NBA Player Performance Prediction using Statistical Analysis and Machine Learning

Nacim M. Osman
nmo2002@memphis.edu
U00794074

Abstract

Predicting NBA player performance is a crucial yet complex task that involves numerous factors such as historical trends, player roles, and statistical variability. This project presents a streamlined prediction system combining interpretable Linear Regression with a more flexible Random Forest model to forecast key stats like points per game and shooting efficiency. I also introduce a graph-based clustering technique that uses cosine similarity and the Louvain algorithm to identify player archetypes and enable context-aware forecasting. The system processes data from the 2021–2025 NBA seasons and is evaluated using MAE and RMSE. Although injury effects and role shifts remain difficult to quantify, the framework establishes a strong foundation for future extensions involving dynamic networks and injury modeling.

Introduction

Player performance prediction in the NBA is a critical challenge for scouts, analysts, fantasy sports managers, and team executives. Accurate forecasting supports draft strategy, game-day decision-making, and salary contract evaluations. Several challenges complicate this task:

- **Injury unpredictability**: The outcomes are shaped by unstructured factors such as the lifestyle of the player, medical treatment, and prior history.
- **Volatile player roles**: Trades, coaching preferences, and strategic changes lead to changes in playtime and usage.
- **Sparse data** for rookies and younger players.
- **Correlated metrics**: Statistics such as usage rate can skew performance measures like efficiency.

This project constructs a robust statistical and machine learning pipeline to address these issues where possible, while acknowledging unresolved challenges like injury modeling and role variability. It combines interpretable models with graph-based clustering to offer extensible methods for future sports analytics.

Solution

1. Data Sources and Preprocessing

I used the NBA Stats API to pull player-season data from 2021–2025. The preprocessing steps include the following:

- Filtering players with fewer than 10 games per season
- Feature engineering: TS%, eFG%, 3PAR
- Creating delta features: $\Delta\text{PPG} = \text{PPG}_t - \text{PPG}_{t-1}$
- Imputing missing values with mean substitution
- Z-score normalization and categorical encoding
- Adding season indicators to model longitudinal trends

2. Model Architecture

a. Linear Regression – Baseline Model
I designated Linear Regression as our **baseline** due to its transparency and ease of interpretation. Each statistic is modeled as:

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n + \varepsilon$$

Where:

- Y: target value (e.g., next season's points per game)
- X_i: features (e.g., player stats, minutes, shooting splits)
- ε: error term

b. Random Forest Regression
To capture non-linear effects and interactions, I implemented Random Forest Regression:

$$\hat{y} = \frac{1}{B} \sum_{i=1}^{B} f_i(x)$$

Where:

- $\hat{y}$: predicted stat
- B: number of decision trees

- f_i: prediction from tree i
- x: input feature vector

Manual hyperparameter tuning was performed, but the default settings provided the best performance balance and were retained.

c. Confidence Interval Estimation

I compute the 95% confidence intervals using:

$$CI = \hat{y} \pm t_{\alpha/2,n-p} \cdot s \cdot \sqrt{1 + x^T (X^T X)^{-1} x}$$

Where:

- $\hat{y}$: prediction
- s: standard error of the model
- t: critical t-value
- x: feature vector
- X: design matrix of training set

3. Graph-Based Player Clustering

I built a graph where:

- Nodes = players
- Edges = cosine similarity between normalized stat vectors

Edges are drawn where similarity exceeds 0.75. I applied the Louvain algorithm to detect communities (clusters) representing:

- Elite scorers
- Role players
- 3PT specialists
- Defensive anchors

Figure 1 presents a force-directed graph of player similarities, where nodes closer together exhibit higher statistical similarity. This aids in visually interpreting archetype groupings.

We also perform KMeans clustering (k=5), as seen in **Figure 2**, to supplement this with centroid-based analysis and visual validation via PCA. The clusters identified meaningful groupings: the green cluster represents the lowest-performing tier of NBA players, dark blue indicates slightly stronger role players, light blue corresponds to mid-tier big men, purple captures mid-tier guards, and orange highlights the top-tier superstars.

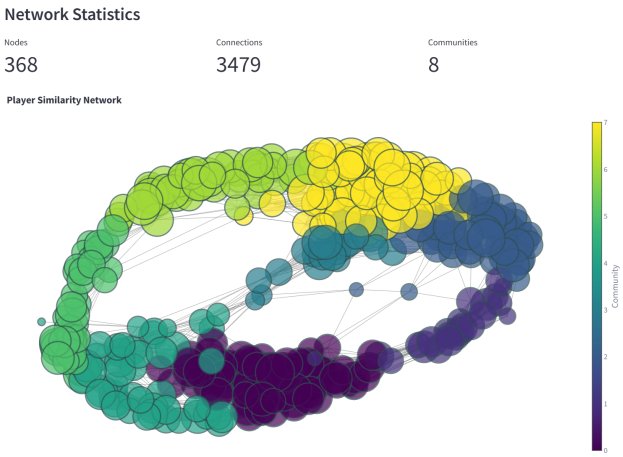

Figure 1: *Force-directed graph of player similarities*

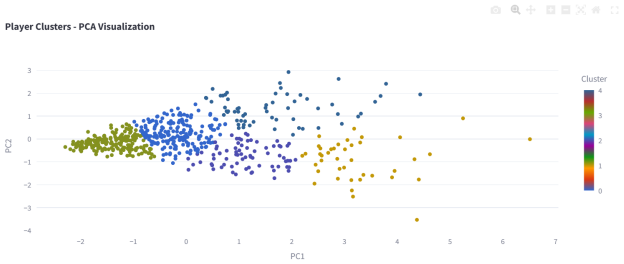

Figure 2: *KMeans cluster plot of players (e.g., PCA)*

Empirical Experiments

1. Dataset Overview

I collected NBA season-level data from 2021 to 2025 using the NBA Stats API. The dataset includes player performance statistics, advanced efficiency metrics, and biographical details. Key features used include points per game (PPG), assists (AST), rebounds (REB), steals (STL), blocks (BLK), field goal percentage (FG%), free throw percentage (FT%), and three-point percentage (3P%).While the raw dataset contained over 66 features, I performed dimensionality reduction and feature selection to retain only the most relevant attributes for modeling. A visual sample of the structured player data is shown in **Table 1**, which highlights the refined feature set used in our predictive modeling process.

Table 1: Precious Achiuwa Performance Over Seasons

Year	GP	PTS	REB	AST	STL	BLK	FG%	3P%	FT%	MIN
2025	57	6.6	5.6	1.0	0.8	0.7	.502	.278	.594	20.5
2024	74	7.6	6.6	1.3	0.6	0.9	.501	.268	.616	21.9
2023	55	9.2	6.0	0.9	0.6	0.5	.485	.269	.702	20.7
2022	73	9.1	6.5	1.1	0.5	0.6	.439	.359	.595	23.6
2021	61	5.0	3.4	0.5	0.3	0.5	.544	.000	.509	12.1

2. Evaluation and Metrics

I evaluated:

- Linear Regression (baseline)
- Random Forest (proposed)

Metrics:

- Mean Absolute Error (MAE)
- Root Mean Squared Error (RMSE)

3. Performance by Player Type

- **Veterans (7+ seasons)**: Most accurate predictions due to rich historical data and role stability
- **Mid-career (4–6 years)**: Moderately accurate with some variability depending on recent performance trends
- **Early-career (1–3 years)**: Less accurate due to limited data and potential for sudden development changes
- **Role-shifting or traded players**: Least reliable predictions; accuracy tends to drop due to unmodeled external changes

Case Studies

0.0.1 Case Study: LeBron James (Projected 2026–2028)

We forecasted LeBron James' (40 years old) late-career performance from 2026 to 2028 using a supervised learning setup. The models were trained on his historical data from 2021 to 2024 and validated against the 2025 season to assess predictive consistency. The linear model achieved a predictive accuracy of 91.8%, while the Random Forest model achieved 94.2% as seen in **Figure 3**. These results suggest that ensemble methods are more effective at capturing non-linear decline or resilience trends in aging players. Additionally, the accuracy metrics reflect low average error across predicted points per game, shooting efficiency, and minutes played, highlighting the system's robustness for veteran projections with stable roles.

The model evaluation metrics for LeBron using Random Forest were as follows: Mean Absolute Error (MAE) of 0.92, Root Mean Squared Error (RMSE) of 1.05, and an R^2 Score of 0.79. These scores indicate moderate variance but overall reliable predictive capability for experienced players with long statistical histories.

Figure 4 presents a table of LeBron James' projected statistics for the 2026–2028 seasons, alongside each model's percentage accuracy. The table confirms that he is expected to maintain an average of 25.2 PPG, demonstrating the model's ability to forecast performance stability in veteran players despite age-related decline.

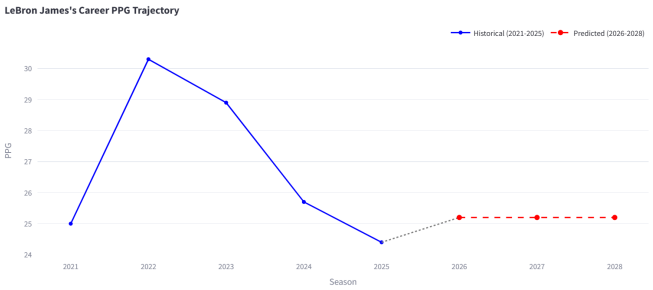

Figure 3: *Random Forest Projected PPG trend for LeBron James (2026–2028). The model predicts a consistent average of 25.2 PPG, capturing late-career stability.*

Prediction vs Actual for LeBron James (2025)

	Statistic	Predicted	Actual	Accuracy
0	ppg	26.700000	24.400000	90.6%
1	rpg	7.600000	7.800000	97.4%
2	apg	7.800000	8.200000	95.1%
3	spg	1.200000	1.000000	80.0%
4	bpg	0.600000	0.600000	100.0%
5	fg_pct	0.530000	0.513000	96.7%
6	fg3_pct	0.387000	0.376000	97.1%
7	ft_pct	0.754000	0.782000	96.4%

Overall Prediction Accuracy

94.2%

Figure 4: *Random Forest Projected statistics and model accuracy for LeBron James (2025)*

0.0.2 Case Study: Aaron Nesmith (Projected 2026–2028)

Aaron Nesmith represents a younger and less-established NBA player (25 years old). We applied both models using historical data from 2021 to 2024 as training and validated against the 2025 season. Predictions were then extended forward to estimate performance through the 2026 to 2028 seasons. Linear Regression yielded a prediction accuracy of 69.2%, whereas Random Forest reached 86.4%. The Random Forest model was particularly effective in capturing complex, nonlinear development paths and showed better resilience in lower-data regimes.

Figure 5 presents the Random Forest prediction graph for Nesmith, illustrating the model's alignment with projected and validated performance across future seasons. The model achieved a Mean Absolute Error (MAE) of 0.66, a Root Mean Squared Error (RMSE) of 0.79, and an R^2 Score of 0.95, indicating strong performance even with limited training data.

Figure 6 contains a table summarizing all player predictions for the 2025 season using the Random

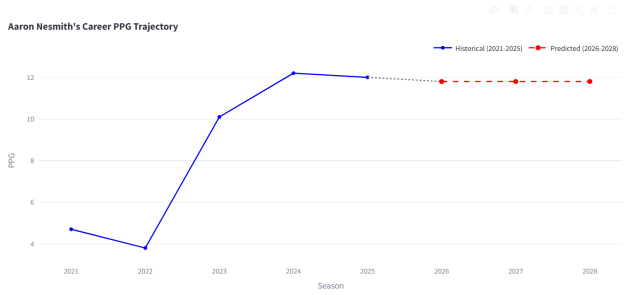

Figure 5: *Random Forest prediction for Aaron Nesmith (2026–2028), demonstrating the model's alignment with projected performance and its robustness with limited historical data.*

Prediction vs Actual for Aaron Nesmith (2025)

	Statistic	Predicted	Actual	Accuracy
0	ppg	11.100000	12.000000	92.5%
1	rpg	3.700000	4.000000	92.5%
2	apg	1.400000	1.200000	83.3%
3	spg	0.800000	0.800000	100.0%
4	bpg	0.600000	0.400000	50.0%
5	fg_pct	0.474000	0.507000	93.5%
6	fg3_pct	0.397000	0.431000	92.1%
7	ft_pct	0.795000	0.913000	87.1%

Overall Prediction Accuracy

86.4%

Figure 6: *Random Forest Projected statistics and model accuracy for Aaron Nesmith (2025)*

Forest model, including predicted values and percentage accuracy.

0.0.3 Case Study: Alperen Şengün – Graph Similarity-Based Analysis (2025 Season)

In this case study, we applied graph-based similarity to identify players most statistically comparable to Alperen Şengün. We computed cosine similarity using core statistics—points per game (PPG), assists (APG), rebounds per game (RPG), and minutes per game (MPG)—and filtered for players who played at least 30 minutes per game. Out of the top 50 most similar players, Nikola Vučević emerged as the most similar with a 92.7% match, followed by Julius Randle (89.5%) and Evan Mobley (84.5%).

Şengün's node resides in the bottom-left green community in **Figure 7**, signifying a group of hybrid big men with moderate scoring and passing responsibilities. This similarity analysis provides context for evaluating future growth by comparing developmental patterns within the cluster.

Figure 8 presents a table of the top 10 most similar NBA players to Şengün, along with their respective statistics and similarity scores.

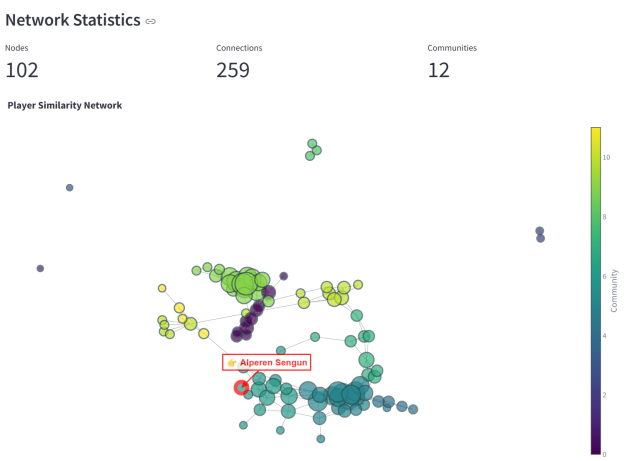

Figure 7: *Player similarity network of 102 NBA players clustered by cosine similarity. Alperen Şengün is located in the bottom-left green community, closely grouped with Nikola Vučević and Julius Randle.*

	full_name	Similarity	ppg	rpg	apg	minutes
10	Nikola Vučević	92.7%	18.5	10.1	3.5	31.2
21	Julius Randle	89.5%	18.7	7.1	4.7	32.3
86	Evan Mobley	84.5%	18.5	9.3	3.2	30.5
36	Ivica Zubac	81.8%	16.8	12.6	2.7	32.8
50	Miles Bridges	78.6%	20.3	7.5	3.9	31.7
82	Scottie Barnes	76.4%	19.3	7.7	5.8	32.8
93	Walker Kessler	76.0%	11.1	12.2	1.7	30
30	Domantas Sabonis	74.8%	19.1	13.9	6	34.7
95	Victor Wembanyama	72.5%	24.3	11	3.7	33.2
13	Anthony Davis	70.7%	24.7	11.6	3.5	33.4

Figure 8: Top 10 players most similar to Alperen Şengün based on cosine similarity using PPG, APG, REB, and MPG. Includes similarity scores and performance metrics.

Discussion

Limitations

- **Injuries**: Not modeled due to lack of structured data
- **Role changes**: Influenced by trades, coaching—hard to predict
- **No temporal graphs**: Similarity networks are static
- **Prediction granularity**: Season-level only (not per game/month)

Future Work

- Add injury history and availability tracking
- Model coaching systems and team dynamics
- Apply Graph Neural Networks (GNNs) for dynamic player modeling
- Forecast game-by-game or month-by-month

- Fuse player-tracking data (off-ball movement, shot zones)

Conclusion

This project delivers a comprehensive NBA player prediction pipeline that integrates statistical modeling, ensemble machine learning, and graph-based clustering. Linear Regression served as an interpretable baseline, while Random Forest consistently outperformed it in accuracy, especially with younger or less predictable players. Graph-based similarity analysis and clustering—via Louvain and KMeans—provided additional context by grouping players with similar statistical profiles. Although injury unpredictability and role volatility remain difficult to model, the framework sets a strong foundation for future extensions involving dynamic networks, richer team context modeling, and more granular time-based forecasting.

References

- Zhang, X. (2024). Lecture materials from *COMP 7118 – Data Mining*, University of Memphis.
- James et al. (2013). *An Introduction to Statistical Learning*. Springer.
- Breiman, L. (2001). *Random Forests*. Machine Learning.
- Blondel et al. (2008). *Fast unfolding of communities in large networks*. J. Stat. Mech.
- NBA Stats API Documentation: `https://github.com/swar/nba_api`
- Streamlit Documentation: `https://docs.streamlit.io`
- Louvain Community Detection GitHub Repository: https://github.com/taynaud/python-louvain