8005196

	Estimate	Std. Error	t value	Pr(>|t|)
PTS:AgeBracket29-36	-157877.248	79903.887	-1.9758394	0.0488884
PTS:AgeBracket37+	-942790.331	240544.391	-3.9194027	0.0001051
AgeBracket<24:GS.	810960.539	2132878.267	0.3802189	0.7039932
AgeBracket24-28:GS.	4241555.779	1450775.405	2.9236474	0.0036645
AgeBracket29-36:GS.	4273185.044	1337985.052	3.1937465	0.0015203
AgeBracket37+:GS.	-11697362.365	5167572.884	-2.2636086	0.0241556
AgeBracket<24:MPG	8310.195	88306.457	0.0941063	0.9250738
AgeBracket24-28:MPG	207417.821	78616.628	2.6383454	0.0086700
AgeBracket29-36:MPG	365327.893	82462.405	4.4302357	0.0000123
AgeBracket37+:MPG	1326099.842	276837.772	4.7901695	0.0000024
PositionBig:AST.	239519.273	84366.072	2.8390473	0.0047654
PositionForward:AST.	333276.855	138844.844	2.4003546	0.0168544
PositionGuard:AST.	1420.160	53793.859	0.0264000	0.9789520
PositionBig:FTr	-5095852.786	3130323.716	-1.6278996	0.1043663
PositionForward:FTr	-18649098.892	5934904.967	-3.1422742	0.0018065
PositionGuard:FTr	-3667637.358	2368220.866	-1.5486889	0.1222803
MPG:PositionForward	-34237.261	73938.426	-0.4630510	0.6435901
MPG:PositionGuard	-31065.022	49987.116	-0.6214606	0.5346654
TeamPayrollHigh:AgeBracketSimp24-28	-6160519.216	2245941.848	-2.7429558	0.0063741
TeamPayrollLow:AgeBracketSimp24-28	-3074398.229	1812000.225	-1.6966876	0.0905661
TeamPayrollMedium:AgeBracketSimp24-28	-4086223.296	1959216.525	-2.0856415	0.0376708
TeamPayrollHigh:AgeBracketSimp29+	4568207.267	2691067.483	1.6975447	0.0904039
TeamPayrollLow:AgeBracketSimp29+	-1698982.111	1992902.017	-0.8525166	0.3944588
TeamPayrollMedium:AgeBracketSimp29+	-1772927.441	2048721.939	-0.8653822	0.3873694

Table 13: Model Version 27

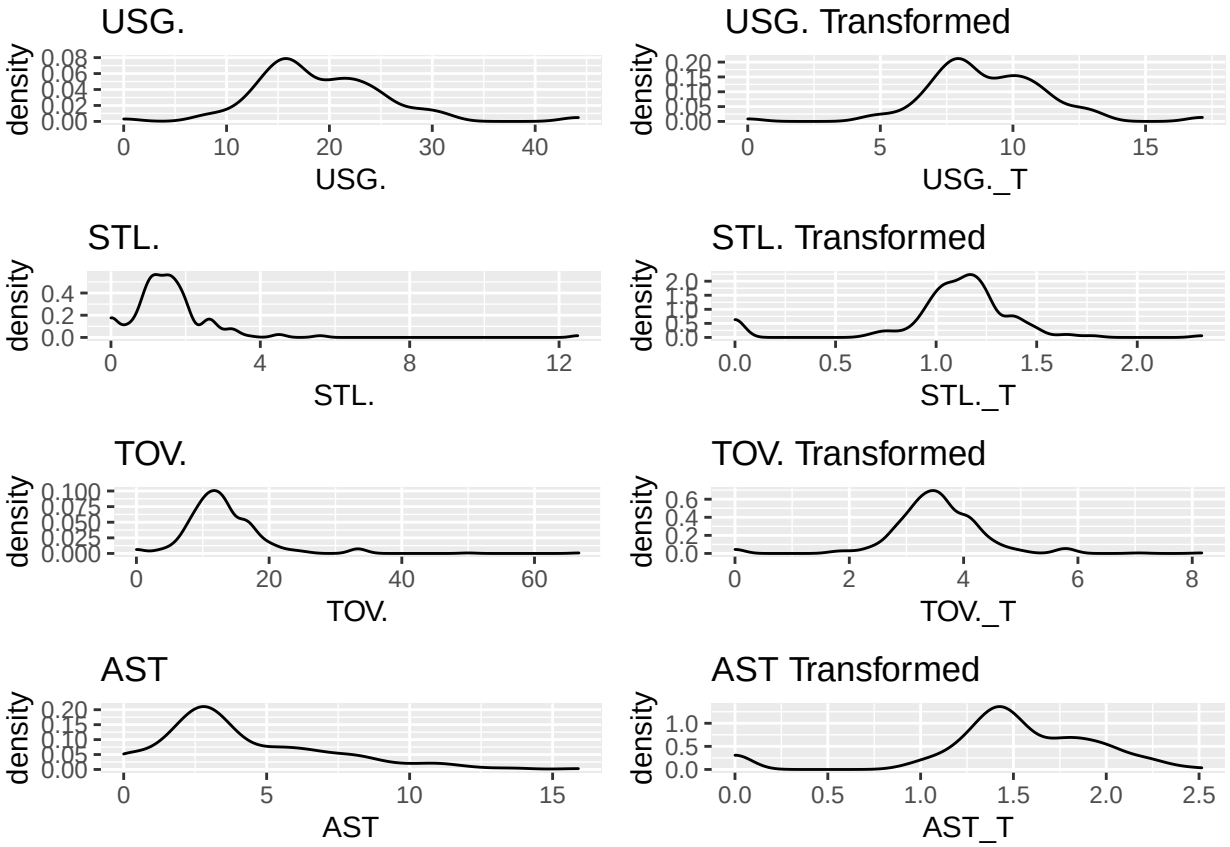
Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
4133000	0.7390569	0.7152731	0.71912



As we can see from the diagnostic plots, this model still violates several assumptions, although it is not as extreme as early models. Future models will attempt to improve the validity of the model.

Model Version 33

The changes between Model Version 27 and Model Version 33 included a repeated pattern of simplifying the model using backwards AIC, then adding new variables and interactions. In order to improve the validity of the model, I decided to perform transformations on my numerical predictors. I used the PowerTransform() function to carry out the Box-Cox procedure and transform the predictors, and if the transformation did not negatively impact the R^2 of the cross validation technique, I kept it. As a result, I performed transformations on USG., STL., TOV., and AST. Looking at the graphs below, the left column displays the densities for the untransformed variables, while the right column displays the densities for the transformed variables.



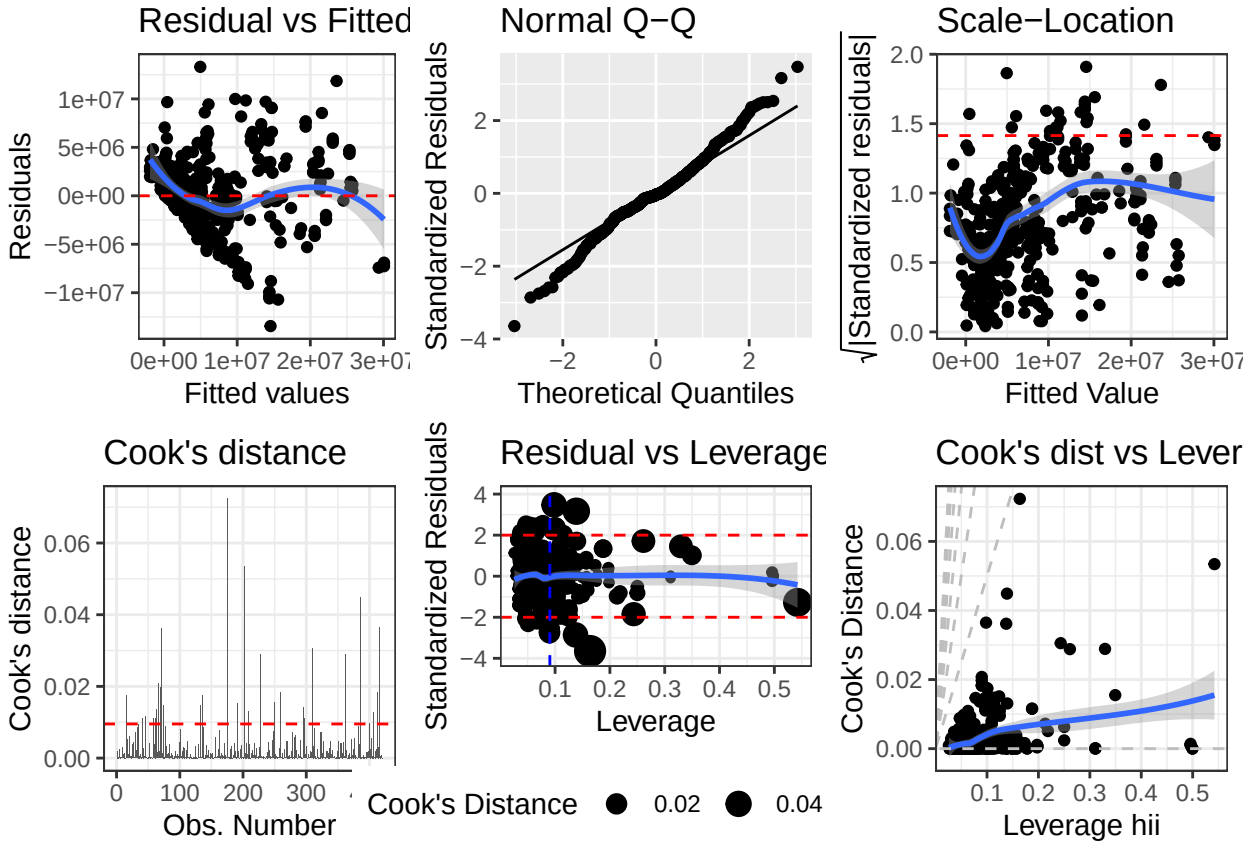
These transformations produced the following model:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-777565.797	2300168.006	-0.3380474	0.7355122
FTA	838280.225	224685.485	3.7309051	0.0002196
VORP	1809814.693	342662.537	5.2816240	0.0000002
TeamPayrollLow	99146.006	1415810.408	0.0700277	0.9442080
TeamPayrollMedium	1376560.015	1522890.392	0.9039127	0.3666084
BLK	-1126799.337	319231.513	-3.5297246	0.0004665
Rk	3134.105	1987.812	1.5766604	0.1156971
USG._T	180112.991	148820.622	1.2102690	0.2269201
AST_T	-1247062.632	719661.002	-1.7328473	0.0839256
STL._T	1683351.010	689423.255	2.4416801	0.0150697
DWS	-413083.629	477509.069	-0.8650802	0.3875349
TOV._T	560918.460	298732.413	1.8776619	0.0611852
PTS:AgeBracket<24	-149943.323	91541.408	-1.6379836	0.1022442
PTS:AgeBracket24-28	17814.289	70385.370	0.2530965	0.8003291
PTS:AgeBracket29-36	-138404.188	79839.923	-1.7335211	0.0838058
PTS:AgeBracket37+	-839849.339	239565.050	-3.5057256	0.0005092
AgeBracket<24:GS.	486357.332	2118214.515	0.2296072	0.8185193
AgeBracket24-28:GS.	4003147.780	1461002.569	2.7400005	0.0064306
AgeBracket29-36:GS.	3986465.065	1336152.199	2.9835411	0.0030314
AgeBracket37+:GS.	-10862393.922	5134922.056	-2.1153961	0.0350396
AgeBracket<24:MPG	37886.187	88593.896	0.4276388	0.6691538
AgeBracket24-28:MPG	199510.844	73288.729	2.7222582	0.0067792
AgeBracket29-36:MPG	378698.823	79521.852	4.7621982	0.0000027
AgeBracket37+:MPG	1251102.999	279395.673	4.4778897	0.0000100

	Estimate	Std. Error	t value	Pr(>|t|)
PositionBig:AST.	230794.735	65536.291	3.5216325	0.0004805
PositionForward:AST.	276450.014	117186.068	2.3590689	0.0188209
PositionGuard:AST.	-76315.086	39371.535	-1.9383315	0.0533154
PositionBig:FTr	-4591898.451	2994599.512	-1.5333932	0.1260025
PositionForward:FTr	-18126091.733	5555090.512	-3.2629696	0.0012013
PositionGuard:FTr	-2660717.841	2288905.084	-1.1624413	0.2457782
TeamPayrollHigh:AgeBracketSimp24-28	-6891558.120	2232621.464	-3.0867562	0.0021704
TeamPayrollLow:AgeBracketSimp24-28	-3840342.870	1823384.202	-2.1061622	0.0358388
TeamPayrollMedium:AgeBracketSimp24-28	-5521337.563	1984100.139	-2.7827918	0.0056555
TeamPayrollHigh:AgeBracketSimp29+	4096952.565	2687956.352	1.5241887	0.1282847
TeamPayrollLow:AgeBracketSimp29+	-2848283.708	2020939.400	-1.4093860	0.1595304
TeamPayrollMedium:AgeBracketSimp29+	-3202818.414	2112497.856	-1.5161286	0.1303096

Table 15: Model Version 33

Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
4105000	0.7416098	0.7180586	0.73166



The diagnostic plots look slightly improved, but the model is still not valid.

Model Version 38

Between Model Version 33 and Model Version 38, some changes included adjusting the Position variable, taking out interactions and predictors as recommended by backwards AIC, and adding in new interactions and predictors. The Kaggle R^2 for Model Version 37 was 0.75746, but the same violations are present. For this model, I aimed to improve both the validity and the R^2. To do this, I decided to remove bad leverage points. The following table displays the number of leverage points and outliers

Leverage/Outliers	Yes	No
Yes	16	125
No	15	264

The following 16 observations are considered bad leverage points for Model Version 37: 15, 16, 59, 63, 71, 75, 132, 136, 176, 191, 260, 310, 386, 407, and 416.

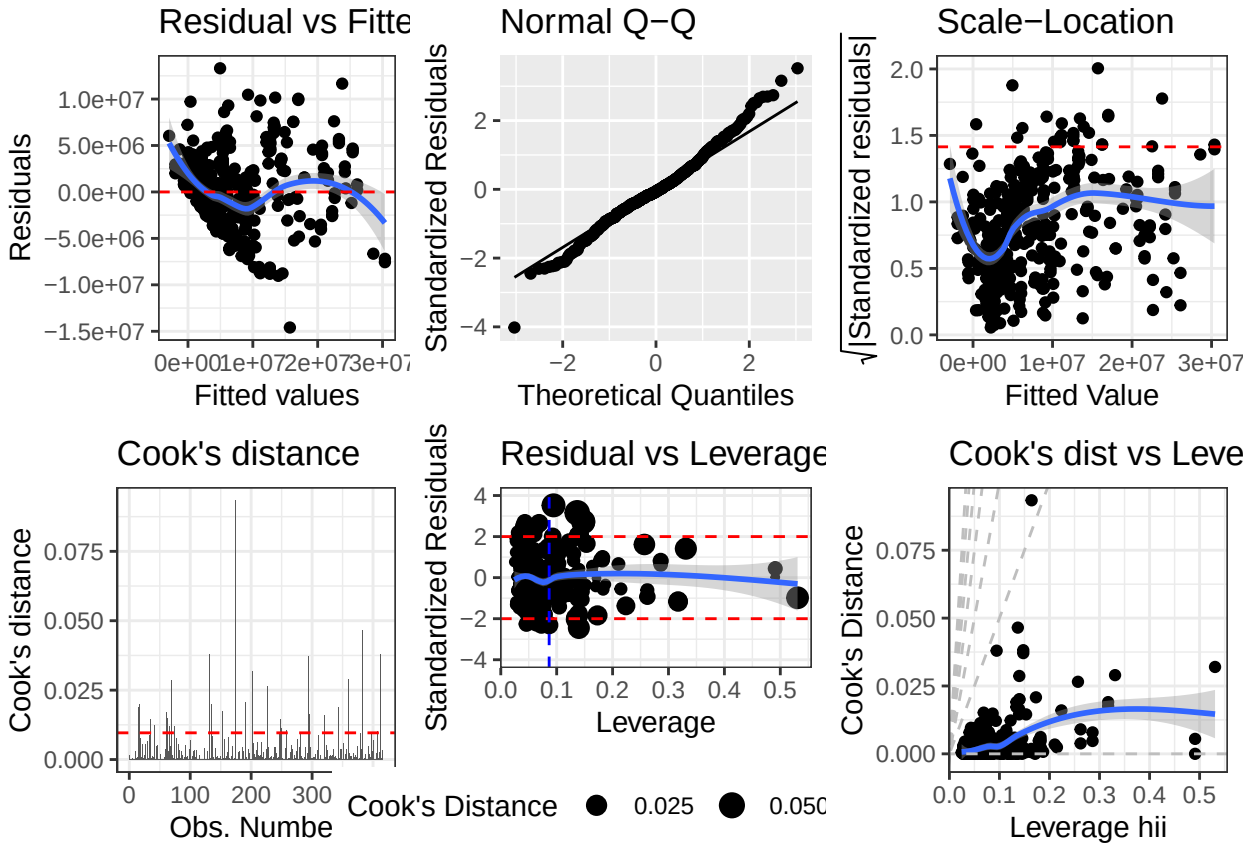
After trying to remove all of them and receiving a drop in R^2, I decided to only remove the worst leverage points: Observations 59, 260, 310, and 407. Here is the resulting model.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-530.335	2131976.806	-0.0002488	9.998017e-01
TeamPayrollLow	342915.861	1348397.079	0.2543137	7.993900e-01
TeamPayrollMedium	1283600.763	1432800.150	0.8958687	3.708871e-01
GS.	1972361.795	901336.687	2.1882631	2.925554e-02
FTA	975205.143	211365.394	4.6138354	5.403902e-06
VORP	2825675.509	560404.414	5.0422078	7.123015e-07
BLK	-1164243.544	313623.571	-3.7122323	2.360012e-04
Rk	3594.660	1910.171	1.8818528	6.061606e-02
AST_T	-1536326.031	698429.662	-2.1996861	2.842757e-02
STL._T	1573675.928	667792.344	2.3565348	1.895052e-02
TOV._T	546444.998	284880.341	1.9181562	5.583639e-02
PTS:AgeBracket<24	-61665.437	81414.420	-0.7574265	4.492614e-01
PTS:AgeBracket24-28	48093.221	67487.913	0.7126198	4.765161e-01
PTS:AgeBracket29-36	-100126.258	74089.819	-1.3514172	1.773616e-01
PTS:AgeBracket37+	-723561.474	241456.280	-2.9966563	2.907798e-03
AgeBracket<24:MPG	-17140.828	59018.144	-0.2904332	7.716425e-01
AgeBracket24-28:MPG	308103.066	65096.301	4.7330349	3.120453e-06
AgeBracket29-36:MPG	453512.009	71881.444	6.3091667	7.773321e-10
AgeBracket37+:MPG	846527.140	242484.703	3.4910538	5.373341e-04
PositionBig:AST.	179802.293	72620.820	2.4759056	1.372249e-02
PositionForward:AST.	552028.617	131796.231	4.1885008	3.490173e-05
PositionGuard:AST.	-60051.372	38005.275	-1.5800799	1.149162e-01
PositionBig:FTr	-5006960.451	2899749.270	-1.7266874	8.503227e-02
PositionForward:FTr	-22846163.537	5522901.690	-4.1366232	4.337551e-05
PositionGuard:FTr	-3655272.780	2156425.383	-1.6950611	9.087874e-02
TeamPayrollHigh:AgeBracketSimp24-28	-6300594.580	2195367.248	-2.8699502	4.333675e-03
TeamPayrollLow:AgeBracketSimp24-28	-4876843.638	1610069.330	-3.0289650	2.620704e-03
TeamPayrollMedium:AgeBracketSimp24-28	-6068169.299	1715854.332	-3.5365294	4.552449e-04
TeamPayrollHigh:AgeBracketSimp29+	4027920.266	2643164.450	1.5239007	1.283609e-01
TeamPayrollLow:AgeBracketSimp29+	-2506457.594	1832490.834	-1.3677872	1.721828e-01
TeamPayrollMedium:AgeBracketSimp29+	-3034721.140	1882477.059	-1.6120893	1.077683e-01

	Estimate	Std. Error	t value	Pr(>|t|)
PositionBig:WS	-507804.699	273946.881	-1.8536612	6.455840e-02
PositionForward:WS	-1301861.465	407501.109	-3.1947434	1.515832e-03
PositionGuard:WS	-658402.815	312984.410	-2.1036281	3.606421e-02

Table 18: Model Version 38

Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
3972000	0.7461839	0.7242574	0.77175



Model Version 39

Next, we simplified Model Version 38 by performing the backwards BIC step function, which suggested that we remove the PTS:AgeBracket interaction. This gave us exactly 30 betas in Model Version 39, which I am using as my final model.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-1197915.582	1959777.961	-0.6112507	0.5413934
TeamPayrollLow	170885.688	1337669.467	0.1277488	0.8984143
TeamPayrollMedium	996594.816	1425032.540	0.6993488	0.4847553
GS.	2045279.651	906076.175	2.2572933	0.0245480
FTA	888348.127	149681.279	5.9349314	0.0000000
VORP	2861747.513	558699.488	5.1221588	0.0000005

	Estimate	Std. Error	t value	Pr(>|t|)
BLK	-1034578.552	310161.049	-3.3356173	0.0009336
Rk	4378.202	1890.543	2.3158435	0.0210901
AST_T	-1320348.136	687819.331	-1.9196148	0.0556434
STL._T	1866078.754	646550.782	2.8862060	0.0041183
TOV._T	529254.649	284207.990	1.8622089	0.0633333
AgeBracket<24:MPG	-27698.219	56583.950	-0.4895066	0.6247611
AgeBracket24-28:MPG	345069.434	61028.076	5.6542735	0.0000000
AgeBracket29-36:MPG	470875.566	65320.786	7.2086635	0.0000000
AgeBracket37+:MPG	219319.537	111643.275	1.9644671	0.0501930
PositionBig:AST.	146577.134	70188.413	2.0883381	0.0374222
PositionForward:AST.	564331.966	132087.965	4.2723950	0.0000244
PositionGuard:AST.	-72549.483	37393.581	-1.9401588	0.0530884
PositionBig:FTr	-5154693.664	2575401.980	-2.0015103	0.0460369
PositionForward:FTr	-21788273.152	5388724.289	-4.0433082	0.0000636
PositionGuard:FTr	-3205474.928	1879261.511	-1.7057099	0.0888666
TeamPayrollHigh:AgeBracketSimp24-28	-5146334.947	2024142.226	-2.5424769	0.0113968
TeamPayrollLow:AgeBracketSimp24-28	-3598500.673	1247949.020	-2.8835318	0.0041526
TeamPayrollMedium:AgeBracketSimp24-28	-4626907.655	1305403.737	-3.5444265	0.0004417
TeamPayrollHigh:AgeBracketSimp29+	964052.286	2229682.038	0.4323721	0.6657124
TeamPayrollLow:AgeBracketSimp29+	-3552570.824	1390247.107	-2.5553521	0.0109906
TeamPayrollMedium:AgeBracketSimp29+	-4103018.664	1400911.081	-2.9288216	0.0036043
PositionBig:WS	-492304.532	271395.207	-1.8139765	0.0704576
PositionForward:WS	-1540925.826	394605.092	-3.9049821	0.0001112
PositionGuard:WS	-727599.676	309199.673	-2.3531709	0.0191150

Table 20: Model Version 39

Residual Standard Error	R^2	Adjusted R^2	Kaggle R^2
4013000	0.7382043	0.7185357	0.76852

Results and Discussion

Final Model

Our final model, Model Version 39, is given by:

$$\hat{Salary} = -1197916 + 170886(TeamPayrollLow) + 996595(TeamPayrollMedium) + 2045280(GS.) + 888348(FTA)$$
$$+2861748(VORP) - 1034579(BLK) + 4378(Rk) - 1320348(AST\_T) + 1866079(STL.\_T) + 529255(TOV.\_T)$$
$$-27698(AgeBracket < 24 * MPG) + 345069(AgeBracket24-28 * MPG) + 470876(AgeBracket29-36 * MPG)$$
$$+219320(AgeBracket37 + * MPG) + 146577(PositionBig * AST.) + 564332(PositionForward * AST.)$$
$$-72550(PositionGuard * AST.) - 5154694(PositionBig * FTr) - 21788273(PositionForward * FTr)$$
$$-3205475(PositionGuard * FTr) - 5146335(TeamPayrollHigh * AgeBracketSimp24 - 28)$$
$$-3598501(TeamPayrollLow * AgeBracketSimp24-28) - 4626908(TeamPayrollMedium * AgeBracketSimp24-28)$$

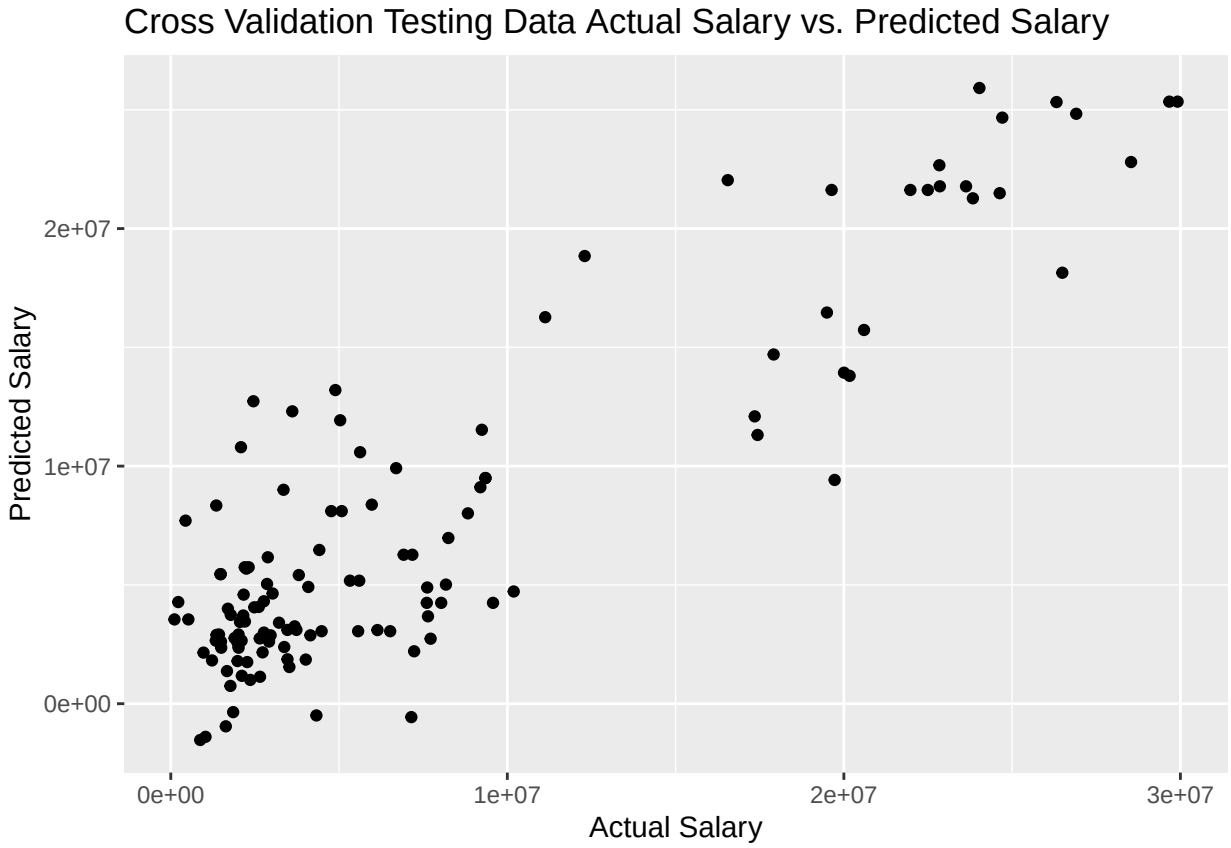
$$+964052(TeamPayrollHigh * AgeBracketSimp29+) - 3552571(TeamPayrollLow * AgeBracketSimp29+)$$
$$-4103019(TeamPayrollMedium * AgeBracketSimp29+) - 492304(PositionBig * WS)$$
$$-1540926(PositionForward * WS) - 727600(PositionGuard * WS)$$

Below is the Analysis of Variance Table for the final model.

Table 21: ANOVA Table for Final Model

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
TeamPayroll	2	6.205750e+14	3.102875e+14	19.2684104	0.0000000
GS.	1	8.988487e+15	8.988487e+15	558.1721918	0.0000000
FTA	1	1.208491e+15	1.208491e+15	75.0455512	0.0000000
VORP	1	1.882040e+15	1.882040e+15	116.8719645	0.0000000
BLK	1	9.642974e+13	9.642974e+13	5.9881488	0.0148473
Rk	1	2.866115e+11	2.866115e+11	0.0177982	0.8939389
AST_T	1	1.801656e+13	1.801656e+13	1.1188023	0.2908380
STL._T	1	2.157218e+10	2.157218e+10	0.0013396	0.9708224
TOV._T	1	6.964931e+12	6.964931e+12	0.4325123	0.5111508
AgeBracket:MPG	4	2.655656e+15	6.639139e+14	41.2281046	0.0000000
Position:AST.	3	7.040922e+14	2.346974e+14	14.5743721	0.0000000
Position:FTr	3	5.320351e+14	1.773450e+14	11.0128739	0.0000006
TeamPayroll:AgeBracketSimp	6	5.614656e+14	9.357760e+13	5.8110352	0.0000082
Position:WS	3	2.529336e+14	8.431121e+13	5.2356057	0.0014939
Residuals	386	6.215924e+15	1.610343e+13	NA	NA

Using cross validation, we split the initial training data set into separate training and testing sets using the seed 313.



Using this technique, we achieve the following R^2 values:

Table 22: Final Model R^2 Values

My Training R^2	My Testing R^2	Full Training R^2	Kaggle R^2
0.6848591	0.7986384	0.7382043	0.76852

We examine VIF values to see if there is an issue with multicollinearity.

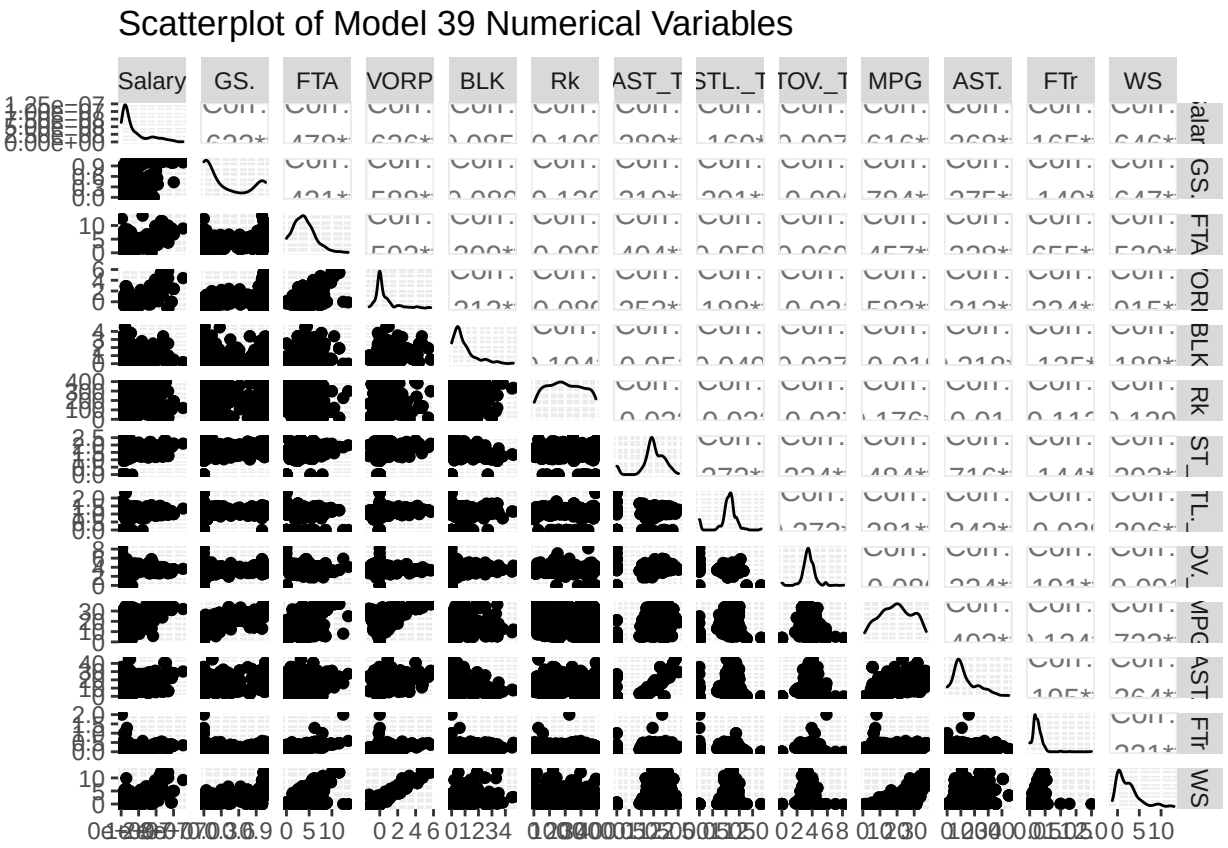
Table 23: Model Version 39 VIF

	GVIF	Df	GVIFˆ(1/(2*Df))
TeamPayroll	14.434282	2	1.949166
GS.	3.579298	1	1.891903
FTA	3.428254	1	1.851554
VORP	11.495898	1	3.390560
BLK	1.749856	1	1.322821
Rk	1.218925	1	1.104050
AST_T	3.232612	1	1.797946
STL._T	1.474057	1	1.214107
TOV._T	1.573365	1	1.254339
AgeBracket:MPG	271.260122	4	2.014528
Position:AST.	72.773769	3	2.043286
Position:FTr	38.904128	3	1.840769
TeamPayroll:AgeBracketSimp	929.066477	6	1.767410

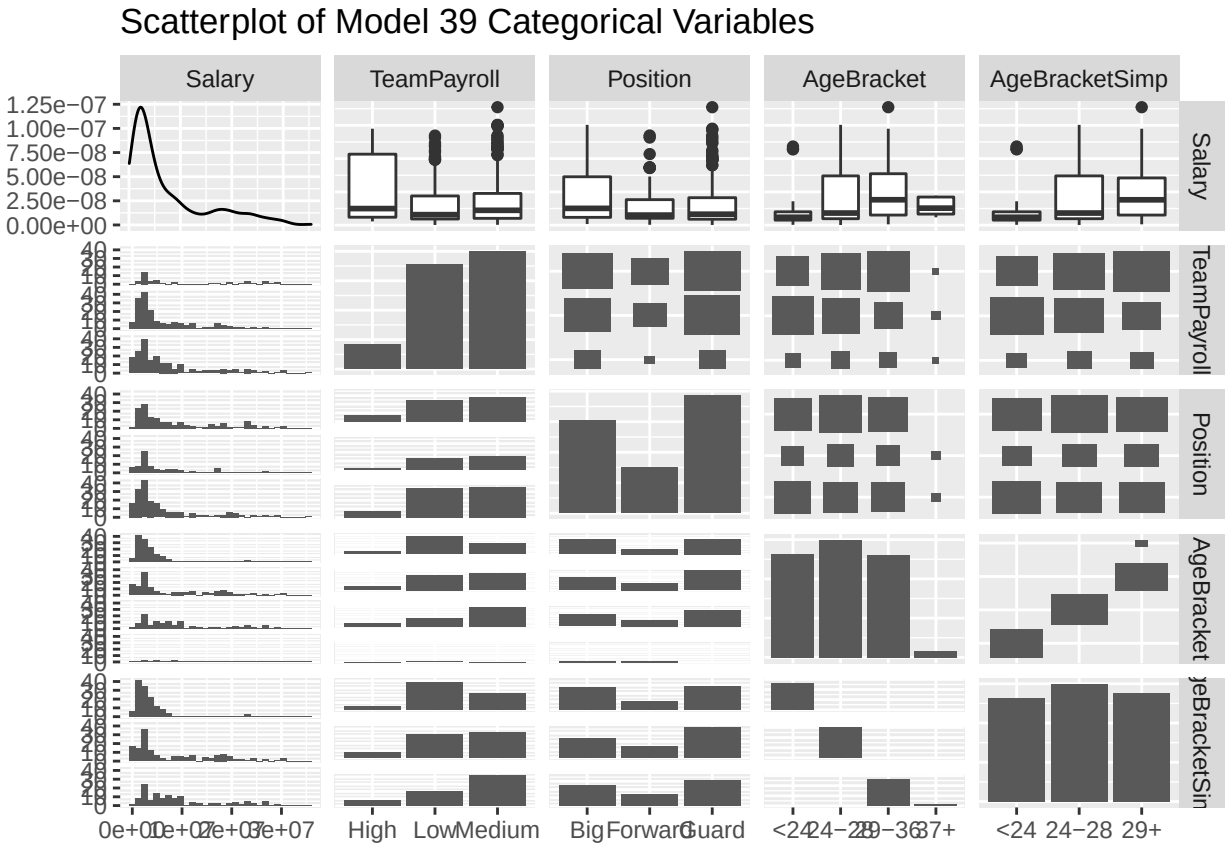
	GVIF	Df	GVIF^(1/(2*Df))
Position:WS	105.758804	3	2.174634

All values are below the threshold of 5, so there is no multicollinearity issue.

Here is a scatterplot of the numerical predictors used in the final model:



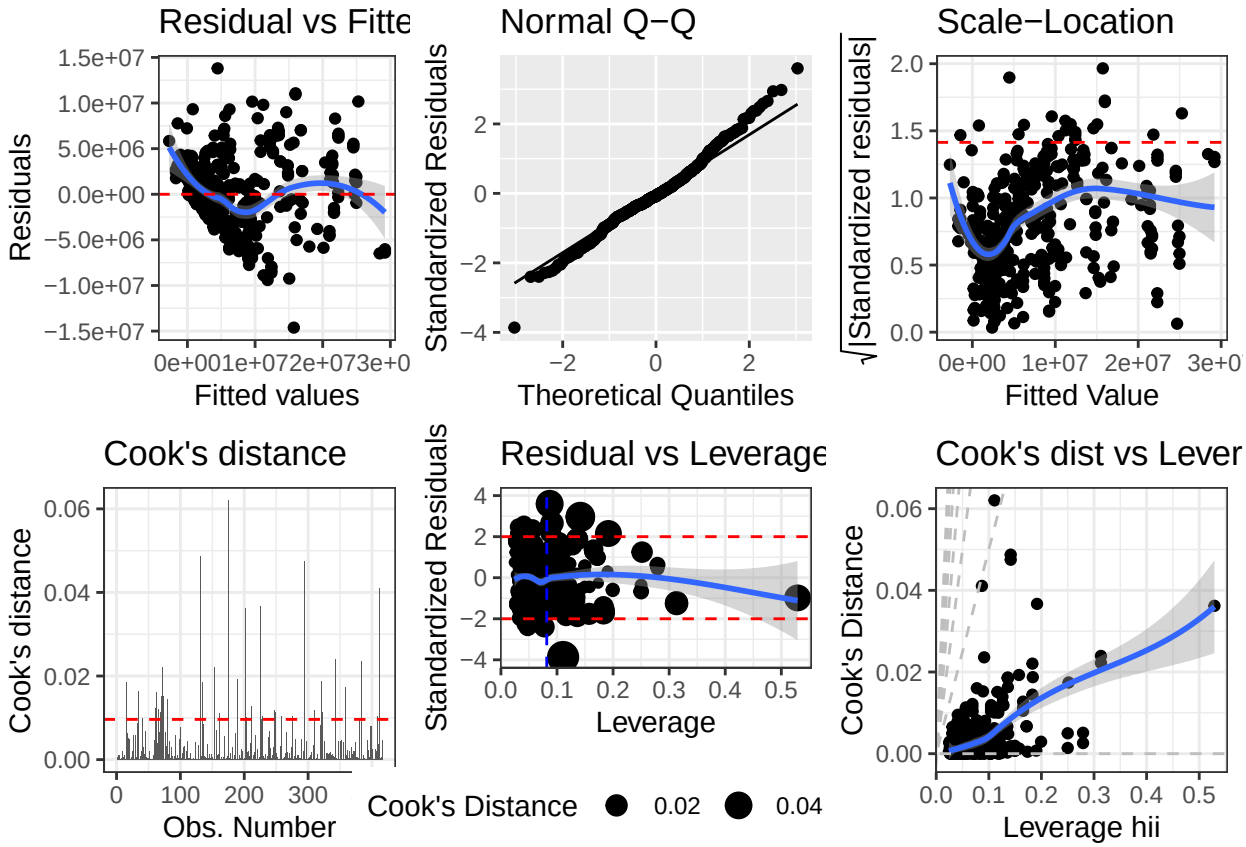
Here is a scatterplot of the categorical predictors used in the final model:



Diagnostics

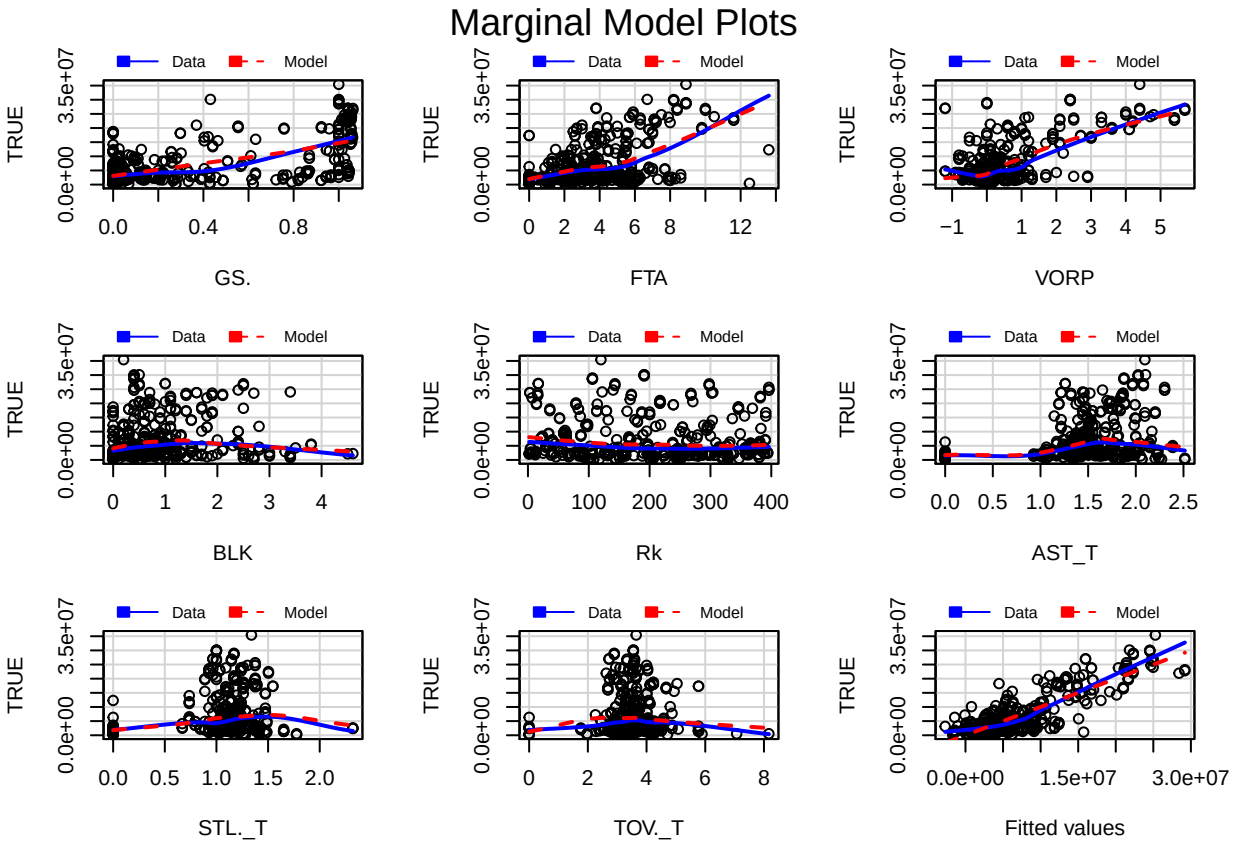
Leverage/Outliers	Yes	No
Yes	7	121
No	15	273

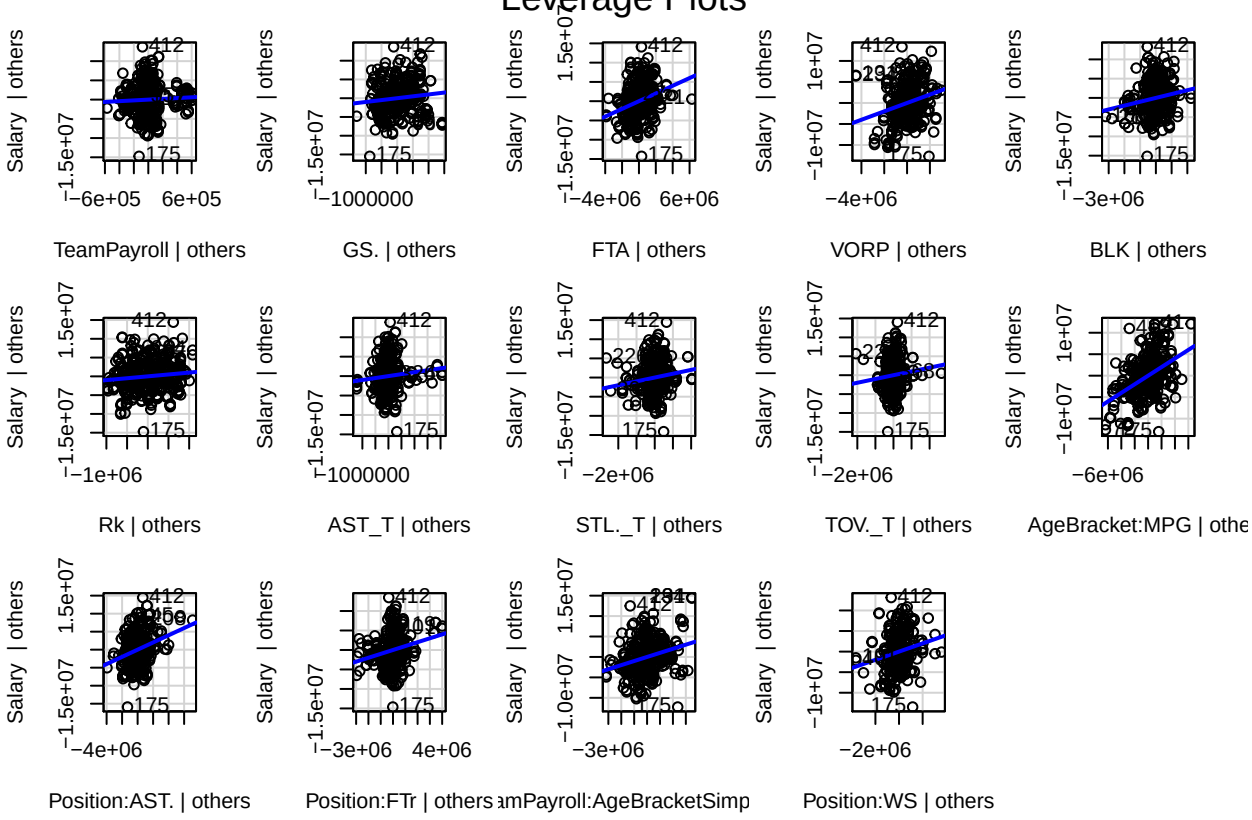
There are only 7 bad leverage points. We examine the diagnostics plots.



While there are still violations, the model is greatly improved from the early stages. The assumption of normality is met, so the Box-Cox transformations seemed to have been beneficial. There are still violations of random residuals and constant variance, although the blue lines are flatter than before.

Here are the marginal model plots and the leverage plots.


Marginal Model Plots
Data
Model
TRUE
GS.
FTA
VORP
BLK
Rk
AST_T
STL._T
TOV._T
Fitted values


Leverage Plots
Salary | others
TeamPayroll | others
GS. | others
FTA | others
VORP | others
BLK | others
Rk | others
AST_T | others
STL._T | others
TOV._T | others
AgeBracket:MPG | othe
Position:AST. | others
Position:FTr | others
mPayroll:AgeBracketSimp
Position:WS | others

Limitations and Conclusions

My final model was able to achieve the highest R^2 in the Kaggle competition. However, this score was still only 0.76852, and the BIC score was 12800.37. This suggests that there was a lot of unexplained error in my model. This dataset proved to be difficult to fit a strong model to, at least with the tools available to us, as the majority of R^2 scores on Kaggle were below 0.60. My R^2 of 0.77 suggests that my model was a better fit than most and did a good job at predicting NBA player salaries, but it was still not a great fit as the R^2 was below 0.80 and there were also still several outstanding problems.

There are several limitations to this model. First, there are still violations that detract from the validity of the model. As seen in the diagnostic plots, the residuals are not randomly distributed and the assumption of constant variance is also violated. If I were to try to improve this model in the future, I would go back to the y-variable transformation and spend more time trying to modify that model to build up its robustness. I had been able to achieve a Kaggle R^2 of 0.56893 using the transformation, which still would have placed in the top half of the class, and I feel like I could have gotten that number higher if I continued to work on that version of the model instead of abandoning it.

Also, as seen in the summary of the final model, there are several predictors that are not significant. Although removing them altogether resulted in a lower R^2, I would explore other options such as replacing them with new variables and interactions to simplify my model without sacrificing too much error. I also had 30 betas, which was the maximum allowed without penalty. Reducing this number would be another goal in trying to improve this model.

While my model may have had several limitations, I was proud of the R^2 score that I was able to achieve after 42 different model versions and over a hundred hours of work. This was a very enjoyable and formative project on a topic that I am passionate about, and instilled in me a newfound interest in statistics.

References

Almohalwas, Akram. 2020. *Chapter 5 Updated Winter 2020.*
Basketball Reference. 2021. *Basketball Statistics and History.* www.basketball-reference.com/.
Kaggle. 2021. *NBA Players' Salaries.* https://www.kaggle.com/c/nba-players-salaries/data.
Sheather, Simon J. 2009. *A Modern Approach to Regression with R.* New York: Springer.