# Decision Tree Classification of NBA Game Outcomes

Andres Castellanos

## Motivation:

Basketball is one of the many sports that excites us data scientists due to the vast amount of data that could potentially be extracted and analyzed. For example, the National Basketball Association (NBA) generates vast quantities of game and player statistics each season, which can be potentially gathered for us to understand which statistical features strongly influence game outcomes, which can offer valuable insights for teams, analysts, and fans like me. The project I propose is to build a decision tree classifier to predict whether a home team wins or loses based on in-game performance metrics (e.g., Rebounds, Assists, turnovers, etc.).

Decision trees are very useful for us as they provide interpretable and computationally efficient models. Which makes it ideal to explain decisions and to identify any dominant patterns in the data set given. Since we are dealing with NBA data, this project will primarily be application-driven, by quantifying which statistics best predict victories, this project will aim to bridge computational foundations reasoning to real world sports analytics with real in game data.

## Datasets:

The data I used in this project comes from an SQLite database titled 'nba2024.db' which contains details about game, team, and player statistics focusing on the 2024-2025 NBA season.

Fortunately, this data was gathered with the assistance from Professor Kropko, by using the Python scripts he assisted me with the notebook that extracts, cleans, and even provides a useful structure that makes it convenient for analysis, this notebook is titled 'newNBAdata.ipynb' . Further details can be found in my repository link. Repository

Note: Primary table used for modeling is the join between 'games' and 'teamgames'.

| Database Summary | |
|---|---|
| Tables | games<br>teamgames<br>players<br>teams<br>playergames |
| Records | 1,200 games<br>30 teams |

# Decision Tree Classification of NBA Game Outcomes

Andres Castellanos

| | |
|---|---|
| | 450 players |
| Key Variables | points<br>assists<br>rebounds<br>turnovers<br>team win/loss outcome |
| Output/Outcome | Home team win = 1<br>Otherwise = 0 |

## Related Work:

With sports analytics being a big field, there has been a growing of research done on applying machine learning methods to predict sport teams winning outcomes, with one of the growing domains being basketball. Below are some related papers/research I have found that could be relevant to what I am trying to achieve.

1. Integration of machine learning XGBoost and SHAP models for NBA game outcome prediction and quantitative analysis methodology: An interesting paper discusses the application of a real-time predictive model for NBA game outcomes, integrating the machine learning XGBoost and SHAP algorithm. The authors found that the XGBoost algorithm was highly effective in predicting NBA game outcomes.
2. The Application of Machine Learning Techniques for Predicting Match Results in Team Sport: A Review: In this paper, they review surveys of various ML applications for predicting outcomes of team sports from 1996-2019; they list some good algorithms, data types, and evaluation metrics.

## Technical Plan:

**Inputs:** of my plan will select numerical and categorical features from the nba2024.db (e.g., points, assists, rebounds, turnovers, shooting percentage).

Outputs: The output of my decision tree will be a binary output (1 or 0) , in which a 1 signifies the home team winning and a 0 otherwise.  a team victory

Technical Plan for Implementing in Python:

1. Load the NBA data from SQLite. The following is the example code I used to access the 2025 NBA data:

```
import pandas as pd
```

```
import sqlite3

nba = sqlite3.connect('nba2024.db')

myquery = ''' SELECT * FROM games INNER JOIN teamgames ON games.gameid = teamgames.gameid ''' pd.read_sql_query(myquery, con=nba)
```

2. Create the training/test split for now I have it as (80/20) tentatively
3. Train decision trees with the varying depths
4. Analyze the importance of any features and visualizing the decision boundaries
5. Lastly, I will measure the performance and overall computational cost

## Evaluation Plan:

My model will be used to evaluate model effectiveness and computational behavior through:

- Accuracy Metrics: Accuracy score precision, recall, and F1-score
- Planning to also do a complexity analysis to see how memory usage varies across different tree depths.
- Perhaps could do some qualitative assessment of the decision paths. (e.g. “if turnovers < 10 and rebounds > 5, then we predict home team to win.”)

This consists of my plan on how I will evaluate the model's effectiveness.