Using PCA and Clustering Techniques to Predict if an NBA Player is an All-Star

Additional Resources:
The data used in this project can be found at the following website: https://www.kaggle.com/datasets/sumitrodatta/nba-aba-baa-stats?select=Player+Per+Game.csv.

The code created for this project can be found at https://rpubs.com/cart0509/1132541

Background:

The National Basketball League (NBA) consists of 30 teams split into two conferences (15 teams per conference). Each year, at the mid-way point of the season, fans, coaches, and sportswriters vote for the most exemplary players in each conference to be represented in a game between the two conferences. Each All-Star team is comprised of 12 players for a total of 24 players being selected as All-Stars each year. There is a wide variety of skills that help qualify a player for All-Star selection. For example, some players possess strong scoring capabilities usually coupled with accurate shooting; others are defensively minded and rebound the ball at a high rate. Because of the vast range of abilities of NBA players, determining what qualifies a player as an All-Star can be challenging. This project attempts to make sense of these different attributes and predict if a player is an All-Star or not.

Purpose:

The first purpose of this project is dimensionality reduction. As described before, there are many characteristics and attributes that NBA players possess, and there are many ways to measure these attributes. We use Principal Component Analysis (PCA) to attempt to reduce the dimensionality of our data to better understand and visualize which attributes are related to a player's status as an All-Star or non-All-Star.

The second purpose of this project is to classify if a player is selected as an All-Star or not. To do this, we use the chosen number of Principal Components (PCs) to find cluster centroids across these PCs. The two clusters that we calculate centroids for are All-Stars and non-All-Stars. Using these cluster centroids, we assign a cluster label for new instances based on the Euclidean distance between the principal components of the new instance and the cluster centroids. The new instance receives label of the cluster with the closer centroid. We use this methodology to predict new instances and then analyze the results.

Data Pre-processing:

The dataset used contains many different csv files with data about various aspects of the NBA. In this project, we focus on two datasets: Player Per Game and All-Star Selections. The player per game dataset contains information about each player for a particular year. This data contains both categorical and numeric attributes about each player including team, position, age, and many statistics about their performance in that year. The All-Star Selections data contains far fewer attributes. The attributes of importance to us are name and season. We know that each player in this dataset was

an All-Star, so the absence of a player for a given year means they were not an All-Star that year.

Before using this data to perform our analysis, we created functions in R to clean to data for greater usability. The first action that we performed on the Player Per Game dataset was eliminating repeated players. When a player was traded mid-season, the dataset would represent this player three times: one set of statistics for each team and a combined total. We only are interested in the total, so our user created function eliminated any repeated instances and just used the combined total. Another user-created function was used to fill null values in the data. The null values in the data occurred for players that did not play many games and had yet to take a regular shot, a 3-pt shot, or a free throw. To fill these attributes, the user-created function used an estimate of the average for each attribute. The last user-created function combined the Player Per Game data and the All-Star Selections data. To do this, a new attribute called “All-Star” was created in the Player Per Game dataset. For each player and year, the All-Star selections dataset was searched and if there was a match in player and year, the All-Star attribute was set to TRUE, otherwise FALSE. This additional “All-Star” attribute eliminated the need for the All-Star Selections dataset for the rest of the analysis.

The last change to the Player Per Game data was the elimination of any categorical attribute. Because our PCA analysis will require only numeric attributes, we eliminate all categorical attributes and do not consider them for our analysis.

Before we use the Player Per Game data for our analysis, we split the data into a training and a testing set. For our training set, we use every player instance from 2000-2019. For our test set, we use only the player instances from 2020-2023. Though we have access to much more data for our training and testing sets, we choose to use only the players from the 2000 and later because they will be more representative of the instances we are trying to predict (recent occurrences, 2020-2023). Because the NBA has changed in playing style and player tendencies, we want data that represents the modern game. We choose 2020-23 for our testing set because this gives us ample data to test upon that represents recent occurrences; however, not so large of a set that it takes away from the PCA training process.

PCA Analysis:

We perform PCA analysis on the training set of data using the built-in R function “prcomp”. To decide how many PCs, we will keep going forward in our analysis, we generate a Scree plot (figure 1).

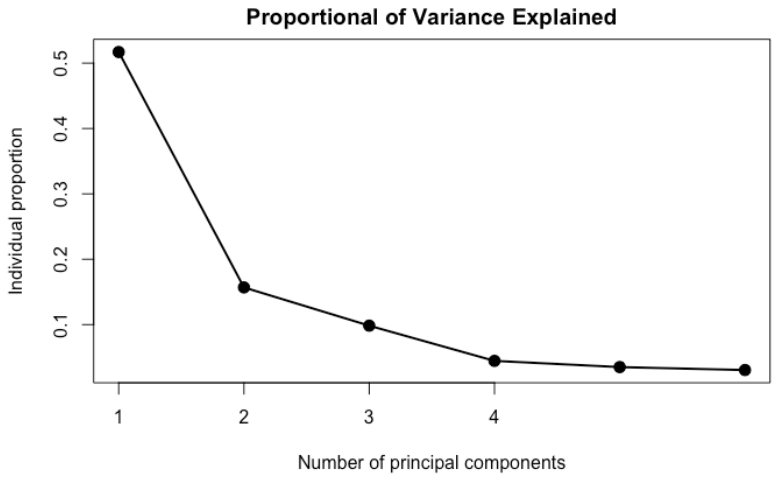

Figure 1

We find that the “elbow” in the plot lies at two principal components, so we choose two principal components for our analysis. Additionally, choosing two principal components helps us visualize our variables against the two components in a Biplot. We generate the Biplot (figure 2).

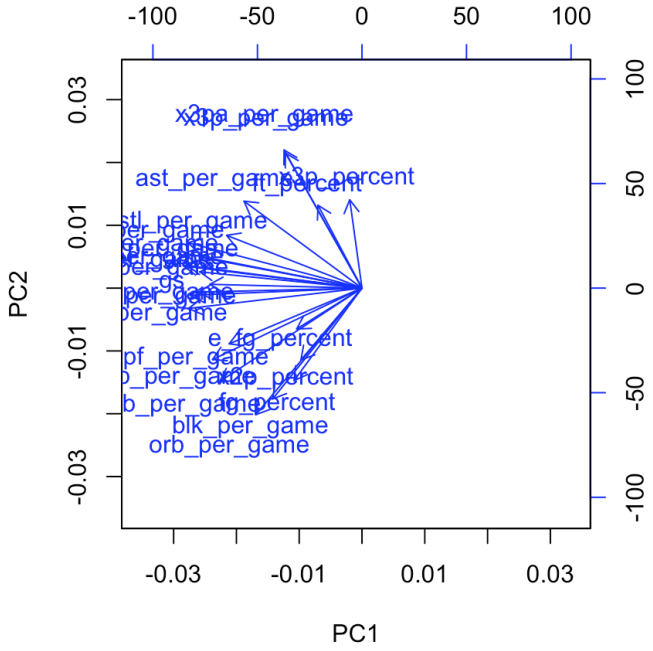

Figure 2

From the Biplot, we can see that negative values of PC1 for a particular instance are associated with higher positive values for all the variables. PC2, however, shows somewhat even distribution between positive and negative associations for the variables. From a subjective perspective, PC2 seems to be split between two play

styles: high shooting percentages from further away (3-pointers) and assists are associated with positive PC2 values, while rebounding and high shooting percentages from closer range (2-pointers) are associated with negative PC2 values. The former style is more fitting of smaller, more versatile players while the latter is associated with bigger, more defensive minded players.

In this preliminary analysis, we can see that PC1 will likely help us identify which players are all-stars. From our background in general basketball knowledge, we know that an increases in each of the attributes is related to an increase in performance. From our analysis, we see that higher magnitude negative values are associated with high values for each of our attributes. Therefore, we can expect All-Star players to have higher magnitude negative PC1 scores than non-All-Stars.

For PC2 however, the association is less straightforward. This PC is split between the variables, and a positive or negative PC2 score may not have a strong impact in determining whether a player is an All-Star.

Clustering and Predictive Analysis:

To cluster our data, we treat the group with attribute “All-Star” = TRUE as one cluster and “All-Star” = FALSE as our other cluster. To visualize the two clusters against our two PCs, we generate a scatter plot (figure 3).

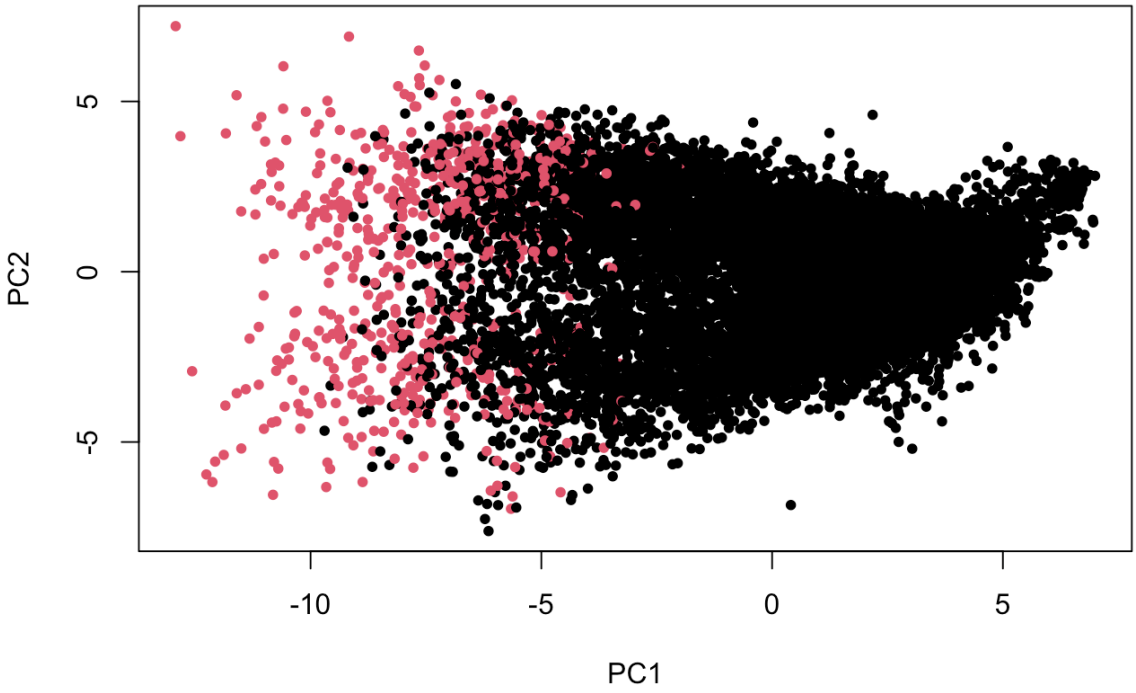

Figure 3

In this plot, red point represents players that were All-Stars and black points represent non-All-Stars. As we can see, higher magnitude, negative PC1 values are associated with All-Stars. Additionally, we can see that PC2 has seemingly little association with All-Stars. There are also many instances towards the middle of the plot that appear hard to classify due to the overlapping of the two clusters.

To classify our test data, we find the centroids across our first two PCs for the training data. We find that our centroid for the All-Star cluster is (–7.242039, 0.411792) and the centroid for the non-All-Star cluster is (0.42201856, -0.02399654) where the X coordinate represents the first PC, and the Y coordinate represents the second. As we predicted, there is a much greater cluster separation in PC1 than PC2. To calculate distances from our test data points to our clusters, we perform PCA once again on our test dataset. We generate a Biplot (Figure 4).

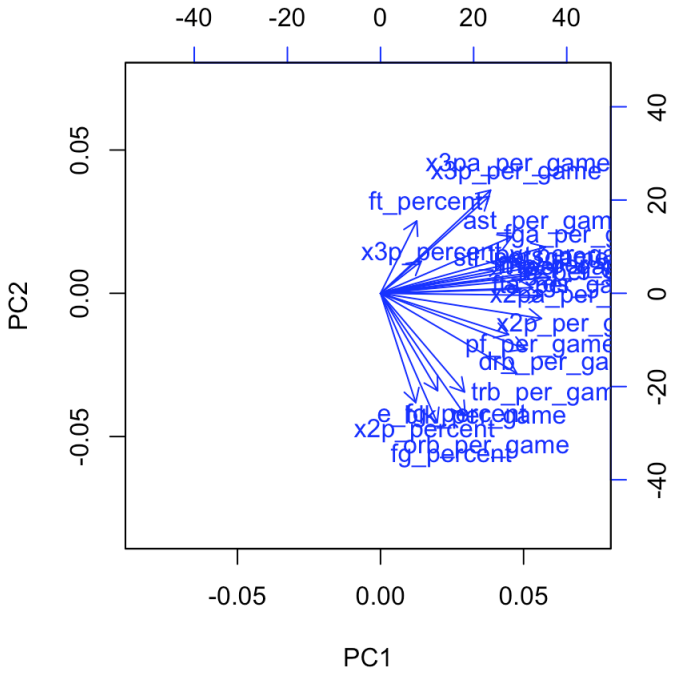

Figure 4

From our Biplots in figure 3 and figure 4, we can see that PC1 appears to be going in the opposite direction for our test set (positively correlated with each attribute in figure 4, negatively correlated in figure 3). PC2 appears to be going the same direction. To correct for this so the distance measurements are not skewed, we make the X coordinate of our centroids (PC1) negative.

Using a user-created function, we iterate through each instance in the test set and calculate the Euclidean distance between the instance's PCs and the cluster centroids. An attribute called "pred_label" is created. If the instance's PCs are closer to the All-Star centroid, we assign pred_label = TRUE; otherwise pred_label is set to FALSE. We now have two attributes in the test set that describe labels; "All-Star" shows the true cluster label (TRUE for All-Star, FALSE for non-All-Star) and the predicted cluster label, "pred_label". Comparing the predicted label to the true label allows us to analyze different metrics to determine the legitimacy of our classification methods.

Results and Conclusions:

We use user-generated functions to assess accuracy, precision, and recall scores for our predicted labels. Because this is an imbalanced class problem (many

more instances of non-All-Stars than All-Stars), precision and recall will help us gauge usefulness for both the majority and minority classes.

Our results were: Accuracy: 0.877, precision: 0.282, recall = 0.991. The overall accuracy indicates legitimacy of the techniques used. However, the precision and recall indicate that the model classified many false positives and not many false negatives. This means that for many of the instances, the model predicts the player to be an All-Star when they are not. There are a few reasons this happens.

One reason is that within the All-Star clustering area (the area of points that are classified as All-Stars), there are many non-All-Stars in the training data. Some of these points are impossible to classify as non-All-Stars as their distance from the centroid is very small. This wide clustering area causes our model to predict far too many All-Stars compared to the true number.

A second reason for the inaccuracy of the model is that the second PC gives us very little information about whether a player is an All-Star or not. This is seen by the marginal differences in PC2 between the All-Star and non-All-Star groups. By including PC2 in the Euclidean distance measures, we likely do not see an increase in accuracy of the model.

In the future, more PCs should be analyzed to see if there are great differences between cluster means. If there are significant differences, perhaps these PCs could be used to differentiate between clusters more effectively. Nonetheless, this analysis has shown the benefit of PCA for dimensionality reduction of NBA statistics and visualization. Additionally, while the clustering methods described have some strong deficiencies, the overall accuracy shows some viability for the classification process shown in this project.