Final Mini Project – Matt Lindeman

**Introduction**

At the outset of this project, I wondered what an NBA player’s personal statistics can predict about how he fits into his team. The purpose of this project is to see how well an NBA player’s individual numeric data can predict his primary position and categorize his team role.

**EDA**

The data for this project comes from the [Sports Reference](https://www.basketball-reference.com/) website and contains the individual player data for the 2023-24 NBA regular season. While the entire summary statistics table is too large to include here, I’ll mention a few things that stood out to me. There are 572 players and 34 columns in the dataset (information on what the columns are and how the terms are defined can be found at the end of the report on the [Glossary](#Glossary) page). The average points per game per player is 8.4 pts, while the league high is 34.7 points per game. This shows that even though scoring is extremely important, many players will not be scoring huge numbers and will need to contribute in other ways to gain playing time. NBA players shoot on average 40% of their shots from three-point range, while multiple players shot 0 threes all season, and one player shot only three-pointers all season. This shows that most players need to have the ability to shoot threes to play in the modern NBA.

Correlation matrix of all the numeric variables in the dataset:

A grid of red and blue squares

Description automatically generated

There is a lot of information that can be gained from this matrix; however, I would just like to point out the two most negatively correlated variables, 3PAr and ORB. Generally, this means that the more offensive rebounds a player averages, the lower their proportion of three-point shots will be, and vice versa. Tying this back to one of the summary statistics I mentioned above regarding NBA players taking, on average, 40% of their shots from three, I think this correlation shows the small group of players that can survive in the NBA without having developed a three-point shot, and those are the players that are skilled offensive rebounders. This matrix also shows me that multicollinearity could be a potential issue to account for when fitting models later in my analysis.

Two tables representing boxplots of 3PAr and ORB by position:

A chart with different colored squares

Description automatically generated A graph of different colored boxes

Description automatically generated

These boxplots based on position are useful for my analysis because my predictive model that will be discussed later in the report aims to predict a player’s position based on their statistics. These plots show that both 3PAr and ORB will likely be useful variables in the predictive model because they are distinctive between positions.

Line plots by age for PTS and WS:

A graph showing a line of age

Description automatically generated with medium confidence A graph showing a line

Description automatically generated

These line plots helped me to get a glimpse into when NBA players regularly reach their performance peaks. From these plots, it appears to me that players generally start entering their prime around 27 years old, with the largest peak at 29 years old, and a lasting high until about 33 years old. I combined this finding with another interesting summary statistic that shows that the mean player age this season is 25.7 years old. Considering that most players enter the NBA at 19 years old, the average age is 26 years old, and player’s peaks generally end at 33 years old, it creates this general career arc that shows that players will take roughly 7 years to reach their full potential, with then 7 years at their full potential before retiring or taking a steep decline in production. I would also like to address the spike at age 39. LeBron James is still one of the best players in the NBA at that age, as well as being the only player that old currently in the league, so those numbers are based only on his production.

**Methods**

* Logistic Regression
  + Linear classification model that estimates the probability an input belongs to a particular class (in this case basketball position 1-5)
  + Regularization (C) and Penalty (L1/L2) explored
  + CV Accuracy: 0.491
* Linear SVC
  + Linear classification model that uses support vector machines to find the hyperplanes that best separate the classes
  + Regularization (C) and Loss (hinge/squared hinge) explored
  + CV Accuracy: 0.435
* Ridge Classifier
  + Linear classification model that applies ridge regression to choose classes
  + Regularization (alpha) explored
  + CV Accuracy: 0.448
* Decision Tree Classifier
  + Non-linear classification model that recursively splits the data into subsets, aiming to maximize the purity of the subsets
  + Max depth and minimum samples split explored
  + CV Accuracy: 0.412
* Random Forest Classifier
  + Ensemble learning method that builds multiple decision trees during training and combines their predictions
  + Number of estimators, maximum depth, minimum samples split, and minimum samples leaf explored
  + CV Accuracy: 0.472
* Ada Boost Classifier
  + Ensemble learning method that combines multiple weak learners to create a strong learner, with each subsequent model focusing on the misclassifications of the previous model
  + Number of estimators and learning rate explored
  + CV Accuracy: 0.432
* Gradient Boosting Classifier
  + Ensemble learning method that builds on weak learners to minimize a loss function, creating a strong learner
  + Number of estimators and learning rate explored
  + CV Accuracy: 0.474
* Feed Forward Neural Network
  + Artificial neural network where connections between nodes do not form a cycle, but consists of an input layer, one or more hidden layers, and an output layer
  + Number of hidden layers, number of neurons in each hidden layer, activation, learning rate, and dropout rate explored
  + CV Accuracy: 0.516

**Model Selection**

Linear models were consistently outperforming tree-based models by a slight margin; however, tree-based models seemed to respond better to the hyperparameter tuning than the linear models did. Ensemble methods were about the same as tree-based models in that they performed slightly worse than linear models but did respond better to hyperparameter tuning, so they did not show significant promise over single models. The linear models not responding well to hyperparameter tuning was a big challenge because through default settings they were outperforming tree-based and ensemble methods. The classification itself was also challenging because overall, point guards and centers were much easier to identify statistically than shooting guards, small forwards, and power forwards, so often the misclassifications would come from the latter three positions. I also had a big challenge combatting overfitting. Most of the models tended to overfit the training data even with hyperparameter tuning. The reason the linear models didn’t make the cut was their difficulty in hyperparameter tuning, and for tree-based and ensemble models, it was their poor performance on validation data and overfitting issues.

**Best Model**

The model that performed the best was the feed forward neural network. The hyperparameters that I tuned for this model were number of hidden layers, number of neurons in each hidden layer, activation, learning rate, and dropout rate. I eventually landed on 2 hidden layers, one layer has 128 neurons and the other has 64, the swish activation function, a learning rate of 0.001, and a dropout rate of 0.6. The performance metrics I used were precision, recall, f1-score, and accuracy. Precision returned an average of 0.6, recall an average of 0.57, f1-score an average of 0.57, and with an accuracy of 0.57. This model outperformed the other models in accuracy and is 0.37 points higher than an accuracy would be with random guessing since there are five basketball positions to classify from. I also performed a cluster analysis to find what roles the NBA players could be split into. I used a WCSS score, silhouette score, and a dendrogram to decide on their being 6 clusters, which can be seen in this t-SNE plot.

A diagram of a number of colored dots

Description automatically generated

I then categorized the 6 clusters into the role they fill for their teams based on the statistics averaged across the players in those clusters. Cluster 0 are a team’s key players. They are the ones that are likely playing the most, scoring the most, and contributing most to their team winning. Cluster 1 are a team’s reserve players. These players likely won’t play in every game of the season and get most of their playing time due to other injuries or a game not being close in score. Cluster 2 are a team’s back court players. These players are likely starters or big contributors off the bench. I refer to them as back court players because they are mostly in the guard positions and average a high number of assists of three-point shots. Cluster 3 are a team’s key reserve players. These players are not starters but will likely get minutes off the bench in every single game. They can contribute in many ways and likely will not have a particular specialty. Cluster 4 are a team’s front court players. These players are likely starters or big contributors off the bench. I refer to them as front court players because they are mostly in forward or center positions and average a high number of rebounds, blocks, and two point shot attempts. Cluster 5 are a team’s bench warmers. These players will rarely get playing time in a whole season, and are likely on the roster for development, knowledge, and injury insurance.

**Conclusion**

While predicting NBA players positions purely off their in-game statistics isn’t perfect, I believe that the feed forward neural network model I used provided accurate enough predictions to understand where on the court a player is likely to be playing. This model and cluster analysis could help teams understand what kind of statistics they need to look for in a player when needing to fill a certain hole in their roster. It could also help when players are repositioning to determine what position they might play best in. My purpose was to learn how well NBA in-game statistics could categorize a player, and the combination of my model and my cluster analysis have done well at addressing that question. However, there are many improvements that could be made in the future. More data, such as shot location tendencies, could be added in to further give distinctions to the players. Height could also be added in, which would likely give a boost to the accuracy of position classification. Further exploration could also be done within the clusters to figure out the skillsets of the players within. For example, finding which players are pass-first type of players within the key players cluster, or finding players that are good at defense and threes within the key reserves cluster.

**Glossary**

Player: NBA player’s name

Pos: Primary Position

Age: Player’s Age on Feb. 1st of the season

Tm: Team

G: # of games played

GS: # of games started

MP: Minutes played per game

FG: Field goals made per game

FGA: Field goals attempted per game

FG%: Field goal percentage

3P: Three-point field goals made per game

3PA: Three-point field goals attempted per game

3P%: Three-point field goal percentage

2P: Two-point field goals made per game

2PA: Two-point field goals attempted per game

2P%: Two-point field goal percentage

eFG%: Effective field goal percentage that adjusts for the greater worth of a three-point shot

FT: Free throws made per game

FTA: Free throws attempted per game

FT%: Free throw percentage

ORB: Offensive rebounds per game

DRB: Defensive rebounds per game

TRB: Total rebounds per game

AST: Assists per game

STL: Steals per game

BLK: Blocks per game

TOV: Turnovers per game

PF: Personal fouls per game

PTS: Points per game

TS%: True shooting percentage that takes into account twos, threes, and free throws

3PAr: Percentage of field goals attempts from three-point range

USG%: Percentage of team plays used by a player while on the floor

WS: Estimate of the number of wins contributed by the player

OBPM: Offensive points contributed per 100 possessions above a league-average player

DBPM: Defensive points contributed per 100 possessions above a league-average player

BPM: Box score points contributed per 100 possessions above a league-average player

VORP: Value over replacement level player