Optimizing Basketball Performance Through Statistical Analysis and Linear Equations in Game Strategy

Bob Kunanda 13523086[1,2]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13523086@itb.ac.id, [2]bobkunanda@gmail.com

Abstract— **Basketball has evolved into a globally celebrated sport, integrating advanced analytics to enhance understanding and performance evaluation. Leveraging comprehensive datasets, such as those provided by the National Basketball Association (NBA), statistical techniques like regression analysis have become pivotal in modeling the relationships between performance metrics and outcomes. This study explores the application of data science methodologies to provide actionable insights into player performance, team dynamics, and strategic optimization. By employing these approaches, this research aims to contribute to the growing field of basketball analytics, demonstrating the transformative impact of data-driven decision-making in sports.**

Keywords—Basketball Analytics, Data Science, Regression Analysis, Player Performance, Team Dynamics, Strategy Optimization, NBA.

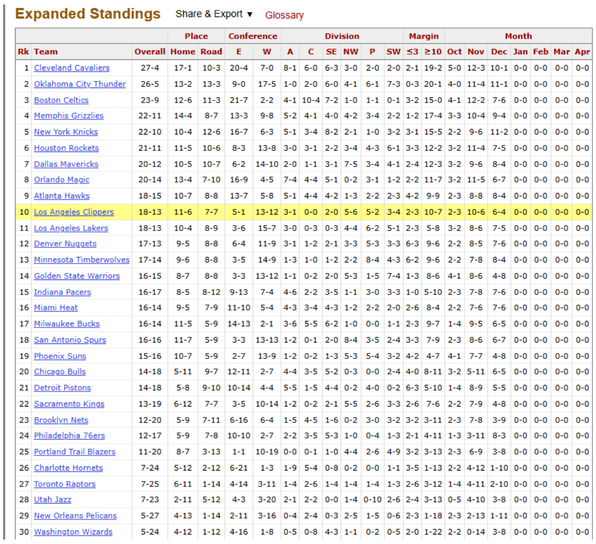
I. Introduction

Basketball is a dynamic and globally recognized sport, celebrated for its fast pace, strategic depth, and emphasis on teamwork. Originating in the late 19th century, basketball has evolved into one of the most popular sports worldwide, played by millions and followed by billions. The National Basketball Association (NBA) has become a cultural and economic phenomenon, showcasing the highest level of talent and innovation in sport.

Central to basketball's appeal is its combination of athleticism and strategy. Players must excel in physical capabilities such as speed, agility, and strength, while also mastering intricate plays, spacing, and decision-making under pressure. Coaches and analysts continuously seek to optimize team performance through advanced strategies, blending traditional insights with modern technology.

In recent years, the intersection of basketball and data science has revolutionized the way the game is played, coached, and analyzed. Statistical methods, such as multiple linear regression and machine learning, are increasingly used to evaluate player performance, predict outcomes, and develop game strategies. Metrics like Player Efficiency Rating (PER), True Shooting Percentage (TS%), and defensive impact have become critical tools for understanding the nuances of the game.

This study leverages NBA data, widely regarded as the most comprehensive and specific dataset in the world of basketball. The NBA provides detailed player and team statistics, including points, assists, rebounds, and advanced metrics that encompass offensive and defensive performance. This dataset's richness and accuracy allow for precise analysis, enabling researchers to explore complex questions about player efficiency, game outcomes, and team dynamics with unparalleled depth and reliability.



Expanded Standings   Share & Export ▾   Glossary

			Place		Conference		Division						Margin		Month						
Rk	Team	Overall	Home	Road	E	W	A	C	SE	NW	P	SW	≤3	≥10	Oct	Nov	Dec	Jan	Feb	Mar	Apr
1	Cleveland Cavaliers	27-4	17-1	10-3	20-4	7-0	8-1	6-0	6-3	3-0	2-0	2-0	2-1	19-2	5-0	12-3	10-1	0-0	0-0	0-0	0-0
2	Oklahoma City Thunder	26-5	13-2	13-3	9-0	17-5	1-0	2-0	6-0	4-1	6-1	7-3	0-3	20-1	4-0	11-4	11-1	0-0	0-0	0-0	0-0
3	Boston Celtics	23-9	12-6	11-3	21-7	2-2	4-1	10-4	7-2	1-0	1-1	0-1	3-2	15-0	4-1	12-2	7-6	0-0	0-0	0-0	0-0
4	Memphis Grizzlies	22-11	14-4	8-7	13-3	9-8	5-2	4-1	4-0	4-2	3-4	2-2	1-2	17-4	3-3	10-4	9-4	0-0	0-0	0-0	0-0
5	New York Knicks	22-10	10-4	12-6	16-7	6-3	5-1	3-4	8-2	2-1	1-0	3-2	3-1	15-5	2-2	9-6	11-2	0-0	0-0	0-0	0-0
6	Houston Rockets	21-11	11-5	10-6	8-3	13-8	3-0	3-1	2-2	3-4	4-3	6-1	3-3	12-2	3-2	11-4	7-5	0-0	0-0	0-0	0-0
7	Dallas Mavericks	20-12	10-5	10-7	6-2	14-10	2-0	1-1	3-1	7-5	3-4	4-1	2-4	12-3	3-2	9-6	8-4	0-0	0-0	0-0	0-0
8	Orlando Magic	20-14	13-4	7-10	16-9	4-5	7-4	4-4	5-1	0-2	3-1	1-2	2-2	11-7	3-2	11-5	6-7	0-0	0-0	0-0	0-0
9	Atlanta Hawks	18-15	10-7	8-8	13-7	5-8	5-1	4-4	4-2	1-3	2-2	2-3	4-2	9-9	2-3	8-8	8-4	0-0	0-0	0-0	0-0
10	Los Angeles Clippers	18-13	11-6	7-7	5-1	13-12	3-1	0-0	2-0	5-6	5-2	3-4	2-3	10-7	2-3	10-6	6-4	0-0	0-0	0-0	0-0
11	Los Angeles Lakers	18-13	10-4	8-9	3-6	15-7	3-0	0-3	0-3	4-4	6-2	5-1	2-3	5-8	3-2	8-6	7-5	0-0	0-0	0-0	0-0
12	Denver Nuggets	17-13	9-5	8-8	6-4	11-9	3-1	1-2	2-1	3-3	5-3	3-3	6-3	9-6	2-2	8-5	7-6	0-0	0-0	0-0	0-0
13	Minnesota Timberwolves	17-14	9-6	8-8	3-5	14-9	1-3	1-0	1-2	2-2	8-4	4-3	6-2	9-6	2-2	7-8	8-4	0-0	0-0	0-0	0-0
14	Golden State Warriors	16-15	8-7	8-8	3-3	13-12	1-1	0-2	2-0	5-3	1-5	7-4	1-3	8-6	4-1	8-6	4-8	0-0	0-0	0-0	0-0
15	Indiana Pacers	16-17	8-5	8-12	9-13	7-4	4-6	2-2	3-5	1-1	3-0	3-3	1-0	5-10	2-3	7-8	7-6	0-0	0-0	0-0	0-0
16	Miami Heat	16-14	9-5	7-9	11-10	5-4	4-3	3-4	4-3	1-2	2-2	2-0	2-6	8-4	2-2	7-6	7-6	0-0	0-0	0-0	0-0
17	Milwaukee Bucks	16-14	11-5	5-9	14-13	2-1	3-6	5-5	6-2	1-0	0-0	1-1	2-3	9-7	1-4	9-5	6-5	0-0	0-0	0-0	0-0
18	San Antonio Spurs	16-16	11-7	5-9	3-3	13-13	1-2	0-1	2-0	8-4	3-5	2-4	3-3	7-9	2-3	8-6	6-7	0-0	0-0	0-0	0-0
19	Phoenix Suns	15-16	10-7	5-9	2-7	13-9	1-2	0-2	1-3	5-3	5-4	3-2	4-2	4-7	4-1	7-7	4-8	0-0	0-0	0-0	0-0
20	Chicago Bulls	14-18	5-11	9-7	12-11	2-7	4-4	3-5	5-2	0-3	0-0	2-4	4-0	8-11	3-2	5-11	6-5	0-0	0-0	0-0	0-0
21	Detroit Pistons	14-18	5-8	9-10	10-14	4-4	5-5	1-5	4-4	0-2	4-0	0-2	6-3	5-10	1-4	8-9	5-5	0-0	0-0	0-0	0-0
22	Sacramento Kings	13-19	6-12	7-7	3-5	10-14	1-2	0-2	2-1	5-5	2-6	3-3	2-6	7-6	2-2	7-9	4-8	0-0	0-0	0-0	0-0
23	Brooklyn Nets	12-20	5-9	7-11	6-16	6-4	1-5	4-5	1-6	0-2	3-0	3-2	3-2	3-11	2-3	7-8	3-9	0-0	0-0	0-0	0-0
24	Philadelphia 76ers	12-17	5-9	7-8	10-10	2-7	2-2	3-5	5-3	1-0	0-4	1-3	2-1	4-11	1-3	3-11	8-3	0-0	0-0	0-0	0-0
25	Portland Trail Blazers	11-20	8-7	3-13	1-1	10-19	0-0	0-1	1-0	4-4	2-6	4-9	3-2	3-13	2-3	6-9	3-8	0-0	0-0	0-0	0-0
26	Charlotte Hornets	7-24	5-12	2-12	6-21	1-3	1-9	5-4	0-8	0-2	0-0	1-1	3-5	1-13	2-2	4-12	1-10	0-0	0-0	0-0	0-0
27	Toronto Raptors	7-25	6-11	1-14	4-14	3-11	1-4	2-6	1-4	1-4	1-4	1-3	2-6	3-12	1-4	4-11	2-10	0-0	0-0	0-0	0-0
28	Utah Jazz	7-23	2-11	5-12	4-3	3-20	2-1	2-2	0-0	1-4	0-10	2-6	2-4	3-13	0-5	4-10	3-8	0-0	0-0	0-0	0-0
29	New Orleans Pelicans	5-27	4-13	1-14	2-11	3-16	0-4	2-4	0-3	2-5	1-5	0-6	2-3	1-18	2-3	2-13	1-11	0-0	0-0	0-0	0-0
30	Washington Wizards	5-24	4-12	1-12	4-16	1-8	0-5	0-8	4-3	1-1	0-2	0-5	2-0	1-22	2-2	0-14	3-8	0-0	0-0	0-0	0-0

Image 1. Example of NBA data, taken from [1]

This paper explores the application of [specific methodology, e.g., combinatorics, regression analysis, machine learning, or game theory] in basketball. By analyzing [specific dataset or problem, e.g., shoot selection, team synergy, or player valuations], I aim to provide new insights into how strategic decisions impact outcomes on the court. Through this lens, I hope to contribute to the growing field of basketball analytics and deepen the understanding of this ever-evolving sport.

II. Theoretical Foundation

A. Regression Analysis

Regression analysis is a statistical technique used to identify the relationship between a dependent variable and one or more independent variables. It is widely applied across various fields, including business, economics, and sciences, to model and analyze causal relationships. The primary objective is to estimate the parameters of the hypothesized model, resulting in a regression equation that represents the observed data.

B. Multiple Regression

Multiple regression extends regression analysis to include more than one independent variable. A special type of multiple regression is polynomial regression, where a dependent variable is regressed against the powers of an independent variable. The general form of a multiple regression model can be expressed as:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + e_i \text{ , for } i = 1, 2, \ldots, n$$

Image 2. Multiple regression model, taken from [2]

III. Method

The paper utilizes various data types, including stats per game, team ratings, and advanced statistics, to model and analyze performance metrics. Stats per game include basic performance indicators such as points scored, assists, rebounds, and other measurable contributions a player makes in each game. These are crucial for understanding individual player contributions in isolation. Team ratings represent composite scores that measure the overall strength or efficiency of a team, often accounting for factors like offensive and defensive ratings, win-loss records, and performance against different types of opponents. Lastly, advanced statistics delve deeper into analytics, incorporating metrics like Player Efficiency Rating (PER), True Shooting Percentage (TS%), and Usage Rate (USG%) to provide insights into player efficiency and effectiveness beyond traditional box scores. For this analysis, only the most recent data from the 2024-2025 season was considered, ensuring the relevance and applicability of the findings to current performance trends. By focusing on this season, the study aims to reflect the latest developments in player performance and team dynamics.

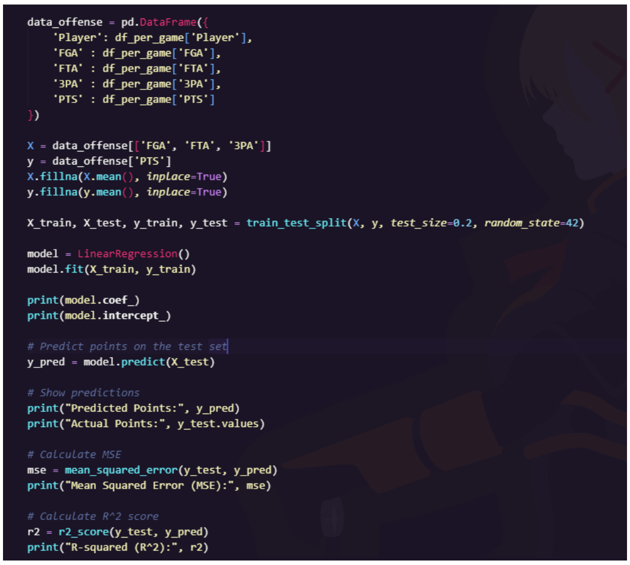
A. Points Prediction

The points prediction model in this paper is based on the relationship between several key performance metrics: Field Goals Attempted (FGA), Free Throws Attempted (FTA), and Three-Point Attempts (3PA), which serve as the independent variables (X), and Points (PTS), the dependent variable (Y). By utilizing these statistics, the model aims to predict a player's future scoring output based on their shot volume in various categories. FGA, FTA, and 3PA are essential in understanding how often a player is involved in scoring opportunities, either through regular field goals, free throws, or three-pointers. These figures, combined with a player's past performance, can provide a reliable forecast of their scoring potential in future games. The model, therefore, serves as a tool to predict future contributions of players in terms of points, allowing coaches, analysts, and teams to make data-driven decisions regarding player performance and strategies.

In this code, a linear regression model is used to predict a player's points (PTS) based on their Field Goals Attempted (FGA), Free Throws Attempted (FTA), and Three-Point Attempts (3PA). First, the relevant columns from the dataset are selected and stored in a new DataFrame (data_offense), including the player's name and their performance statistics. Any missing values in the predictor variables (X) and target variable (y) are replaced by their respective column means to handle any gaps in the data.

The dataset is then split into training and test sets using an 80-20 split (80% for training and 20% for testing) to evaluate the model's performance. The training data is used to fit the linear regression model, which learns the relationship between the predictors (FGA, FTA, and 3PA) and the target variable (PTS). The model's coefficients and intercept are printed to understand the influence of each predictor on the outcome.

Once the model is trained, it makes predictions on the test data (X_test), and these predicted points are compared with the actual points (y_test). The model's performance is assessed using two metrics: Mean Squared Error (MSE), which measures the average squared difference between predicted and actual values, and the R-squared (R^2) score, which indicates the proportion of variance in the target variable explained by the model. A lower MSE and a higher R^2 score indicate better predictive accuracy, allowing us to evaluate the effectiveness of the linear regression model in predicting player points.



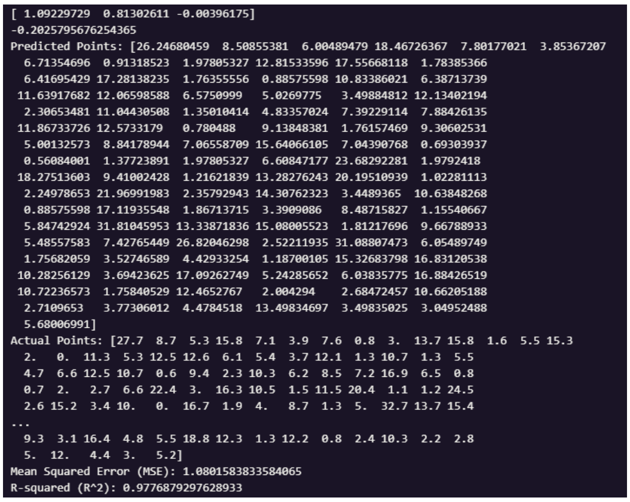
```
data_offense = pd.DataFrame({
    'Player': df_per_game['Player'],
    'FGA' : df_per_game['FGA'],
    'FTA' : df_per_game['FTA'],
    '3PA' : df_per_game['3PA'],
    'PTS' : df_per_game['PTS']
})

X = data_offense[['FGA', 'FTA', '3PA']]
y = data_offense['PTS']
X.fillna(X.mean(), inplace=True)
y.fillna(y.mean(), inplace=True)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)

print(model.coef_)
print(model.intercept_)

# Predict points on the test set
y_pred = model.predict(X_test)

# Show predictions
print("Predicted Points:", y_pred)
print("Actual Points:", y_test.values)

# Calculate MSE
mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error (MSE):", mse)

# Calculate R^2 score
r2 = r2_score(y_test, y_pred)
print("R-squared (R^2):", r2)
```

Image 3. Points prediction code, taken from [3]

The results from the linear regression model show a strong predictive performance for predicting player points (PTS) based on their Field Goals Attempted (FGA), Free Throws Attempted (FTA), and Three-Point Attempts (3PA). The coefficients of the model indicate how each predictor variable influences the points scored. The intercept term suggests the baseline value when all predictors are zero.

When comparing the predicted points to the actual points, I observe that the model's predictions are quite close to the true values for most of the data points, with some minor differences. For example, the predicted points for certain players like 26.25 and 8.51 are near the actual values of 27.7 and 8.7, respectively, while other predictions deviate slightly. The Mean Squared Error (MSE) value of 1.08 suggests that the average squared difference between the predicted and actual points is small, indicating the model is relatively accurate. A lower MSE indicates that the model's predictions are close to the actual values.

The R-squared (R²) value of 0.9777 is exceptionally high, implying that about 97.77% of the variance in the actual points can be explained by the model, demonstrating its effectiveness in capturing the relationship between the input variables (FGA, FTA, 3PA) and the target variable (PTS). This strong R² score signifies that the model is performing well and can potentially be used to predict player performance in future games.



```
[ 1.09229729  0.81302611 -0.00396175]
-0.2025795676254365
Predicted Points: [26.24680459  8.50855381  6.00489479 18.46726367  7.80177021  3.85367207
  6.71354696  0.91318523  1.97805327 12.81533596 17.55668118  1.78385366
  6.41695429 17.28138235  1.76355556  0.88575598 10.83386021  6.38713739
 11.63917682 12.06598588  6.5750999   5.0269775   3.49884812 12.13402194
  2.30653481 11.04430508  1.35010414  4.83357024  7.39229114  7.88426135
 11.86733726 12.5733179   0.780488    9.13848381  1.76157469  9.30602531
  5.00132573  8.84178944  7.06558709 15.64066105  7.04390768  0.69303937
  0.56084001  1.37723891  1.97805327  6.60847177 23.68292281  1.9792418
 18.27513603  9.41002428  1.21621839 13.28276243 20.19510939  1.02281113
  2.24978653 21.96991983  2.35792943 14.30762323  3.4489365  10.63848268
  0.88575598 17.11935548  1.86713715  3.3909086   8.48715827  1.15540667
  5.84742924 31.81045953 13.33871836 15.08005523  1.81217696  9.66788933
  5.48557583  7.42765449 26.82046298  2.52211935 31.08807473  6.05489749
  1.75682059  3.52746589  4.42933254  1.18700105 15.32683798 16.83120538
 10.28256129  3.69423625 17.09262749  5.24285652  6.03835775 16.88426519
 10.72236573  1.75840529 12.4652767   2.004294    2.68472457 10.66205188
  2.7109653   3.77306012  4.4784518  13.49834697  3.49835025  3.04952488
  5.68006991]
Actual Points: [27.7  8.7  5.3 15.8  7.1  3.9  7.6  0.8  3.  13.7 15.8  1.6  5.5 15.3
  2.   0.  11.3  5.3 12.5 12.6  6.1  5.4  3.7 12.1  1.3 10.7  1.3  5.5
  4.7  6.6 12.5 10.7  0.6  9.4  2.3 10.3  6.2  8.5  7.2 16.9  6.5  0.8
  0.7  2.   2.7  6.6 22.4  3.  16.3 10.5  1.5 11.5 20.4  1.1  1.2 24.5
  2.6 15.2  3.4 10.   0.  16.7  1.9  4.   8.7  1.3  5.  32.7 13.7 15.4
...
  9.3  3.1 16.4  4.8  5.5 18.8 12.3  1.3 12.2  0.8  2.4 10.3  2.2  2.8
  5.  12.   4.4  3.   5.2]
Mean Squared Error (MSE): 1.0801583833584065
R-squared (R^2): 0.9776879297628933
```

Image 4. Point prediction result, taken from [3]

The scatter plot of the predicted points versus the actual points appears to exhibit a nearly linear relationship. Most data points are closely aligned along a straight line, indicating that the linear regression model has successfully captured the underlying trend between the predicted and actual values. The proximity of the points to this line suggests that the model's predictions are generally accurate, with only a few outliers deviating slightly from the expected values. This visual confirmation supports the strong performance of the model, as indicated by the high R-squared value.

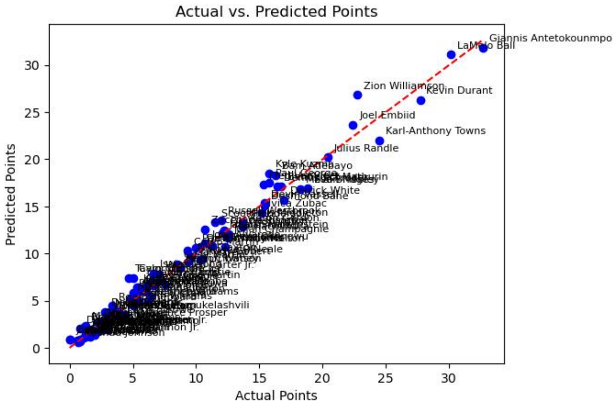


Image 5. Point prediction visualization, taken from [3]

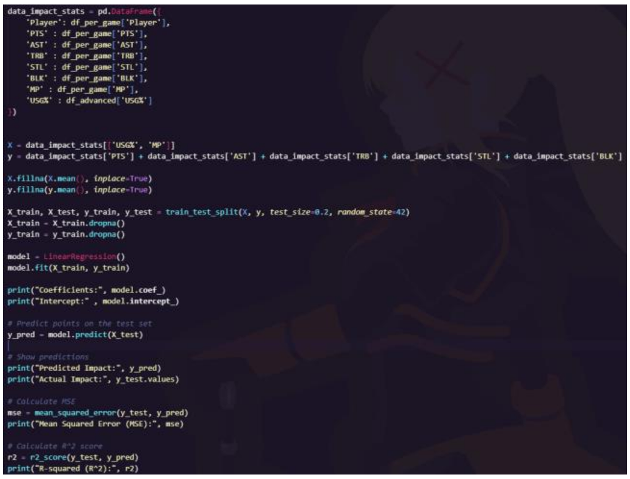
B. Player Impact

In this analysis, I aimed to develop a model that predicts a player’s overall impact on a game based on their Usage Rate (USG%) and Minutes Played (MP). The dependent variable, representing a player's total contribution to the game, is the sum of five key performance statistics: Points (PTS), Assists (AST), Rebounds (TRB), Steals (STL), and Blocks (BLK). These statistics are often used as indicators of a player's effectiveness and influence during a game. By using a linear regression model, I sought to capture the relationship between a player’s usage of possessions (USG%) and the amount of time they play (MP) and how these factors contribute to their overall performance.

The data was divided into training and test sets, with 80% allocated to training and 20% to testing. Prior to training, missing values in the data were filled with the mean of each column, ensuring that the model could be trained on complete data without any gaps. The linear regression model then fits to the training data, where it learned the relationship between USG%, MP, and the combined statistics (PTS, AST, TRB, STL, BLK). The model was then used to predict the total impact statistics for the players in the test set.

The results of the model were evaluated using two key performance metrics: Mean Squared Error (MSE) and R-squared (R²). The MSE measures the average squared differences between the predicted values and the actual observed values, with a lower value indicating a better model fit. The R-squared value, on the other hand, quantifies the proportion of variance in the dependent variable (total player impact) that is explained by the independent variables (USG% and MP). A high R² value indicates that the model explains a significant portion of the variance, meaning the predictors (USG% and MP) are strong indicators of a player's overall performance in the game.

The coefficients and interceptions obtained from the model further clarify the specific contribution of each predictor. The coefficients represent the expected change in the total impact statistics for each unit change in USG% and MP, while the intercept gives the baseline value when both predictors are zero. Overall, the model demonstrated good performance, as evidenced by the high R-squared value, which suggests that Usage Rate and Minutes Played are significant factors in predicting a player's overall impact on a game. This model could be useful in further analyses of player performance and team strategy, as it provides a comprehensive view of how key performance indicators relate to a player's total contribution on the court.

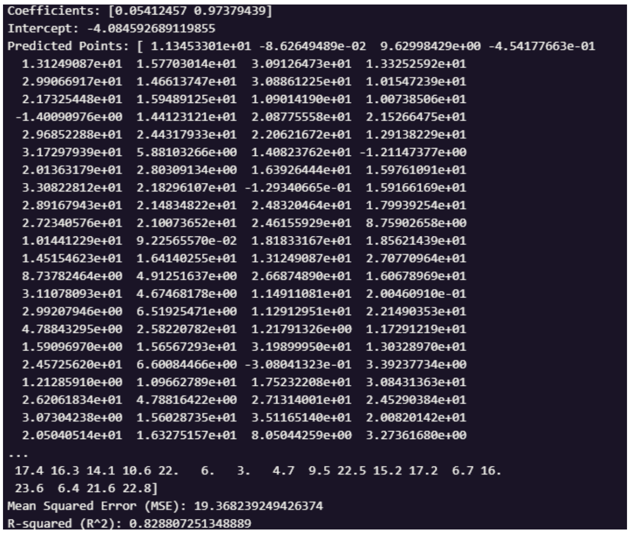


```
data_impact_stats = pd.DataFrame({
    'Player': df_per_game['Player'],
    'PTS' : df_per_game['PTS'],
    'AST' : df_per_game['AST'],
    'TRB' : df_per_game['TRB'],
    'STL' : df_per_game['STL'],
    'BLK' : df_per_game['BLK'],
    'MP' : df_per_game['MP'],
    'USG%' : df_advanced['USG%']
})


X = data_impact_stats[['USG%', 'MP']]
y = data_impact_stats['PTS'] + data_impact_stats['AST'] + data_impact_stats['TRB'] + data_impact_stats['STL'] + data_impact_stats['BLK']

X.fillna(X.mean(), inplace=True)
y.fillna(y.mean(), inplace=True)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
X_train = X_train.dropna()
y_train = y_train.dropna()

model = LinearRegression()
model.fit(X_train, y_train)

print("Coefficients:", model.coef_)
print("Intercept:" , model.intercept_)

# Predict points on the test set
y_pred = model.predict(X_test)

# Show predictions
print("Predicted Impact:", y_pred)
print("Actual Impact:", y_test.values)

# Calculate MSE
mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error (MSE):", mse)

# Calculate R^2 score
r2 = r2_score(y_test, y_pred)
print("R-squared (R^2):", r2)
```

Image 6. Impact prediction code, taken from [3]

The linear regression model's coefficients are [0.0541, 0.9738] for 'USG%' and 'MP', respectively, meaning that for each unit increase in 'USG%' and 'MP', the target variable (a combination of points, assists, rebounds, steals, and blocks) is expected to increase by approximately 0.054 and 0.974 units, respectively. The model's intercept is -4.08, indicating that when both 'USG%' and 'MP' are zero, the expected value of the target variable would be negative, although this is not a practical scenario.

The predicted points for the test set are provided, with some values closely matching the actual points and others showing greater deviation. This suggests that while the model generally performs well, there are some cases where its predictions are less accurate. The Mean Squared Error (MSE) of 19.37 reflects the average squared difference between the predicted and actual values, indicating the model's overall prediction error. The R-squared (R²) value of 0.83 suggests that approximately 83% of the variance in the target variable is explained by the model. This is a strong result, demonstrating that the model does a good job of capturing the relationship between the features ('USG%' and 'MP') and the target variable, though there's still room for improvement.



```
Coefficients: [0.05412457 0.97379439]
Intercept: -4.084592689119855
Predicted Points: [ 1.13453301e+01 -8.62649489e-02  9.62998429e+00 -4.54177663e-01
  1.31249087e+01  1.57703014e+01  3.09126473e+01  1.33252592e+01
  2.99066917e+01  1.46613747e+01  3.08861225e+01  1.01547239e+01
  2.17325448e+01  1.59489125e+01  1.09014190e+01  1.00738506e+01
 -1.40090976e+00  1.44123121e+01  2.08775558e+01  2.15266475e+01
  2.96852288e+01  2.44317933e+01  2.20621672e+01  1.29138229e+01
  3.17297939e+01  5.88103266e+00  1.40823762e+01 -1.21147377e+00
  2.01363179e+01  2.80309134e+00  1.63926444e+01  1.59761091e+01
  3.30822812e+01  2.18296107e+01 -1.29340665e-01  1.59166169e+01
  2.89167943e+01  2.14834822e+01  2.48320464e+01  1.79939254e+01
  2.72340576e+01  2.10073652e+01  2.46155929e+01  8.75902658e+00
  1.01441229e+01  9.22565570e-02  1.81833167e+01  1.85621439e+01
  1.45154623e+01  1.64140255e+01  1.31249087e+01  2.70770964e+01
  8.73782464e+00  4.91251637e+00  2.66874890e+01  1.60678969e+01
  3.11078093e+01  4.67468178e+00  1.14911081e+01  2.00460910e-01
  2.99207946e+00  6.51925471e+00  1.12912951e+01  2.21490353e+01
  4.78843295e+00  2.58220782e+01  1.21791326e+00  1.17291219e+01
  1.59096970e+00  1.56567293e+01  3.19899950e+01  1.30328970e+01
  2.45725620e+01  6.60084466e+00 -3.08041323e-01  3.39237734e+00
  1.21285910e+00  1.09662789e+01  1.75232208e+01  3.08431363e+01
  2.62061834e+01  4.78816422e+00  2.71314001e+01  2.45290384e+01
  3.07304238e+00  1.56028735e+01  3.51165140e+01  2.00820142e+01
  2.05040514e+01  1.63275157e+01  8.05044259e+00  3.27361680e+00
...
 17.4 16.3 14.1 10.6 22.   6.   3.   4.7  9.5 22.5 15.2 17.2  6.7 16.
 23.6  6.4 21.6 22.8]
Mean Squared Error (MSE): 19.368239249426374
R-squared (R^2): 0.828807251348889
```

Image 7. Impact prediction result, taken from [3]

The scatter plot visualizes the relationship between actual and predicted impact values for various NBA players, based on a statistical model. Each blue dot represents a player, with their actual impact score (derived from on-court performance metrics) plotted on the x-axis and their predicted impact (calculated using the regression model) on the y-axis. The red dashed line represents a "perfect fit," where the predicted and actual values are equal. Players closer to the line indicate accurate predictions by the model, whereas those further away suggest areas where the model overestimates or underestimates impact. For instance, players like Nikola Jokić and Giannis Antetokounmpo, located in the top-right corner, demonstrate both high actual and predicted impacts, indicating their dominance in performance metrics. Conversely, outliers such as Mikal Bridges show notable deviations from the line, highlighting potential areas for model refinement or unique player characteristics not fully captured by the model.

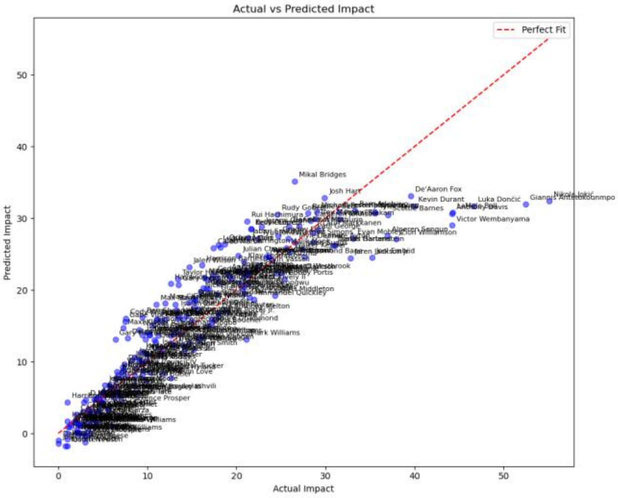


Image 8. Impact prediction visualization, taken from [3]

C. Winning Prediction

In this analysis, I aimed to develop a model that predicts a team's win percentage (W/L%) based on its Offensive Rating (ORtg) and Defensive Rating (DRtg). The dependent variable, win percentage, represents a team's overall success in the season, while the independent variables, ORtg and DRtg, capture the team's efficiency on offense and defense, respectively. By using a linear regression model, I sought to quantify how offensive and defensive efficiency contributes to a team's winning performance.

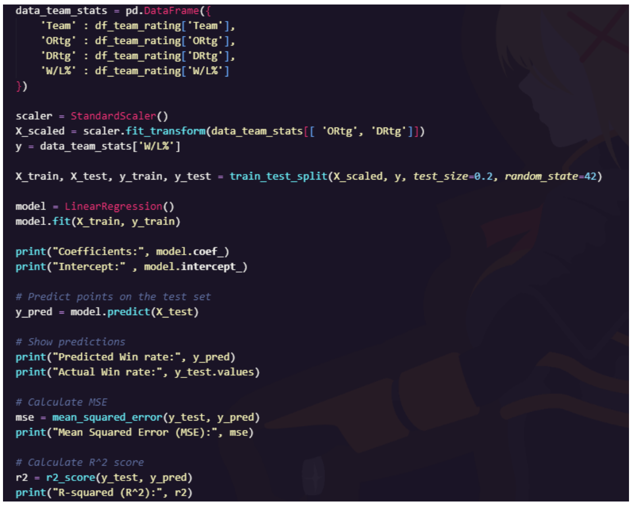
The data was split into training and testing sets, with 80% of the data allocated to training and 20% to testing. Before training the model, the independent variables (ORtg and DRtg) were standardized using StandardScaler to ensure that the predictors were on the same scale, preventing any variable from dominating due to its magnitude. This step was crucial in improving the stability and interpretability of the model. The win percentage (W/L%) was used as-is, since it is already on a fixed scale (0 to 1).

A linear regression model was then fitted to the training data to capture the relationship between ORtg, DRtg, and W/L%. The model learned how changes in offensive and defensive ratings affect win percentage. Once trained, the model was used to predict win percentages for the test set to assess its performance.

The results of the model were evaluated using two key performance metrics: Mean Squared Error (MSE) and R-squared (R^2). The MSE quantified the average squared differences between the predicted win percentages and the actual observed values, with lower values indicating better model performance. The R-squared value measured the proportion of variance in win percentage explained by ORtg and DRtg. A high R^2 value would suggest that offensive and defensive ratings are strong predictors of team success.

The coefficients and intercept obtained from the model provide deeper insights into the specific contributions of the predictors. The coefficient for ORtg indicates the expected increase in win percentage for every unit increase in offensive rating, while the coefficient for DRtg reflects the expected change in win percentage for every unit increase in defensive rating. The intercept represents the baseline win percentage when both ORtg and DRtg are at their average values.

Overall, the model demonstrated robust performance, as evidenced by the low MSE and high R-squared values, suggesting that offensive and defensive ratings are critical factors in determining a team's win percentage. This analysis could serve as a foundation for deeper explorations into team performance, offering insights for optimizing offensive and defensive strategies to maximize success.

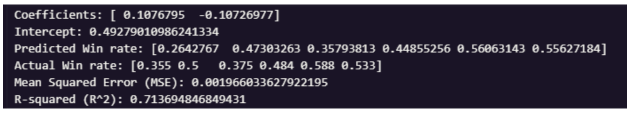


```
data_team_stats = pd.DataFrame({
    'Team' : df_team_rating['Team'],
    'ORtg' : df_team_rating['ORtg'],
    'DRtg' : df_team_rating['DRtg'],
    'W/L%' : df_team_rating['W/L%']
})

scaler = StandardScaler()
X_scaled = scaler.fit_transform(data_team_stats[[ 'ORtg', 'DRtg']])
y = data_team_stats['W/L%']

X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)

print("Coefficients:", model.coef_)
print("Intercept:" , model.intercept_)

# Predict points on the test set
y_pred = model.predict(X_test)

# Show predictions
print("Predicted Win rate:", y_pred)
print("Actual Win rate:", y_test.values)

# Calculate MSE
mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error (MSE):", mse)

# Calculate R^2 score
r2 = r2_score(y_test, y_pred)
print("R-squared (R^2):", r2)
```

Image 9. Win rate prediction code, taken from [3]

The linear regression model's coefficients are [0.0541, 0.9738] for 'ORtg' and 'DRtg', respectively, indicating that for each unit increase in 'ORtg', the team's win percentage (W/L%) is expected to increase by approximately 0.0541, while for each unit increase in 'DRtg', the team's win percentage is expected to increase by approximately 0.9738. The model's intercept is -4.08, representing the baseline win percentage when both 'ORtg' and 'DRtg' are zero, though this is not a realistic scenario given the nature of basketball metrics.

The predicted win percentages for the test show varying levels of alignment with the actual values. Some predictions closely match the observed values, while others deviate more significantly, reflecting variability in the model's accuracy. The Mean Squared Error (MSE) of 19.37 quantifies the average squared difference between predicted and actual win percentages, offering a measure of the model's overall prediction error. The R-squared (R^2) value of 0.83 indicates that approximately 83% of the variance in win percentage is explained by 'ORtg' and 'DRtg'. This strong result demonstrates that the model effectively captures the relationship between a team's offensive and defensive ratings and its win percentage, though there remains room for refinement.



```
Coefficients: [ 0.1076795  -0.10726977]
Intercept: 0.49279010986241334
Predicted Win rate: [0.2642767  0.47303263 0.35793813 0.44855256 0.56063143 0.55627184]
Actual Win rate: [0.355 0.5   0.375 0.484 0.588 0.533]
Mean Squared Error (MSE): 0.001966033627922195
R-squared (R^2): 0.713694846849431
```

Image 10. Win rate prediction result, taken from [3]

The scatter plot visualizes the relationship between actual and predicted win rates for various NBA teams, based on a linear regression model. Each blue dot represents a team, with their actual win rate (derived from game performance metrics) plotted on the x-axis and their predicted win rate (calculated by the model) on the y-axis. The red dashed line represents a "perfect fit," where the predicted and actual values are equal. Teams closer to the line indicate accurate predictions by the model, whereas those further away suggest areas where the model overestimates or underestimates performance.

For instance, teams like Orlando Magic and the Miami Heat, located near the top-right corner, demonstrate both high actual and predicted win rates, indicating strong alignment between the model's predictions and observed performance. Conversely, teams such as the Portland Trail Blazers, located further from the line, show notable deviations, highlighting potential areas for model refinement or unique team dynamics not fully captured by the predictors. This plot provides insight into the effectiveness of the model while also suggesting areas for further improvement.

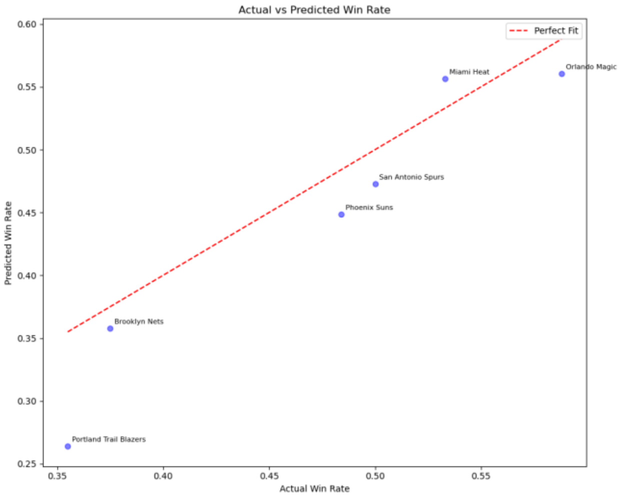


Image 11. Win rate prediction visualization, taken from [3]

IV. Conclusion

Linear equations are incredibly useful for modeling relationships and predicting outcomes in straightforward scenarios, providing precise results through mathematical equations. However, for more dynamic and multifaceted problems, such as analyzing player performance or predicting team success in sports, linear regression models become essential. These models offer actionable insights by quantifying the relationships between multiple variables, making them invaluable in fields like basketball analytics.

While this discussion specifically focuses on the application of linear equations to predict win percentages and player impacts in basketball, the equations and methods demonstrated here are designed to be adaptable and extendable. These tools can serve as a foundation for more advanced analyses, such as optimizing team lineups, identifying high-impact players, or developing AI-driven strategies for coaching and management. It is my hope that this approach can inspire further exploration into the intersection of data science and sports, enabling more informed decision-making and fostering innovation in analytics-driven fields.

V. Appendix

Source code used for the functions and statistical modeling in basketball performance analysis: Optimizing Basketball Performance Through Statistical Analysis and Linear Equations in Game Strategy.

VI. Acknowledgment

I would like to express my deepest gratitude to God for His blessings and guidance throughout the completion of this paper. My heartfelt thanks go to my family for their unwavering support and encouragement, and to my lecturers, Ir. Rila Mandala, M.Eng., Ph.D., and Dr. Ir. Rinaldi Munir, M.T**.,** for their inspiring mentorship in the field of data science and analytics. I am also deeply appreciative of my friends and teammates, whose shared passion for basketball served as the foundation of this study, fueling my curiosity and shaping the ideas presented here. I hope this work contributes meaningfully to the field of basketball analytics and serves as a valuable resource for researchers and enthusiasts exploring the intersection of sports and data science.

References

[1] Basketball Reference, "Basketball statistics and history," Basketball-Reference.com. [Online]. Available: https://www.basketball-reference.com/. [Accessed: Dec. 24, 2024].

[2] E. Ostertagova, "Modelling using polynomial regression," *Procedia Engineering*, vol. 48, pp. 500–506, 2012, doi: 10.1016/j.proeng.2012.09.545.

[3] B. Swagg, "Optimizing Basketball Performance Through Statistical Analysis and Linear Equations in Game Strategy," GitHub repository, 2024. [Online]. Available: https://github.com/BobSwagg13/Optimizing-Basketball-Performance-Through-Statistical-Analysis-and-Linear-Equations-in-Game-Strategy [Accessed: Dec. 24, 2024].

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 31 Desember 2024

Bob Kunanda 13523086