Predicting NBA Statistics

David Adewole
Klaudia Czabanski
Samuel Hannah
Angely Lee

TABLE OF CONTENTS

INTRODUCTION

In professional basketball, using data analytics to predict team success and individual player performance has revolutionized decision-making for coaches, analysts, and management teams. As the competitive landscape evolves, understanding the relationship between player statistics and team wins has become essential for optimizing game strategies and roster decisions. This study is centered on creating and evaluating predictive models that estimate the impact of player performance metrics on team success. By analyzing statistical trends, patterns, and correlations, the research aims to uncover insights—such as identifying the most significant performance metrics—into how individual contributions drive team outcomes.

The findings from this analysis are intended to support key decision-makers, such as team owners and general managers, as they navigate critical aspects of team development, including player selection and game strategy formulation. With data-driven insights, these stakeholders can make more informed decisions, improving their chances of achieving both short-term victories and long-term success. This work underscores the value of measurable data in transforming basketball management, providing a framework for forecasting game outcomes and enhancing performance evaluation processes.

DATA COLLECTION

The dataset "Predictive Modeling of NBA Player Performance[1]," sourced from Kaggle, was utilized to analyze player performance metrics and their relationship to team success. This dataset was selected for its comprehensive collection of player statistics, offering valuable insights for developing predictive models aimed at estimating the impact of individual contributions on team wins.

The dataset includes a variety of variables such as player names, positions, age, team affiliations, and key performance indicators like field goals, three-point shots, assists, turnovers, and more. These variables provided the foundation for exploring the factors that influence team performance. To enhance compatibility and efficiency during analysis in SAS, some variable names were modified. Additionally, the few missing values in the dataset, such as player positions marked as "N/A," were addressed by conducting research to fill in the gaps accurately.

The targeted outcome variable for this study is the total number of games won by each player ("W"), which serves as a proxy for evaluating individual contributions to team success. Other performance metrics, such as shooting percentages, rebounds, and assists, were examined to identify their influence on this outcome. By leveraging this dataset, the study aims to uncover meaningful patterns and correlations that highlight the drivers of team wins.

[1] https://www.kaggle.com/code/xreina8/predictive-modeling-of-nba-player-performance/input

This dataset offers a diverse foundation for predictive modeling, enabling the exploration of key relationships between player performance and team success, which can guide better decision-making in professional basketball management. An explanation of all the variables is as follows:

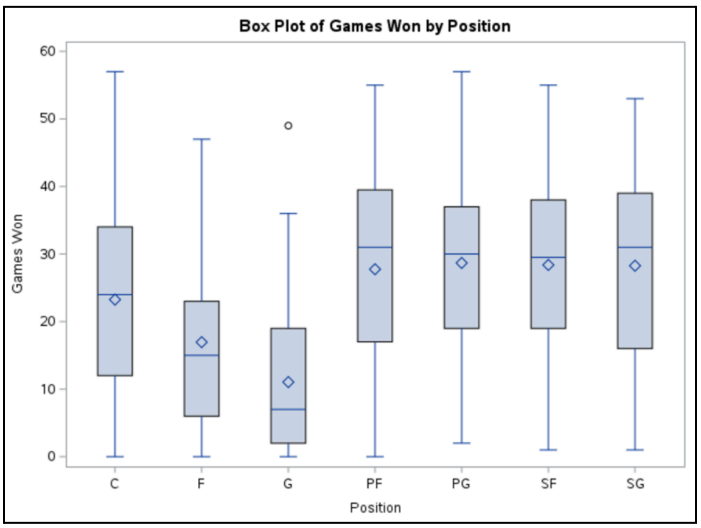
Label	**Description**
PName	The name of the basketball player
POS	The player's position in the game, including 'N/A'
Team	The abbreviation of the team the player is currently playing for this season
Age	The age of the player
GP	The total number of games the player has played in this season
W	The total number of games won by the player
L	The total number of games lost by the player
Min	The total minutes the player has played in this season
PTS	The total points made by the player [target]
FGM	The total number of field goals made by the player
FGA	The total number of field goals attempted by the player
FG% (FGP)	The percentage of successful field goals made by the player
3PM (_3PM)	The total number of 3-point field goals made by the player
3PA (_3PA)	The total number of 3-point field goals attempted by the player
3P% (_3PP)	The percentage of successful 3-point field goals made by the player
FTM	The total number of free throws made by the player
FTA	The total number of free throws attempted by the player
FT% (FTP)	The percentage of successful free throws made by the player
OREB	The total number of offensive rebounds made by the player
DREB	The total number of defensive rebounds made by the player
REB	The total number of rebounds (offensive + defensive) made by the player

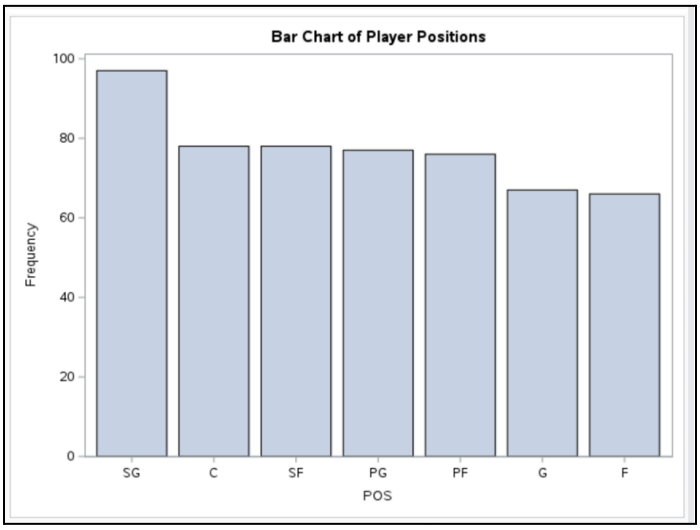
AST	The total number of assists made by the player
TOV	The total number of turnovers made by the player
STL	The total number of steals made by the player
BLK	The total number of blocks made by the player
PF	The total number of personal fouls made by the player
FP	The total number of NBA fantasy points made by the player
DD2	The total number of double-doubles made by the player
TD3	The total number of triple-doubles made by the player
+/- (Efficiency)	The total difference between the player's team scoring and the opponents' scoring while the player is in the game

DATA EXPLORATION AND VISUALIZATION

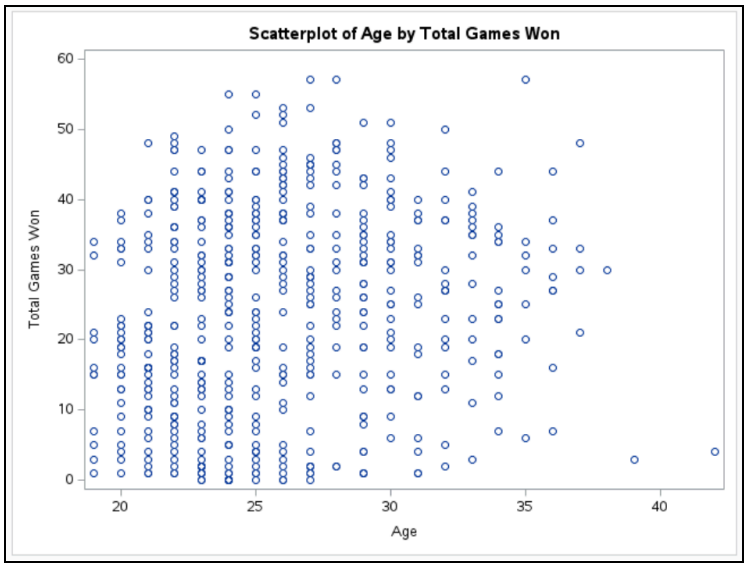
Before creating any predictive models, we conducted an in-depth exploration of the dataset “Predictive Modeling of NBA Player Performance” using various graphs and statistical summaries. This step ensured that the data was logically consistent and helped us gain a comprehensive understanding of each variable to facilitate accurate model interpretation later.

The variable of interest in our study is the total number of games won by the player ("W"). During the initial analysis, we examined its distribution across different player positions, visualized through a box plot. This plot revealed positional trends, showing that players in certain positions, such as point guards (PG) and shooting guards (SG), had higher median wins compared to other positions. The variations in the box plot also show that the minimum number of wins is 0 for some positions. The players that have zero wins may not be included in the predictive model. Similarly, the bar chart of player positions highlighted that shooting guards were the most frequently recorded position in the dataset.





A scatterplot of total games won versus player age provided additional insights into the relationship between experience and performance. Younger players demonstrated a wide range of total wins, with some achieving high success early in their careers while others showed limited impact. This variability may reflect differences in playing time, team dynamics, or individual skill levels among less experienced players. On the other hand, older players tended to exhibit less variation in total games won, with a slight decline in performance consistency. This trend suggests that, while experience and maturity might stabilize a player's contribution, physical limitations or reduced playing time could impact their ability to contribute to team success consistently. These findings highlight the importance of balancing age and position when evaluating player contributions, offering valuable insights for understanding how different factors influence the key outcome variable of total games won.



Correlation analysis was conducted to identify multicollinearity among the performance-related variables and ensure the model's stability. Metrics such as field goals made (FGM), free throws attempted (FTA), and assists (AST) demonstrated high correlation coefficients (above 0.90) with other variables, indicating significant redundancy. For example, FGM and field goals attempted (FGA) were strongly correlated, as were FTA and free throws made (FTM). These high correlations suggest that including all such variables could lead to overfitting and decrease the predictive power of the model by introducing unnecessary complexity. The correlation matrix also revealed strong relationships between other variables, such as points scored (PTS) and fantasy points (FP), further underscoring the need to select input features carefully. To address this, redundant variables were excluded or considered for dimensionality reduction to improve model performance. Certain variables like "losses" (L) and team abbreviations (Team) were excluded from the modeling process. "Losses" were found to be inversely correlated with the outcome variable (wins), but its inclusion could introduce multicollinearity since it is not an independent predictor of individual player performance. Similarly, team identifiers were excluded because they did not contribute meaningful information to the predictive analysis of individual metrics.

Through these exploratory analyses, we uncovered important patterns and trends crucial in guiding variable selection and shaping modeling strategies. By identifying key relationships and addressing potential redundancies, we have established a solid foundation for developing accurate, efficient, and reliable predictive models.

Summary of Wins by Team

Analysis Variable : W W

Team	N Obs	N	Mean	Median	Minimum	Maximum
ATL	18	18	24.722	29.500	1.000	41.000
BKN	20	20	24.750	27.500	0.000	42.000
BOS	18	18	33.778	37.500	3.000	57.000
CHA	17	17	14.294	16.000	0.000	24.000
CHI	17	17	24.765	32.000	3.000	40.000
CLE	17	17	29.471	31.000	3.000	48.000
DAL	21	21	19.857	20.000	1.000	37.000
DEN	16	16	36.125	37.000	13.000	53.000
DET	19	19	8.895	10.000	0.000	16.000
GSW	17	17	27.118	30.000	3.000	44.000
HOU	15	15	13.733	16.000	0.000	22.000
IND	18	18	21.889	27.500	1.000	36.000
LAC	18	18	26.278	29.500	2.000	44.000
LAL	18	18	26.722	30.500	1.000	45.000
MEM	18	18	32.000	37.500	0.000	51.000
MIA	17	17	24.353	28.000	4.000	42.000
MIL	18	18	33.056	33.000	4.000	57.000
MIN	18	18	25.222	27.000	7.000	41.000
NOP	16	16	26.438	27.000	2.000	40.000
NYK	16	16	31.000	36.500	3.000	47.000
OKC	16	16	24.875	26.000	3.000	37.000
ORL	17	17	18.765	20.000	1.000	33.000
PHI	18	18	32.278	38.500	1.000	52.000
PHX	16	16	30.813	33.000	15.000	44.000
POR	21	21	16.905	15.000	0.000	38.000
SAC	20	20	27.950	30.000	1.000	48.000
SAS	21	21	12.190	12.000	1.000	31.000
TOR	18	18	24.667	28.000	4.000	39.000
UTA	20	20	17.250	18.000	0.000	34.000
WAS	20	20	17.300	21.500	0.000	33.000

Summary of Games Played by Team

Analysis Variable : GP GP

Team	N Obs	N	Mean	Median	Minimum	Maximum
ATL	18	18	52.167	64.000	2.000	80.000
BKN	20	20	47.400	50.500	1.000	83.000
BOS	18	18	49.778	62.000	4.000	82.000
CHA	17	17	42.294	46.000	4.000	73.000
CHI	17	17	50.941	67.000	5.000	82.000
CLE	17	17	46.353	48.000	3.000	79.000
DAL	21	21	41.952	45.000	1.000	78.000
DEN	16	16	57.125	61.000	17.000	80.000
DET	19	19	41.737	47.000	1.000	76.000
GSW	17	17	49.529	57.000	4.000	82.000
HOU	15	15	52.467	59.000	4.000	82.000
IND	18	18	49.444	62.000	3.000	80.000
LAC	18	18	52.167	56.000	4.000	81.000
LAL	18	18	49.389	56.000	4.000	81.000
MEM	18	18	51.111	58.500	1.000	80.000
MIA	17	17	45.588	54.000	7.000	80.000
MIL	18	18	47.500	52.000	7.000	81.000
MIN	18	18	50.167	53.000	15.000	79.000
NOP	16	16	52.500	60.000	5.000	79.000
NYK	16	16	56.250	65.500	3.000	82.000
OKC	16	16	52.125	55.000	6.000	76.000
ORL	17	17	44.765	51.000	2.000	80.000
PHI	18	18	51.167	59.000	1.000	80.000
PHX	16	16	56.375	59.500	25.000	79.000
POR	21	21	41.857	52.000	1.000	80.000
SAC	20	20	48.100	53.000	2.000	82.000
SAS	21	21	40.905	38.000	4.000	73.000
TOR	18	18	49.889	55.500	8.000	77.000
UTA	20	20	38.100	44.500	1.000	74.000
WAS	20	20	41.600	50.000	2.000	78.000

Summary of Wins by Player Position

The MEANS Procedure

Analysis Variable : W W

POS	N Obs	N	Mean	Median	Minimum	Maximum
C	78	78	23.256	24.000	0.000	57.000
F	66	66	16.955	15.000	0.000	47.000
G	67	67	11.045	7.000	0.000	49.000
PF	76	76	27.750	31.000	0.000	55.000
PG	77	77	28.675	30.000	2.000	57.000
SF	78	78	28.385	29.500	1.000	55.000
SG	97	97	28.268	31.000	1.000	53.000

Summary of Games Played by Player Position

The MEANS Procedure

Analysis Variable : GP GP

POS	N Obs	N	Mean	Median	Minimum	Maximum
C	78	78	47.462	51.000	1.000	82.000
F	66	66	35.273	34.500	1.000	82.000
G	67	67	25.015	16.000	1.000	76.000
PF	76	76	54.487	62.000	1.000	82.000
PG	77	77	55.338	60.000	4.000	82.000
SF	78	78	56.462	63.000	1.000	83.000
SG	97	97	55.485	61.000	1.000	82.000

Pearson Correlation Coefficients, N = 539
Prob > |r| under H0: Rho=0

	Min	FGM	FGA	_3PM	_3PA	FTM	FTA	OREB	DREB	AST	TOV	STL	BLK	PF	FP	DD2	TD3	Efficiency
Min	1.00000	0.91550	0.91899	0.76417	0.78370	0.74929	0.75713	0.58748	0.83596	0.77136	0.85819	0.86923	0.54583	0.89119	0.94404	0.50694	0.18692	0.22646
Min		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
FGM	0.91550	1.00000	0.98560	0.73479	0.75702	0.88548	0.88950	0.53117	0.82489	0.79034	0.91839	0.77981	0.51148	0.77693	0.97451	0.61454	0.27661	0.27403
FGM	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
FGA	0.91899	0.98560	1.00000	0.80744	0.83502	0.86784	0.86385	0.44125	0.76928	0.80618	0.91710	0.79238	0.43232	0.75570	0.95277	0.53745	0.23390	0.23642
FGA	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
_3PM	0.76417	0.73479	0.80744	1.00000	0.99145	0.56535	0.53231	0.09979	0.47558	0.62109	0.65393	0.65309	0.16792	0.55541	0.69220	0.18594	0.07910	0.23341
_3PM	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	0.0205	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0665	<.0001
_3PA	0.78370	0.75702	0.83502	0.99145	1.00000	0.59563	0.56615	0.11611	0.49800	0.64475	0.68765	0.67081	0.17606	0.57896	0.71469	0.20734	0.08782	0.20269
_3PA	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	0.0070	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0415	<.0001
FTM	0.74929	0.88548	0.86784	0.56535	0.59563	1.00000	0.99135	0.42733	0.69330	0.72284	0.85189	0.64117	0.41685	0.62712	0.86735	0.59872	0.29274	0.27463
FTM	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
FTA	0.75713	0.88950	0.86385	0.53231	0.56615	0.99135	1.00000	0.48943	0.73276	0.71557	0.86056	0.64227	0.45990	0.65577	0.87909	0.63761	0.30942	0.26435
FTA	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
OREB	0.58748	0.53117	0.44125	0.09979	0.11611	0.42733	0.48943	1.00000	0.80197	0.29901	0.48239	0.46724	0.73304	0.71499	0.63024	0.67218	0.18984	0.13434
OREB	<.0001	<.0001	<.0001	0.0205	0.0070	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0018
DREB	0.83596	0.82489	0.76928	0.47558	0.49800	0.69330	0.73276	0.80197	1.00000	0.60706	0.77588	0.66464	0.69040	0.83612	0.88991	0.78930	0.33170	0.25213
DREB	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
AST	0.77136	0.79034	0.80618	0.62109	0.64475	0.72284	0.71557	0.29901	0.60706	1.00000	0.88211	0.77109	0.25233	0.60767	0.83154	0.54595	0.41084	0.27829
AST	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
TOV	0.85819	0.91839	0.91710	0.65393	0.68765	0.85189	0.86056	0.48239	0.77588	0.88211	1.00000	0.76750	0.43599	0.76273	0.92676	0.62797	0.34395	0.20525
TOV	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
STL	0.86923	0.77981	0.79238	0.65309	0.67081	0.64117	0.64227	0.46724	0.66464	0.77109	0.76750	1.00000	0.43223	0.77149	0.83774	0.38224	0.19001	0.24207
STL	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
BLK	0.54583	0.51148	0.43232	0.16792	0.17606	0.41685	0.45990	0.73304	0.69040	0.25233	0.43599	0.43223	1.00000	0.66590	0.59986	0.50887	0.08184	0.20687
BLK	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	0.0576	<.0001
PF	0.89119	0.77693	0.75570	0.55541	0.57896	0.62712	0.65577	0.71499	0.83612	0.60767	0.76273	0.77149	0.66590	1.00000	0.84107	0.50241	0.16008	0.15570
PF	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	0.0002	0.0003
FP	0.94404	0.97451	0.95277	0.69220	0.71469	0.86735	0.87909	0.63024	0.88991	0.83154	0.92676	0.83774	0.59986	0.84107	1.00000	0.68051	0.31699	0.30213
FP	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001
DD2	0.50694	0.61454	0.53745	0.18594	0.20734	0.59872	0.63761	0.67218	0.78930	0.54595	0.62797	0.38224	0.50887	0.50241	0.68051	1.00000	0.51498	0.25847
DD2	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001
TD3	0.18692	0.27661	0.23390	0.07910	0.08782	0.29274	0.30942	0.18984	0.33170	0.41084	0.34395	0.19001	0.08184	0.16008	0.31699	0.51498	1.00000	0.23879
TD3	<.0001	<.0001	<.0001	0.0665	0.0415	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0576	0.0002	<.0001	<.0001		<.0001
Efficiency	0.22646	0.27403	0.23642	0.23341	0.20269	0.27463	0.26435	0.13434	0.25213	0.27829	0.20525	0.24207	0.20687	0.15570	0.30213	0.25847	0.23879	1.00000
Efficiency	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0018	<.0001	<.0001	<.0001	<.0001	<.0001	0.0003	<.0001	<.0001	<.0001	

DATA ANALYSIS

Two main rounds of modeling were conducted for the outcome variable W where three models were evaluated based on their performance: linear regression, classification and regression trees (CART), and neural network. In the first round, all available variables were included as predictors except for PName and Team, as these are identified, and L as it has a direct relationship with W. Based on the results from this initial evaluation, significant predictors were identified. The second round of modeling was then performed using a reduced set of predictors, focusing on the variables deemed significant from the first round. This approach allowed for model refinement and a better understanding of the most influential predictors. For all the models created in both rounds, an 80/20 split was used on the dataset and a seed of 12345. Dummy variables were also created for the POS variable for use in the multiple regression models. Seven classes of player positions were identified, so six dummy variables were created.

Modeling Round 1

Multiple regression was chosen because the outcome variable is numeric. Here, multiple selection methods, stepwise, forward, and backward were explored to see which was the most parsimonious model in order to conclude the best multiple regression model. Three main error metrics – MAPE, MSE, MAE – were also utilized to evaluate the models. After building the models, all three selection methods performed fairly similarly. In fact, the stepwise and forward models performed exactly the same. Both had selected significant variables to be GP, FTA, and Efficiency, whereas backward selection had GP, OREB, STL, and Efficiency. Their error metrics were also the same. There were no signs of overfitting and about 89% of the variability in player wins is reflected in these models. The best final model here could be either the stepwise or forward selection model. For us, we chose stepwise to mainly focus on the second round of modeling.

Another point to highlight is that the number of observations for MAPE (train = 423, test = 104) is inconsistent with MAE and MSE (train = 432, test = 107). We thought it wasn't an issue until we created all our models for both rounds of modeling so all of the error metric tables will include MAPE. We discovered that the outcome variable, W, had 12 observations where players won zero amount of times. If we had caught this sooner, we would have chosen to remove these observations to include MAPE in our evaluations. Thus, for all of the following evaluations, we will mainly only look at MAE and MSE.

W				
W	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	12	2.23	12	2.23

Modeling Round 1 - Multiple Regression Models

Forward

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	81374	27125	1275.15	<.0001
Error	428	9104.23910	21.27159		
Corrected Total	431	90478			

Root MSE	4.61211
R-Square	0.89938
Adj R-Sq	0.89867
AIC	1758.76617
AICC	1758.90701
SBC	1341.03987
ASE (Train)	21.07463
ASE (Validate)	16.66570

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	-0.541125	0.510504	-1.06	0.2898	0
GP	1	0.529854	0.010929	48.48	<.0001	1.41375
FTA	1	-0.008181	0.002037	-4.02	<.0001	1.50947
Efficiency	1	0.033663	0.001494	22.53	<.0001	1.08318

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.451
mape_acc	104	26.105
mae_fit	432	3.432
mae_acc	107	2.998
mse_fit	432	21.075
mse_acc	107	16.666

Backward

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	4	81438	20360	961.74	<.0001
Error	427	9039.42212	21.16961		
Corrected Total	431	90478			

Root MSE	4.60104
R-Square	0.90009
Adj R-Sq	0.89916
AIC	1757.67957
AICC	1757.87722
SBC	1344.02170
ASE (Train)	20.92459
ASE (Validate)	16.12752

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	-0.816902	0.518918	-1.57	0.1162	0
GP	1	0.560131	0.015321	36.56	<.0001	2.79172
OREB	1	-0.012832	0.005153	-2.49	0.0131	1.55013
STL	1	-0.043495	0.012062	-3.61	0.0003	2.37310
Efficiency	1	0.033685	0.001486	22.67	<.0001	1.07652

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.946
mape_acc	104	25.370
mae_fit	432	3.432
mae_acc	107	2.958
mse_fit	432	20.925
mse_acc	107	16.128

Stepwise

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	81374	27125	1275.15	<.0001
Error	428	9104.23910	21.27159		
Corrected Total	431	90478			

Root MSE	4.61211
R-Square	0.89938
Adj R-Sq	0.89867
AIC	1758.76617
AICC	1758.90701
SBC	1341.03987
ASE (Train)	21.07463
ASE (Validate)	16.66570

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	-0.541125	0.510504	-1.06	0.2898	0
GP	1	0.529854	0.010929	48.48	<.0001	1.41375
FTA	1	-0.008181	0.002037	-4.02	<.0001	1.50947
Efficiency	1	0.033663	0.001494	22.53	<.0001	1.08318

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.451
mape_acc	104	26.105
mae_fit	432	3.432
mae_acc	107	2.998
mse_fit	432	21.075
mse_acc	107	16.666

Further exploration was also performed to see if narrowing down the predictors would give better results. In the following model, advanced metrics (_3PP, FGP, FTP, Efficiency) were excluded and only the basic metrics and player positions were included. In addition, participation metrics (GP, Min) and Age were excluded to see how only including basic performance metrics would be carried out. Stepwise selection was also used to see which predictors were significant. As a result, the model performed worse compared to the first models where all predictors were used to create the models. For the training and test sets, MSE increased significantly. In the final model above, MSE training was 21.075 and here it increased to 59.365, so it performed much worse with the selected predictors. MSE also performed better in the test set than training whereas it was the other way around for the "Further Exploration" model below, indicating slight overfitting. On the other hand, MAE increased slightly from train = 3.432 and test = 2.998 to train = 6.231 and test = 6.875 so it performed better when all predictors were included.

Further Exploration - Stepwise Selection with Selected Predictors:

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	6	64832	10805	179.07	<.0001
Error	425	25646	60.34252		
Corrected Total	431	90478			

Root MSE	7.76804
R-Square	0.71655
Adj R-Sq	0.71255
AIC	2212.15867
AICC	2212.49909
SBC	1806.63765
ASE (Train)	59.36475
ASE (Validate)	68.72287

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	8.953047	0.677662	13.21	<.0001	0
_3PM	1	0.065608	0.009337	7.03	<.0001	2.30496
OREB	1	0.032832	0.011864	2.77	0.0059	2.88244
PF	1	0.120817	0.011283	10.71	<.0001	3.96652
FTM	1	-0.017465	0.004697	-3.72	0.0002	1.87092
Efficiency	1	0.028046	0.002558	10.96	<.0001	1.11941
POS_SF	1	3.039488	1.084356	2.80	0.0053	1.02080

The MEANS Procedure

Variable	N	Mean
mape_fit	423	73.830
mape_acc	104	107.247
mae_fit	432	6.231
mae_acc	107	6.875
mse_fit	432	59.365
mse_acc	107	68.723

CART was tested next to predict player wins as it can predict numeric outcome variables. Since this was a regression task, the model was built using residual sum of squares (RSS). Another advantage of CART is that it is not adversely affected by multicollinearity when building trees, allowing all predictors to be considered in the model. In the following CART model, the number of leaves before pruning were 261, and after were 25. The significant variables are GP, Efficiency, _3PM, Min, and Age, in descending order. MAE, and MSE were then computed, and results were fairly similar to the multiple regression results from above. However, The difference between training and test metrics is not extreme, indicating that the model is generalizing well and not overfitting.

Modeling Round 1 - CART Model

Model Information	
Split Criterion Used	Variance
Pruning Method	Cost-Complexity
Subtree Evaluation Criterion	Cost-Complexity
Number of Branches	2
Maximum Tree Depth Requested	10
Maximum Tree Depth Achieved	10
Tree Depth	8
Number of Leaves Before Pruning	261
Number of Leaves After Pruning	25

Number of Observations Read	539
Number of Observations Used	539
Number of Training Observations Used	432
Number of Validation Observations Used	107

Variable Importance

Variable	Variable Label	Training		Validation		Relative Ratio	Count
		Relative	Importance	Relative	Importance		
GP	GP	1.0000	263.6	1.0000	135.6	1.0000	9
Efficiency	Efficiency	0.4028	106.2	0.3293	44.6541	0.8175	7
_3PM	_3PM	0.0491	12.9437	0.0704	9.5493	1.4344	1
Min	Min	0.0433	11.4109	0.0379	5.1338	0.8747	1
Age	Age	0.0457	12.0420	0.0241	3.2691	0.5278	1
STL	STL	0.0754	19.8873	0.0000	0	0.0000	1
FGP	FGP	0.0734	19.3451	0.0000	0	0.0000	1
FTA	FTA	0.0544	14.3381	0.0000	0	0.0000	1
OREB	OREB	0.0507	13.3698	0.0000	0	0.0000	1
AST	AST	0.0482	12.7109	0.0000	0	0.0000	1

The MEANS Procedure

Variable	N	Mean
mape_fit	423	21.8001631
mape_acc	104	29.3792970
mae_fit	432	3.3295582
mae_acc	107	3.7757293
mse_fit	432	18.3806581
mse_acc	107	23.9465153

Next, two models for neural network were created with all the predictors, which can be seen below. One model only had one hidden layer and 26 neurons (equal to the number of predictors), and the other model had two hidden layers and 52 neurons in total, each layer containing 26 neurons. Overall, both models performed almost exactly the same and showed no signs of overfitting in them. In both training and test sets, MAE performed very similarly in both the models between one hidden layer (train = 3.363, test = 2.954) and two hidden layers (train = 3.335, test = 2.963). This also follows for MSE between the model with one hidden layer (train = 20.111, test = 15.865) and two hidden layers (train = 19.941, test = 16.150). With this, it is noticeable that the test sets had a slightly better MSE than training. However, the difference is small and not a major concern. For the simplest model, we chose to go with the neural network model with only one hidden layer and 26 neurons.

Modeling Round 1 - Neural Network

1 hidden layer
26 neurons

Model Information	
Data Source	WORK.NBA_PART
Architecture	MLP
Number of Input Variables	26
Number of Hidden Layers	1
Number of Hidden Neurons	26
Number of Target Variables	1
Number of Weights	885
Optimization Technique	Limited Memory BFGS

Number of Observations Read	539
Number of Observations Used	539
Number Used for Training	432
Number Used for Validation	107

Train: Average Absolute Error	Valid: Average Absolute Error
3.362792	2.954259

The MEANS Procedure

Variable	N	Mean
mape_fit	423	22.998
mape_acc	104	26.507
mae_fit	432	3.363
mae_acc	107	2.954
mse_fit	432	20.111
mse_acc	107	15.865

2 hidden layers
52 neurons

Model Information	
Data Source	WORK.NBA_PART
Architecture	MLP
Number of Input Variables	26
Number of Hidden Layers	2
Number of Hidden Neurons	52
Number of Target Variables	1
Number of Weights	1587
Optimization Technique	Limited Memory BFGS

Number of Observations Read	539
Number of Observations Used	539
Number Used for Training	432
Number Used for Validation	107

Train: Average Absolute Error	Valid: Average Absolute Error
3.334653	2.962990

The MEANS Procedure

Variable	N	Mean
mape_fit	423	22.489
mape_acc	104	26.129
mae_fit	432	3.335
mae_acc	107	2.963
mse_fit	432	19.941
mse_acc	107	16.150

After building all three models, we saw that their performances were highly similar to each other. Looking at MAE in the training sets, they were all almost exactly the same (multiple linear regression = 3.432, CART = 3.33, neural network = 3.63). This also goes for the test sets where multiple linear regression = 2.998, CART = 3.776, neural network = 2.954). With CART, we noticed that both training and test sets performed almost the same, in terms of MAE.

With MSE, CART was also the only model where the training set performed better than the test set as they were 18.381 and 23.947, respectively. Under the multiple linear regression and neural network models, it was the other way around where MSE was better in the test set rather than training. Thus, both models would be good to use for predicting player wins. However, in the end, we agreed that linear regression would be the best model fit as it is the most parsimonious. Furthermore, the most significant predictors for player wins based on this model were GP, FTA, and Efficiency.

Modeling Round 1 - Evaluation

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.451
mape_acc	104	26.105
mae_fit	432	3.432
mae_acc	107	2.998
mse_fit	432	21.075
mse_acc	107	16.666

LINEAR REG

The MEANS Procedure

Variable	N	Mean
mape_fit	423	21.8001631
mape_acc	104	29.3792970
mae_fit	432	3.3295582
mae_acc	107	3.7757293
mse_fit	432	18.3806581
mse_acc	107	23.9465153

CART

The MEANS Procedure

Variable	N	Mean
mape_fit	423	22.998
mape_acc	104	26.507
mae_fit	432	3.363
mae_acc	107	2.954
mse_fit	432	20.111
mse_acc	107	15.865

NEURAL NETWORK

Modeling Round 2

For the second round of modeling, we continued with multiple regression, utilizing the stepwise selection method identified as the most parsimonious in the first round. The stepwise selection was chosen because of its ability to balance model complexity and predictive power by iteratively including and excluding variables based on their statistical significance. The key goal for this round was to refine the model further and validate its performance using additional evaluation metrics. The predictors for this second round were selected based on their significance in the first round of modeling and what we thought would be important in predicting player wins.

Predictors in second model
AGE
GP
FGM
_3PM
FTM
OREB
DREB
AST
STL
BLK
EFFICIENCY

The final stepwise model selected significant predictors to be GP (Games Played), FTM (Free Throws Made), and Efficiency. The analysis of variance indicated that the model explained a substantial portion of the variability in player wins, with an R-Square value of 0.89924 and an

adjusted R-Square of 0.89854. These metrics show that approximately 90% of the variability in wins can be explained by the model, reaffirming its predictive strength. The values for MAE and MSE in the training and validation sets were consistent, indicating no signs of overfitting. Additionally, variance inflation factors (VIF) for the predictors were all below the threshold of 10, confirming that multicollinearity was not an issue. While MAPE was included for comparison, inconsistencies in the number of observations (similar to those in round one) led us to rely on MAE and MSE for evaluation primarily. Despite these discrepancies, the second-round model improved validation error metrics, demonstrating its effectiveness in predicting player wins.

The CART model was revisited in Round 2 to predict player wins, using the residual sum of squares (RSS) as the split criterion for this regression task. CART's ability to handle numeric outcome variables and its resistance to multicollinearity allow all predictors to be evaluated during the model-building process. The model achieved a maximum tree depth of 10, with 271 leaves before pruning and 29 leaves after pruning, using the cost-complexity pruning method to optimize the tree's structure. The reduced tree depth (8) helped simplify the model while retaining its predictive accuracy. The significant variables in this model included GP (Games Played), Efficiency, and FTM (Free Throws Made), which are consistent with other methods, highlighting their strong relationship with the outcome variable. Evaluation metrics, including MAE, MSE, and MAPE, were calculated for both training and validation datasets. The model achieved an MAE of 3.14 (Train) and 3.77 (Validation), along with an MSE of 16.73 (Train) and 23.28 (Validation). These metrics were comparable to the multiple regression model and indicated the CART model's effectiveness. The relatively small difference between training and validation error metrics suggests the model is generalizing well without overfitting. Overall, the CART model provided an interpretable framework for understanding player wins while maintaining competitive predictive performance.

Finally, we developed a neural network with one hidden layer and 11 neurons. The model performed well, achieving an MAE of 3.38 (Training) and 2.87 (Validation), along with an MSE of 20.23 (Training) and 15.47 (Validation). The slightly better performance on the validation set suggests no signs of overfitting or underfitting. MAPE values were also reported (Training = 23.06, Validation = 25.09), though MAE and MSE remained the primary evaluation metrics for consistency. The low validation error highlights the model’s predictive accuracy for player wins. Compared to CART and regression, the neural network demonstrated competitive performance while maintaining simplicity with its single hidden layer.

Modeling Round 2

Multiple Regression - Stepwise Selection

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	81362	27121	1273.29	<.0001
Error	428	9116.20884	21.29955		
Corrected Total	431	90478			

Root MSE	4.61514
R-Square	0.89924
Adj R-Sq	0.89854
AIC	1759.33376
AICC	1759.47461
SBC	1341.60747
ASE (Train)	21.10234
ASE (Validate)	16.70372

Parameter Estimates						
Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	-0.527627	0.510579	-1.03	0.3020	0
GP	1	0.528320	0.010787	48.98	<.0001	1.37549
FTM	1	-0.009776	0.002479	-3.94	<.0001	1.47703
Efficiency	1	0.033700	0.001499	22.48	<.0001	1.08932

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.466
mape_acc	104	26.155
mae_fit	432	3.440
mae_acc	107	3.002
mse_fit	432	21.102
mse_acc	107	16.704

CART Model - Used RSS

Model Information	
Split Criterion Used	Variance
Pruning Method	Cost-Complexity
Subtree Evaluation Criterion	Cost-Complexity
Number of Branches	2
Maximum Tree Depth Requested	10
Maximum Tree Depth Achieved	10
Tree Depth	8
Number of Leaves Before Pruning	271
Number of Leaves After Pruning	29

Number of Observations Read	539
Number of Observations Used	539
Number of Training Observations Used	432
Number of Validation Observations Used	107

The MEANS Procedure

Variable	N	Mean
mape_fit	423	21.1854773
mape_acc	104	29.3405080
mae_fit	432	3.1390508
mae_acc	107	3.7729383
mse_fit	432	16.7273192
mse_acc	107	23.2801754

Neural Network - 1 hidden layer 11 neurons

Model Information	
Data Source	WORK.NBA_PART
Architecture	MLP
Number of Input Variables	11
Number of Hidden Layers	1
Number of Hidden Neurons	11
Number of Target Variables	1
Number of Weights	144
Optimization Technique	Limited Memory BFGS

Number of Observations Read	539
Number of Observations Used	539
Number Used for Training	432
Number Used for Validation	107

Train: Average Absolute Error	Valid: Average Absolute Error
3.375923	2.869599

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.056
mape_acc	104	25.094
mae_fit	432	3.376
mae_acc	107	2.870
mse_fit	432	20.225
mse_acc	107	15.466

Modeling Round 2 - Evaluation

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.466
mape_acc	104	26.155
mae_fit	432	3.440
mae_acc	107	3.002
mse_fit	432	21.102
mse_acc	107	16.704

LINEAR REG

The MEANS Procedure

Variable	N	Mean
mape_fit	423	21.1854773
mape_acc	104	29.3405080
mae_fit	432	3.1390508
mae_acc	107	3.7729383
mse_fit	432	16.7273192
mse_acc	107	23.2801754

CART

The MEANS Procedure

Variable	N	Mean
mape_fit	423	23.056
mape_acc	104	25.094
mae_fit	432	3.376
mae_acc	107	2.870
mse_fit	432	20.225
mse_acc	107	15.466

NEURAL NETWORK

CONCLUSION

When comparing the models from Round 1 to Round 2, several key points were different, including the changes in model variables and the error metrics.

For the multiple linear regression in model 1, all predictors were included in the model, and using the stepwise selection method gave the lowest error metrics on the validation set. (MAE=2.998 and MSE=16.66). In round two, the model had a reduction of variables to create a simpler model. Also, using the stepwise method we found that it still did well to predict wins, but the error measures slightly increased on the validation set when compared with round 1. (MAE=3.002, MSE=16.704)

The CART model was more complex as the number of leaves increased from 25 to 29 leaves after pruning in Round 2, and both went from 10 to 8 in tree depth. This was done to help

prevent overfitting and produce a more accurate model. The additional leaves after pruning made it more complex while still improving its ability to predict wins. The validation MAE (3.77) and MSE (23.28) were increases from the training set in round 2; however, the increases were not drastic enough to conclude an overfitting issue. Here the error measures were lower than in round 1. (MSE=23.95) and (MAE=3.78)

Similarly, the neural network model in Round 2 did well with lower validation error metrics (MAE = 2.87, MSE = 15.47) compared to training error metrics (MAE = 3.38, MSE = 20.23). The slightly better performance in error metrics could be attributed to the reduction of a single hidden layer and the number of neurons in round 1. It dropped from 26 neurons to 11, indicating that the model was simpler. The simpler model still predicted the number of wins in the validation set well. These error measures were only slightly lower when compared with the other validation error metrics in Round 1. (MAE=2.94, MSE=15.865).

Overall, both rounds of models performed comparably well. However, in the second round of models, significantly fewer variables were used to achieve similar results. Therefore, it would be beneficial to utilize the second round of models, as they are more parsimonious. Of the 3 models, the recommended model for prediction is the **multiple linear regression model**. This is the parsimonious model, as it is the simplest of the three and easiest to interpret. Despite the CART and neural network's ability to capture more diverse and complex relationships, the error measures indicate that they do not add significantly more accuracy. Variables like GP (Games Played), Efficiency, and FTM (Free Throws Made) were highlighted as significant predictors. These predictors are easy to track and could be metrics that coaches could use to scout future players or evaluate current players to trade for.

MODEL IMPROVEMENT

As mentioned above, the data was collected from the NBA 2023/2024 regular season. Over time, experts have observed that there have been significant swings regarding the impact certain statistics have on winning. The project could have been taken a step further if data from multiple seasons was utilized. This would have enabled the team to identify those shifts and when they occurred.

Similarly, another extra step the team could have taken would be collecting data from different levels of the sport - professional, college, and high school. As most parties involved in the sport know, certain statistics/qualities are valued more than others, depending on what level of the sport is in question. Including that in the models would have added an additional layer that would be interesting to explore.

Finally, a binary outcome variable could have been created. This would have been centered around the fact that it often takes players about 45 wins to reach the postseason. Creating a binary outcome variable to that effect and exploring the thresholds for each statistic to reach that mark would also provide valuable insight.

APPLICATION

The models above would be valuable to the following groups of people:

- Front Office/Ownership
    - ❖ Drafting rookies
    - ❖ Player contract extensions
    - ❖ Staff hires

- Coaching Staff
    - ❖ Player minute distribution
    - ❖ Player development
    - ❖ Personnel decisions

- Players
    - ❖ Improvement plans/focused training.

MODEL CODE:

```
proc import out=nba datafile="/home/u63401111/sasuser.v94/Final Project/NBA.xlsx"
dbms=xlsx replace; sheet="Data";
run;

/* Scatterplot of Players by Their Total Points */

proc sgplot data=nba;
	scatter x=age y= w;
	title "Scatterplot of Age by Total Games Won";
	xaxis label="Age";
	yaxis label="Total Games Won";

/* Bar Chart of Positions Count */

proc sgplot data=nba;
	vbar pos / categoryorder=respdesc;
	title "Bar Chart of Player Positions";
run;

/* Correlation Matrix */
proc corr data=nba;
	var gp age min fgm fga _3pm _3pa ftm fta oreb dreb ast tov stl blk pf fp dd2 td3
efficiency FGP _3PP FTP REB L;
run;

/* Boxplot */
proc sgplot data=nba;
  vbox w / category=pos;
  title "Box Plot of Points Scored by Basketball Position";
  xaxis label="Position";
  yaxis label="Games Won";
run;

/* Numerical Summaries*/

proc means data=nba n mean median min max maxdec=3;
	var W;
	class pos;
run;

proc means data=nba n mean median min max maxdec=3;
```

```
   var w;
   class team;
run;

proc means data=nba n mean median min max maxdec=3;
   var gp;
   class team;
run;

proc means data=nba n mean median min max maxdec=3;
   var gp;
   class pos;
run;


/* BUILDING MODELS */

/* Creating Dummy Variables */
data nba;
   set nba;
   if POS = "PG" then POS_PG = 1; else POS_PG = 0;
   if POS = "SG" then POS_SG = 1; else POS_SG = 0;
   if POS = "SF" then POS_SF = 1; else POS_SF = 0;
   if POS = "PF" then POS_PF = 1; else POS_PF = 0;
   if POS = "F" then POS_F = 1; else POS_F = 0;
   if POS = "G" then POS_G = 1; else POS_G = 0;
run;

/* Partition Data - 80/20 */
proc surveyselect data=nba samprate=.8 method=srs outall out=nba_part seed=12345;
run;

/* Linear Regression 1 - all predictors ---------------------------------------------------- */

proc hpreg data=nba_part;
   partition rolevar=selected(train='1' validate='0');
   model W = Age GP MIN PTS FGM FGA FGP _3PM _3PA _3PP FTM FTA FTP OREB DREB REB
AST TOV STL BLK PF FP DD2 TD3 EFFICIENCY POS_PG POS_SG POS_SF POS_PF POS_F
POS_G/vif;
   selection method=stepwise;
   output out=linear_pred1 p=W_predict r=W_resid copyvar=(W selected);
run;
```

```
/* Evaluate Linear Regression Model Performance */
data linear_pred1;
   set linear_pred1;
   if selected = 1 then do;
      mape_fit = (abs(W_resid) / W) * 100;
      mae_fit = abs(W_resid);
      mse_fit = W_resid**2;
   end;
   else if selected = 0 then do;
      mape_acc = (abs(W_resid) / W) * 100;
      mae_acc = abs(W_resid);
      mse_acc = W_resid**2;
   end;
run;

/* Performance Metrics */
proc means data=linear_pred1 n mean maxdec=3;
   var mape_fit mape_acc mae_fit mae_acc mse_fit mse_acc;
   title "Performance Metrics: MAPE, MAE, and MSE for Player Wins (W)";
run;

/* Linear Regression 2 */

proc hpreg data=nba_part;
   partition rolevar=selected(train='1' validate='0');
   model W = AGE GP FGM _3PM FTM OREB DREB AST STL BLK EFFICIENCY/vif;
   selection method=stepwise;
   output out=linear_pred2 p=W_predict r=W_resid copyvar=(W selected);
run;

/* Evaluate Linear Regression Model Performance */
data linear_pred2;
   set linear_pred2;
   if selected = 1 then do;
      mape_fit = (abs(W_resid) / W) * 100;
      mae_fit = abs(W_resid);
      mse_fit = W_resid**2;
   end;
   else if selected = 0 then do;
      mape_acc = (abs(W_resid) / W) * 100;
      mae_acc = abs(W_resid);
```

```
      mse_acc = W_resid**2;
   end;
run;

/* Performance Metrics */
proc means data=linear_pred2 n mean maxdec=3;
   var mape_fit mape_acc mae_fit mae_acc mse_fit mse_acc;
   title "Performance Metrics: MAPE, MAE, and MSE for Player Wins (W)";
run;

/* CART Model 1 ---------------------------------------------------- -*/

proc hpsplit data=nba_part nodes=detail;
        partition rolevar=selected(train='1' validate='0');
        class POS;
        model W = POS Age GP MIN PTS FGM FGA FGP _3PM _3PA _3PP FTM FTA FTP OREB
DREB REB AST TOV STL BLK PF FP DD2 TD3 EFFICIENCY;
        grow rss;
        prune cc;
        output out= nbaout_cart1;
        id selected;
run;

/* CART Model - Error Measures */
data nbaout_cart1;
        set nbaout_cart1;
        if selected=1 then
                do;
                        mape_fit=abs((W-P_W)/W)*100;
                        mae_fit=abs(W-P_W);
                        mse_fit=(W-P_W)**2;
                end;
        else if selected=0 then
                do;
                        mape_acc=abs((W-P_W)/W)*100;
                        mae_acc=abs(W-P_W);
                        mse_acc=(W-P_W)**2;
                end;
run;

/* CART Model 2 */
```

```
proc hpsplit data=nba_part nodes=detail;
	partition rolevar=selected(train='1' validate='0');
	model W = AGE GP FGM _3PM FTM OREB DREB AST STL BLK EFFICIENCY;
	grow rss;
	prune cc;
	output out= nbaout_cart2;
	id selected;
run;

/* CART Model 2 - Error Measures */
data nbaout_cart2;
	set nbaout_cart2;
	if selected=1 then
		do;
			mape_fit=abs((W-P_W)/W)*100;
			mae_fit=abs(W-P_W);
			mse_fit=(W-P_W)**2;
		end;
	else if selected=0 then
		do;
			mape_acc=abs((W-P_W)/W)*100;
			mae_acc=abs(W-P_W);
			mse_acc=(W-P_W)**2;
		end;
run;

proc means data=nbaout_cart2 n mean;
	var mape_fit mape_acc mae_fit mae_acc mse_fit mse_acc;
run;

/* Neural Network Model 1 – all predictors ---------------------------------------- */

proc hpneural data=nba_part;
	partition rolevar=selected(train=1);
	target W/level=int;
	input AGE GP FGM _3PM FTM OREB DREB AST STL BLK EFFICIENCY/level=int;
	hidden 11;
	train numtries=10 maxiter=1000;
	id w selected;
	score out=nbaoutneural1;
run;
```

```
data nbaoutneural1;
	set nbaoutneural1;
	if selected=1 then
		do;
			mape_fit=(abs(w-p_w)/w)*100;
			mae_fit=abs(w-p_w);
			mse_fit=(w-p_w)**2;
		end;
	else if selected=0 then
		do;
			mape_acc=(abs(w-p_w)/w)*100;
			mae_acc=abs(w-p_w);
			mse_acc=(w-p_w)**2;
		end;
run;

proc means data=nbaoutneural1 n mean maxdec=2;
	var mape_fit mape_acc mae_fit mae_acc mse_fit mse_acc;
run;

/* Neural Network Model 2 */

proc hpneural data=nba_part;
	partition rolevar=selected(train=1);
	target w/level=int;
	input AGE GP FGM _3PM FTM OREB DREB AST STL BLK EFFICIENCY/level=int;
	hidden 11;
	train numtries=10 maxiter=1000;
	id w selected;
	score out=nbaoutneural2;
run;


data nbaoutneural2;
	set nbaoutneural2;
	if selected=1 then
		do;
			mape_fit=(abs(w-p_w)/w)*100;
			mae_fit=abs(w-p_w);
			mse_fit=(w-p_w)**2;
		end;
	else if selected=0 then
```

```
                do;
                        mape_acc=(abs(w-p_w)/w)*100;
                        mae_acc=abs(w-p_w);
                        mse_acc=(w-p_w)**2;
                end;
run;

proc means data=nbaoutneural2 n mean maxdec=2;
        var mape_fit mape_acc mae_fit mae_acc mse_fit mse_acc;
run;
```