Introduction to problem and data

As basketball has evolved over the decades, there have been numerous aspects of the game that have changed drastically since the earlier decades. Beginning in the 2014–15 season, Golden State Warriors coach Steve Kerr popularized the "Death Lineup," which featured Stephen Curry, Klay Thompson, Harrison Barnes, and Draymond Green on the court with no traditional center. This evolution altered how one thinks about basketball positions.

The once well-defined five traditional positions—Point Guard (PG), Shooting Guard (SG), Small Forward (SF), Power Forward (PF), and Center (C)—have become more positionally flexible. Players in today's game often take on a number of varying responsibilities on the court, so rigid positional labels become increasingly irrelevant. In recognition of this trend, beginning with the 2019–20 season, the NBA official website employed an new, condensed set of positional labels: Guard, Guard-Forward, Forward-Guard, Forward, Forward-Center, Center-Forward, and Center.

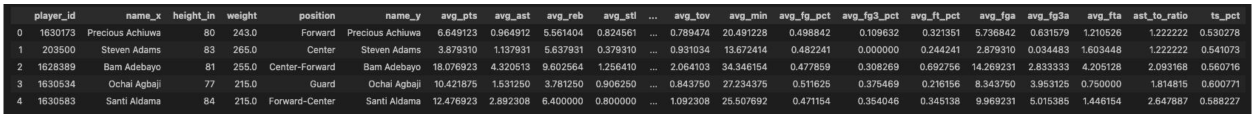
With this trend, the players are now more likely to play at multiple positions. As an example, a player who previously played solely as a PG may now be capable of playing as a SF. This project aims to leverage simple player data to predict their positional assignment, which will aid teams and coaches in determining player flexibility and making lineup adjustments or role changes.

Data Description:

The information for this project is collected using the nba_api Python library, which provides programmatic access to official NBA data. Two of the main endpoints were used to construct the dataset: commonplayerinfo (to collect each player's height, weight, and official position) and playergamelog (to collect game-by-game performance stats for the 2024-25 NBA season).

All endpoint raw data were cleaned and appended with player ID as the primary key to generate a merged dataset where each row is one of the distinct players. The merged dataset comprises a total of 559 NBA active players and 21 attributes: average points, assists, rebounds, shooting percentages, and usage-based measures (such as true shooting percentage and assist-to-turnover ratio).

This data is particularly appropriate for classification problems to forecast a player's positional category based on their physical profile and on-court play. It reflects modern NBA positional versatility, in which players no longer tend to fit traditional five-position profiles.

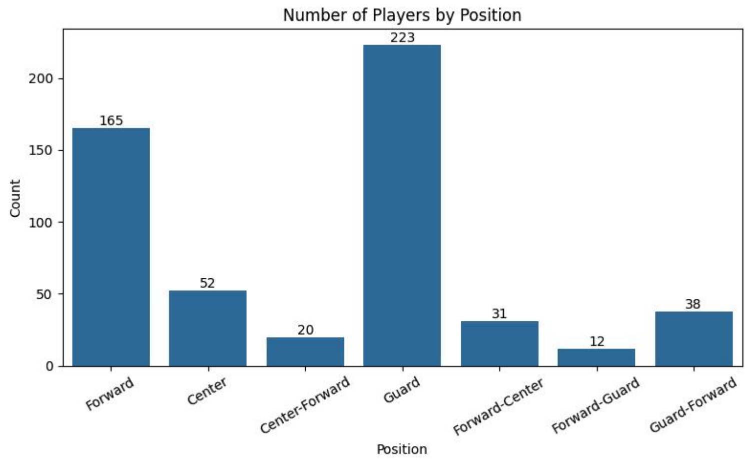


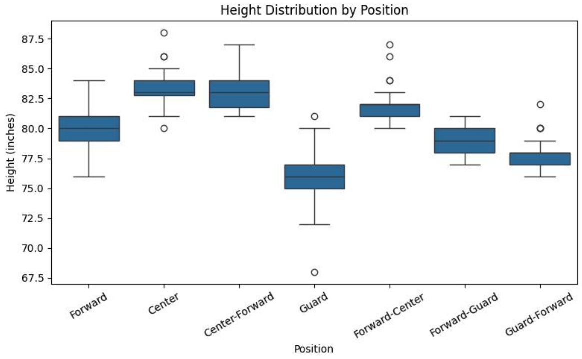
	player_id	name_x	height_in	weight	position	name_y	avg_pts	avg_ast	avg_reb	avg_stl	...	avg_tov	avg_min	avg_fg_pct	avg_fg3_pct	avg_ft_pct	avg_fga	avg_fg3a	avg_fta	ast_to_ratio	ts_pct
0	1630173	Precious Achiuwa	80	243.0	Forward	Precious Achiuwa	6.649123	0.964912	5.561404	0.824561	...	0.789474	20.491228	0.498842	0.109632	0.321351	5.736842	0.631579	1.210526	1.222222	0.530278
1	203500	Steven Adams	83	265.0	Center	Steven Adams	3.879310	1.137931	5.637931	0.379310	...	0.931034	13.672414	0.482241	0.000000	0.244241	2.879310	0.034483	1.603448	1.222222	0.541073
2	1628389	Bam Adebayo	81	255.0	Center-Forward	Bam Adebayo	18.076923	4.320513	9.602564	1.256410	...	2.064103	34.346154	0.477859	0.308269	0.692756	14.269231	2.833333	4.205128	2.093168	0.560716
3	1630534	Ochai Agbaji	77	215.0	Guard	Ochai Agbaji	10.421875	1.531250	3.781250	0.906250	...	0.843750	27.234375	0.511625	0.375469	0.216156	8.343750	3.953125	0.750000	1.814815	0.600771
4	1630583	Santi Aldama	84	215.0	Forward-Center	Santi Aldama	12.476923	2.892308	6.400000	0.800000	...	1.092308	25.507692	0.471154	0.354046	0.345138	9.969231	5.015385	1.446154	2.647887	0.588227

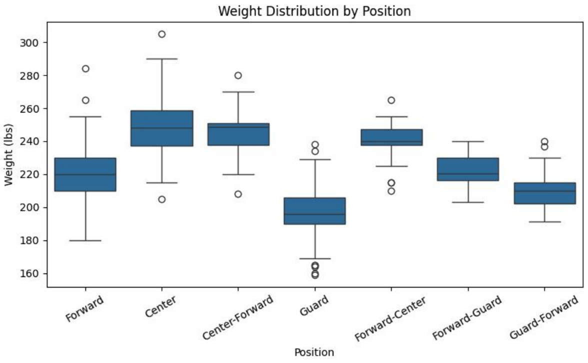
Exploratory Data Analysis

To better comprehend the dataset's structure and its nature, we began with a series of exploratory plots that show data in three dimensions, the core players' data, players' core stats on the court and some advanced stats on the court, based on different players' position.

The initial bar chart shows the distribution of players across seven NBA position types. "Guard" is the most common designation by far, with 223 players, followed by "Forward" with 165. Hybrid roles like "Center-Forward", "Forward-Guard", and "Guard-Forward" are relatively rare, with fewer than 40 players in each. This bias may impact the performance of machine learning models, especially for underrepresented roles.





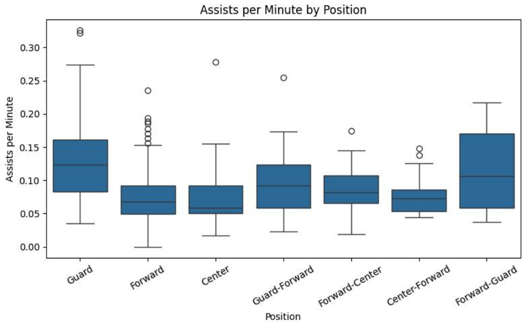


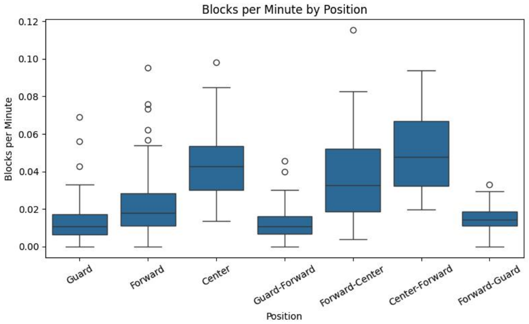
The following two plots—the height and weight boxplots by position—are highlighting the physical distinctions among positions in NBA players. Both Center-Forwards and Centers are the heaviest and tallest of all, having median heights over 83 inches and weights oftentimes more than 250 pounds. Guards tend to be lighter and shorter across the board, typically averaging 75–77 inches in height and 180–220 pounds in weight. Forward-Centers are also well-aligned with traditional interior roles in terms of height and weight. Both of these physical characteristics—height and weight—are employed as good predictors to differentiate between interior positions like Centers and more perimeter roles like Guards.

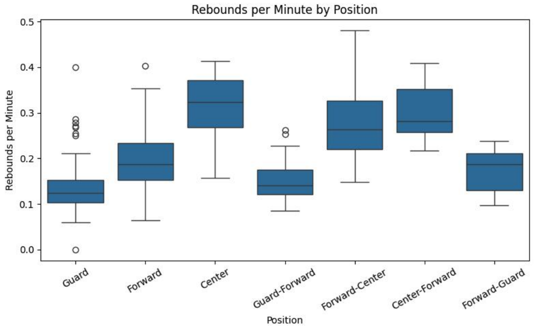
These EDA findings not only confirm intuitive physical patterns by position but also offer evidence for including height and weight as predictive features when modeling NBA positional assignments.

In addition to these physical dimensions described previously, we continued with exploring position-based discrepancies in per-minute performance data measures namely rebounds, assists, and blocks. Through the use of these boxplots, insight can be gleaned on the reflection of on-court roles through contribution during a game.

These plots of per-minute rebounds, per-minute assists, and per-minute blocks altogether elucidate varying positions' function-oriented roles on court. Interior roles such as Centers and Forward-Centers consistently possess greater rebounding and blocking percentages, which reflects their defensive requirements in front of the basket. Guards, on the other hand, possess leading assist percentages, which reflects their role as distributors and coordinators of offense. Half-breed roles such as Guard-Forward and Forward-Guard are more diverse through these statistics, which is appropriate to their versatile role in the game. In general, these trends are quite consistent with conventional views of how various positions statistically contribute depending on their on-court roles.

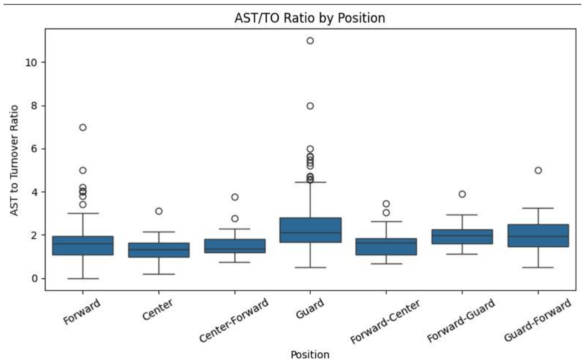


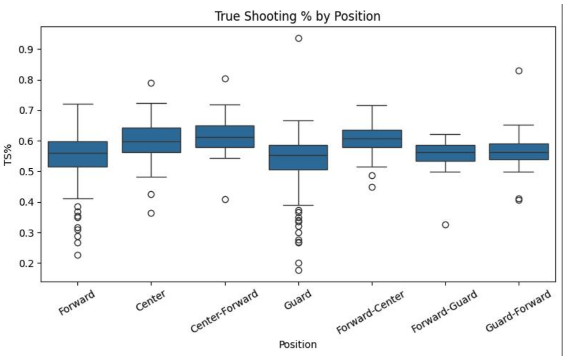




Together, these three players demonstrate the extent to which influential roles agree with statistical probabilities within the NBA. Guards are tasked with distribution (assists), while big men dominate physical statistics (rebounds and blocks). These tendencies not only support our model design choices but also help to confirm the validity of per-minute performance statistics in classifying positions.

Aside from traditional box score metrics, advanced performance statistics like Assist-to-Turnover Ratio (AST/TO) and True Shooting Percentage (TS%) offer better insight into the functional differences between positions.

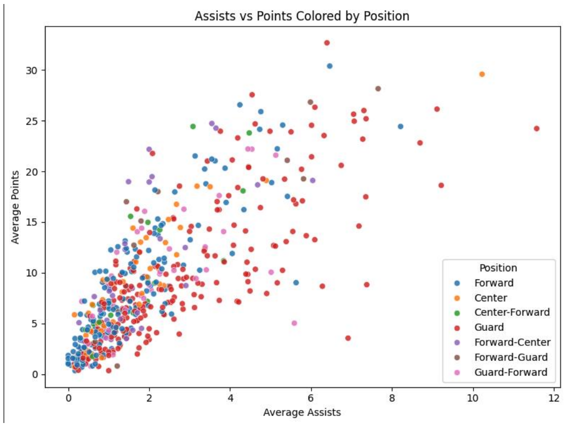


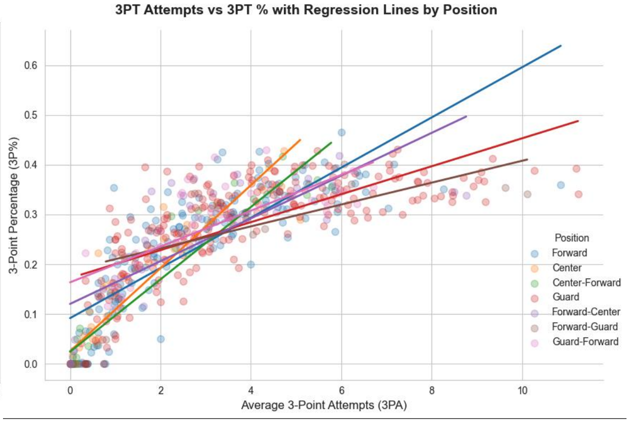


The AST/TO ratio chart and True Shooting Percentage (TS%) chart further illustrate positional differences in style of play and effectiveness. Guards average higher AST/TO ratios that reflect their middle position of handling and decision-making, while big men and hybrid players average lower ratios. Offensively, Centers and Center-Forwards average higher TS% due to their proximity to the basket and efforts at higher-percentage shots, while Guards are more boom-or-bust, motivated by their perimeter shooting.

These trends demonstrate that even advanced, rate-based stats have clear positional patterns. Guards prioritize control and creation with fewer mistakes (high AST/TO), and big men optimize scoring efficiency with simpler, closer-range shots (high TS%). These are core differences when creating position-sensitised prediction models and articulate how style of play affects statistical profiles across positions.

Some other EDA are shown below:

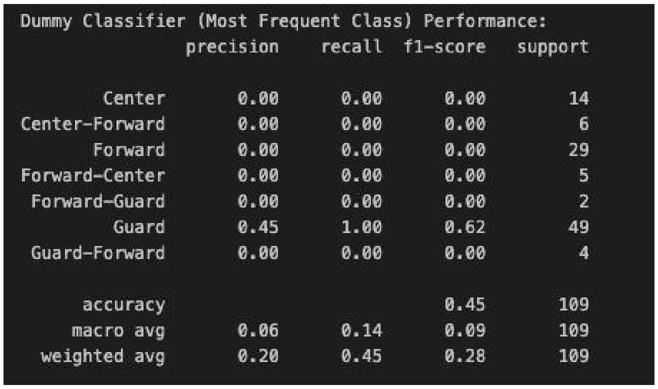




Model and Interpretation

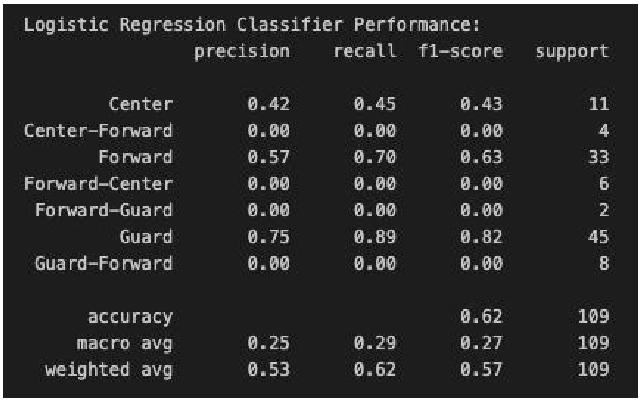
To classify NBA player positions, I decided to implement several classification models and compare their performance to determine which one best predicts a player's positional category based on their stats and physical attributes. Since this is a multi-class classification problem, I adopted an 80–20 train-test split, where the model is trained on 80% of the dataset and tested on the remaining 20%. This setup allows me to evaluate how well each model generalizes to unseen data.

1. Dummy Classification (Baseline)
   To establish the relative performance of my classification models, I first trained a baseline model using a dummy classifier. The baseline merely predicts the most frequent positional class for each player so that comparisons could be made to a simple but reliable approach. This provides me with a benchmark to compare to the extent to which more complex models do actually make any discernible difference.



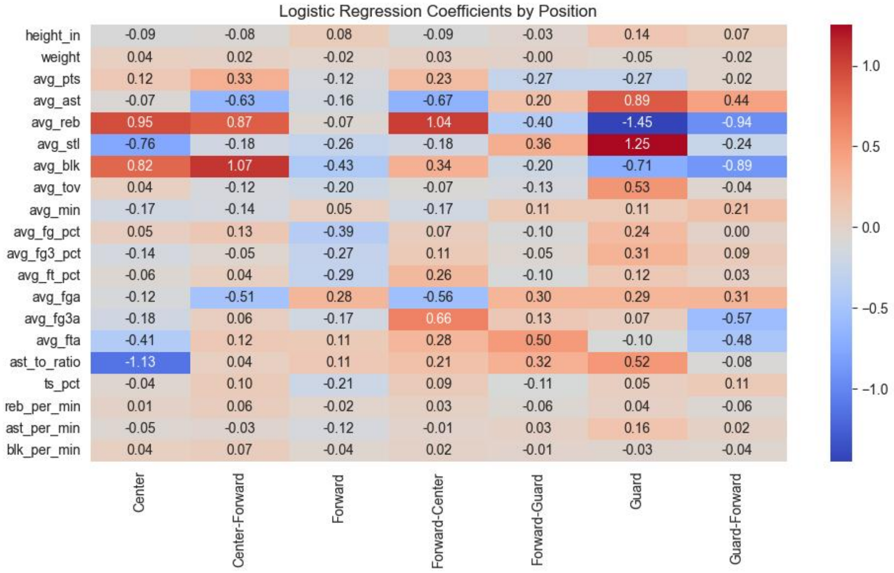
```
Dummy Classifier (Most Frequent Class) Performance:
                precision    recall  f1-score   support

         Center       0.00      0.00      0.00        14
 Center-Forward       0.00      0.00      0.00         6
        Forward       0.00      0.00      0.00        29
 Forward-Center       0.00      0.00      0.00         5
  Forward-Guard       0.00      0.00      0.00         2
          Guard       0.45      1.00      0.62        49
  Guard-Forward       0.00      0.00      0.00         4

       accuracy                           0.45       109
      macro avg       0.06      0.14      0.09       109
   weighted avg       0.20      0.45      0.28       109
```

2. Logistic Regression Model
   The Logistic Regression model was implemented as a linear and interpretable approach to multi-class classification. Trained on the same 80–20 train-test split using L-BFGS optimization, the model achieved a test accuracy of 64%, which, while slightly lower than that of the Random Forest, was notably better than the baseline. It performed reasonably well on majority classes like “Guard” and “Forward,” but struggled significantly with minority positions such as “Guard-Forward” and “Forward-Guard,” where both precision and recall were zero.

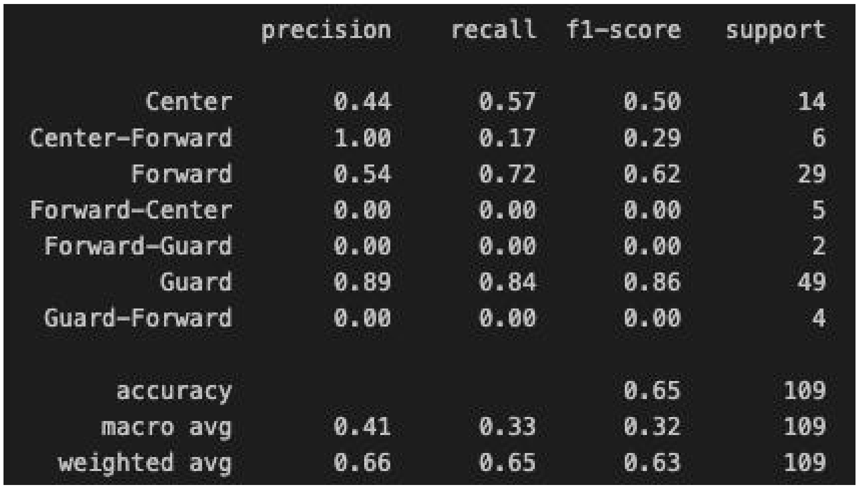


```
Logistic Regression Classifier Performance:
                precision    recall  f1-score   support

         Center       0.42      0.45      0.43        11
 Center-Forward       0.00      0.00      0.00         4
        Forward       0.57      0.70      0.63        33
 Forward-Center       0.00      0.00      0.00         6
  Forward-Guard       0.00      0.00      0.00         2
          Guard       0.75      0.89      0.82        45
  Guard-Forward       0.00      0.00      0.00         8

       accuracy                           0.62       109
      macro avg       0.25      0.29      0.27       109
   weighted avg       0.53      0.62      0.57       109
```

Analysis of the coefficient heatmap revealed that different features contributed differently across positional classes. Guard-related skills like steals per game and assist-to-turnover ratio negatively correlated with the "Center" position, while interior metrics such as blocks and rebounds were strongly associated with roles like "Center" and "Center-Forward." Shooting statistics—including field goal and three-point percentages—were more indicative of guard positions. These results suggest that while Logistic Regression may lack the flexibility of tree-based models, it still offers valuable insights into the linear separability and statistical drivers of player roles.



3. K-Nearest Neighbors (KNN) Model
   To explore a non-parametric, distance-based classification method, I implemented a K-Nearest Neighbors (KNN) model with k=5. Given KNN's sensitivity to feature scale, the input data was standardized using StandardScaler within a pipeline. The model was trained and evaluated using the same 80–20 train-test split applied across all models to ensure consistency and comparability. With an overall test accuracy of 69%, the KNN model performed comparably to the Random Forest and better than the Logistic Regression and baseline models. It showed strong performance on well-represented classes such as "Forward" and "Guard."



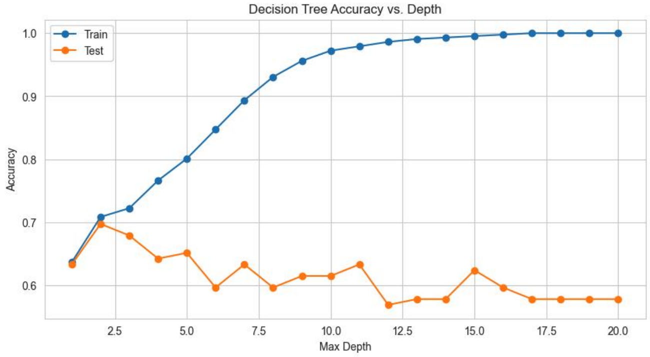
```
                precision    recall  f1-score   support

        Center       0.44      0.57      0.50        14
Center-Forward       1.00      0.17      0.29         6
       Forward       0.54      0.72      0.62        29
Forward-Center       0.00      0.00      0.00         5
 Forward-Guard       0.00      0.00      0.00         2
         Guard       0.89      0.84      0.86        49
 Guard-Forward       0.00      0.00      0.00         4

      accuracy                           0.65       109
     macro avg       0.41      0.33      0.32       109
  weighted avg       0.66      0.65      0.63       109
```

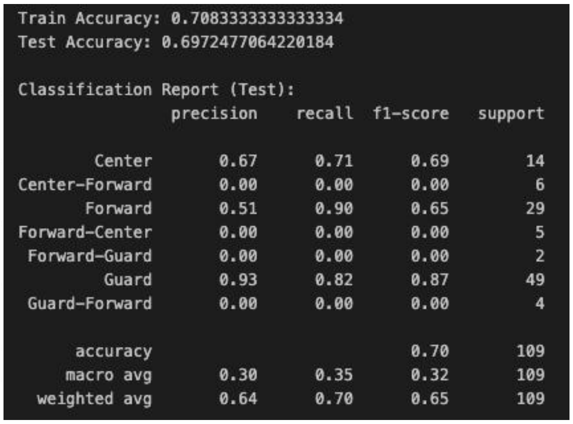
However, KNN performed poorly on underrepresented or hybrid positions like "Guard-Forward" and "Forward-Guard," where it failed to make any correct predictions—yielding precision, recall, and f1-scores of 0.00. The model's macro average f1-score was 0.34, while its weighted average f1-score was 0.66, reflecting its tendency to favor majority classes. These results illustrate KNN's limitations when applied to imbalanced multi-class problems, especially in contexts like NBA positions where certain roles share highly similar statistical profiles in the feature space.

4. Decision Tree Model

The Decision Tree model, with a maximum depth of 2, achieved a training accuracy of 70.8% and a test accuracy of 69.7%, outperforming both KNN and Random Forest overall. It performed well on majority classes like "Guard" and "Forward," but struggled with minority and hybrid positions such as "Guard-Forward" and "Forward-Guard," where predictive performance dropped significantly.

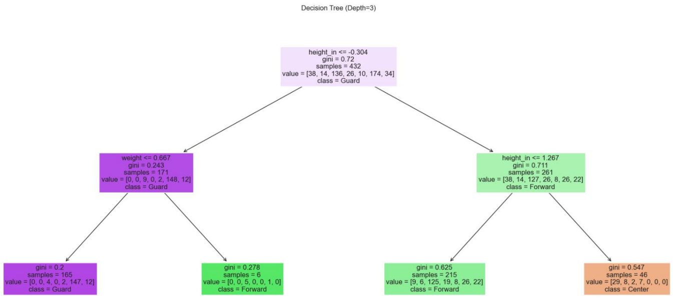
Accuracy vs. depth analysis revealed that deeper trees overfit the training data while harming test performance, confirming that a shallow tree helps balance generalization and interpretability. The chosen depth of 2 acts as a form of regularization, particularly valuable given the class imbalance.





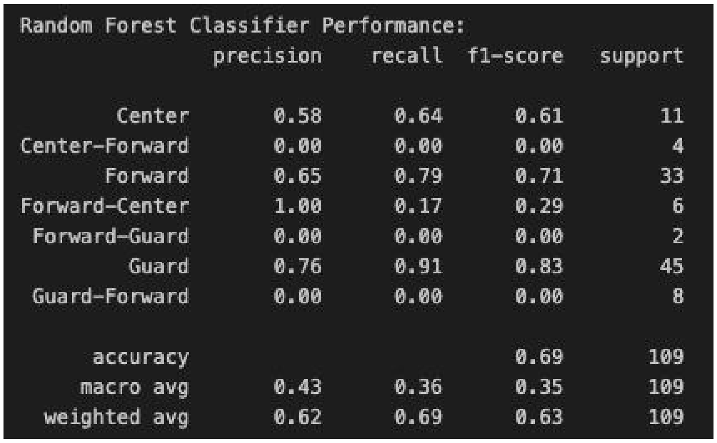
```
Train Accuracy: 0.7083333333333334
Test Accuracy: 0.6972477064220184

Classification Report (Test):
               precision    recall  f1-score   support

       Center       0.67      0.71      0.69        14
Center-Forward       0.00      0.00      0.00         6
      Forward       0.51      0.90      0.65        29
Forward-Center       0.00      0.00      0.00         5
 Forward-Guard       0.00      0.00      0.00         2
        Guard       0.93      0.82      0.87        49
 Guard-Forward       0.00      0.00      0.00         4

     accuracy                           0.70       109
    macro avg       0.30      0.35      0.32       109
 weighted avg       0.64      0.70      0.65       109
```

Visualizations showed that height and weight were the most influential features, with early splits clearly separating perimeter and interior players. Permutation importance further highlighted height, weight, and rebounds as key predictors. The model offers not just strong accuracy, but also interpretability—making it useful for understanding how player attributes relate to on-court roles.



5. Random Forest Classification Model

I implemented a Random Forest model due to its robustness and ability to capture complex, non-linear relationships between features. Given that player positions depend on both physical traits and performance metrics, this ensemble approach was well-suited for the task. Using an 80–20 train-test split, the model achieved a test accuracy of 69%, with strong performance on dominant classes like "Guard" and "Forward."



```
Random Forest Classifier Performance:
                precision    recall  f1-score   support

         Center       0.58      0.64      0.61        11
 Center-Forward       0.00      0.00      0.00         4
        Forward       0.65      0.79      0.71        33
 Forward-Center       1.00      0.17      0.29         6
  Forward-Guard       0.00      0.00      0.00         2
          Guard       0.76      0.91      0.83        45
  Guard-Forward       0.00      0.00      0.00         8

       accuracy                           0.69       109
      macro avg       0.43      0.36      0.35       109
   weighted avg       0.62      0.69      0.63       109
```

While predictions for hybrid and minority roles such as “Forward-Guard” and “Guard-Forward” were less reliable—largely due to class imbalance—the model still showed notable improvements in both macro and weighted f1-scores. This indicates not only better overall precision but also more balanced classification across classes. Key features like shooting efficiency, rebounding, and minutes played were strong signals for predicting player roles.

Discussion

In my models classifying NBA player positions, all the models performed better than the baseline dummy classifier, indicating their ability to extract meaningful patterns from both statistical and physical features. Among the models, the Decision Tree achieved the best performance overall, followed closely by the Random Forest and K-Nearest Neighbors (KNN), with Logistic Regression trailing in accuracy and generalization. The Decision Tree model with a maximum depth of 4 reached a test accuracy of 69.7% and produced the highest weighted f1-score of 0.70. It demonstrated excellent classification for the most supported roles, such as Guards and Forwards, and offered interpretability through its visual decision paths and feature importance analysis. The Random Forest model showed similarly strong results for common positions, although it, too, struggled with hybrid and minority labels like Guard-Forward and Forward-Guard due to class imbalance and ambiguous definitions.

The KNN model delivered solid results on majority classes as well, achieving a 69% accuracy and a weighted f1-score of 0.66. However, it completely failed on underrepresented positions, highlighting its limitations in handling imbalanced multi-class classification. Logistic Regression, while the weakest model overall with an accuracy of 62% and weighted f1-score of 0.57, provided insights into feature influence via model coefficients, though its linearity made it less capable of separating complex patterns in the data.

To improve the performance and applicability of such models, several enhancements should be considered. One key limitation lies in the fuzziness of some hybrid positional labels. Positions like Forward-Guard and Guard-Forward often blend responsibilities and roles, making them difficult to distinguish using statistical features alone. This ambiguity naturally limits model accuracy and consistency. Additionally, severe class imbalance reduced the models' effectiveness for minority positions. Expanding the dataset to include more players in underrepresented roles would help address this issue and improve model generalization. Collecting new data from recent seasons or international leagues could enrich the dataset further and reveal finer distinctions between roles.

From a real-world perspective, this modeling framework could eventually be used by coaches and analysts as a tool to inform position assignments based on a player’s physical and statistical profile. However, such predictions should serve as data-driven suggestions, not rigid labels, as actual deployment depends heavily on team strategy, lineup needs, and coaching philosophies. Addressing label clarity, increasing data coverage, and mitigating class imbalance will collectively make the model a more effective decision-support system in professional basketball contexts.