Basketball Prediction Modeling and Player Reclassification

Kern Khanna, Daniel Ron

Clustering

Background- Currently, every basketball player is assigned a position from 1 through 5, which indicates their role on the court. These "labels" are primarily determined by the player's metrics, including weight and height. The shortest players (guards) are usually involved in ball handling, shooting, and passing and they occupy positions 1-3. Meanwhile, the larger players (forwards) occupy the post and focus on getting rebounds, blocks, and "disrupting the paint."

The major problem with the current system is that basketball players are "pigeonholed" into positions. This process is far from scientific and it greatly affects the evaluation and playing time for players. So, in this study we decided to classify players based on their "playing style" rather than their measurables.

Clustering Algorithm-

To determine how we should reposition basketball players, we decided to conduct an unsupervised learning technique called K-Means clustering to naturally sort a player into positions/clusters. We then conducted PCA Singular Value Decomposition to lower the dimension space and remove multicollinearity.

$$\sum_{i=0}^{n} \min_{\mu_j \in C}(||x_j - \mu_i||^2)$$

Input Data-

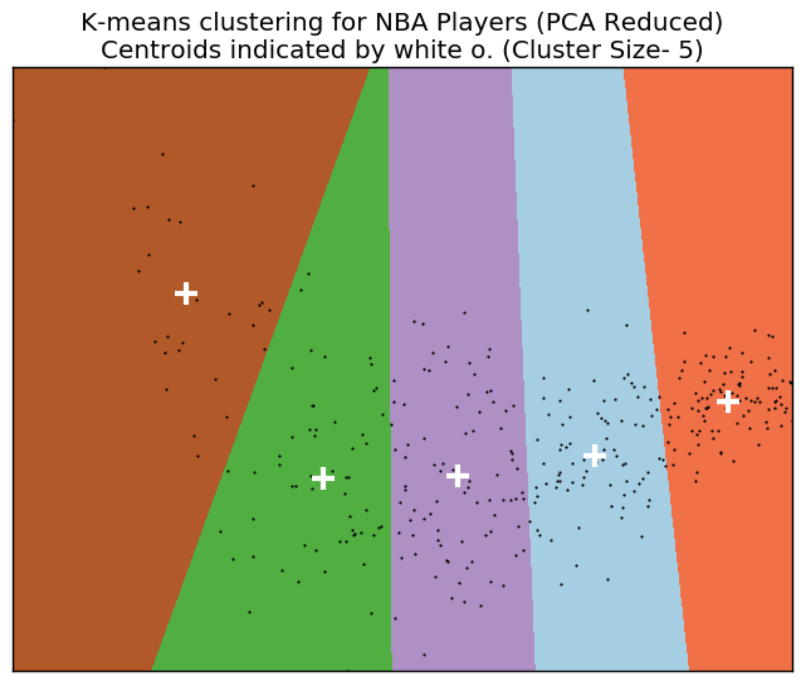
MP - Minutes Played
2P - 2-Point Field Goals
FT% - Free Throw Percentage
AST - Assists
TOV- Turnovers

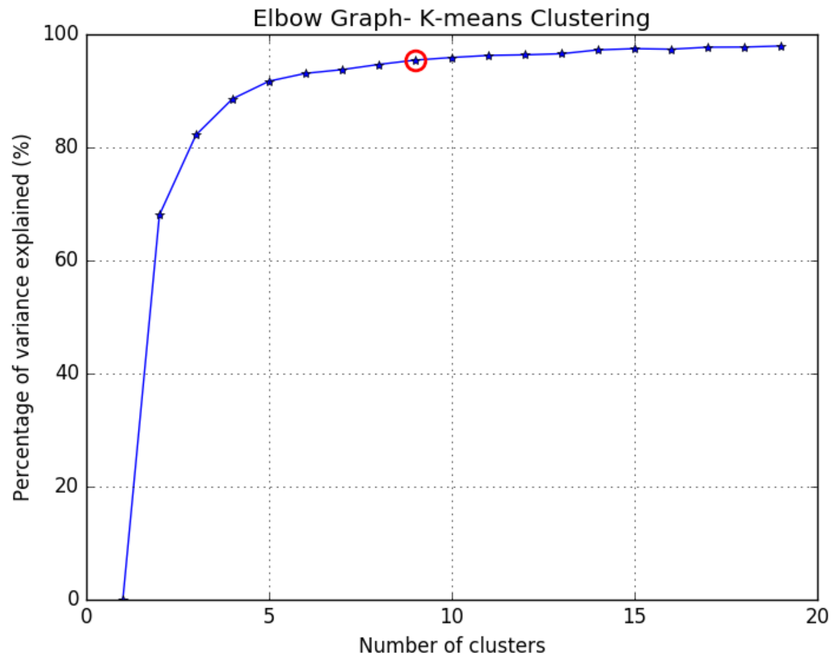
3P - 3-Point Field Goals
2P% - 2-Point Field Goal Percentage
ORB - Offensive Rebounds
STL - Steals
PF- Personal Fouls

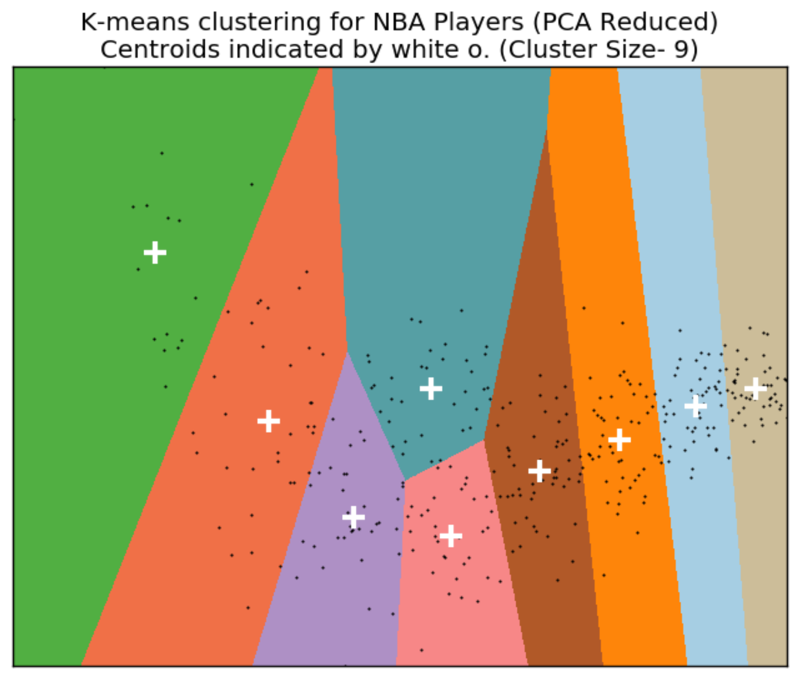
3P% - 3-Point Field Goal Percentage
FT - Free Throws
DRB- Defensive Rebounds
BLK - Blocks
PTS- Points

Example- Draymond Green- ['2808', '100', '.388', '301', '.537', '229', '.696', '134', '635', '598', '119', '113', '259', '240', '1131', '19.3', '18.8']

Data/Graphs- Each of the players was then classified into k-clusters. We didn't achieve much separability with five clusters, so we conducted an "elbow model" to determine the optimal number of clusters. The number of optimal clusters turned out to be 9.


K-means clustering for NBA Players (PCA Reduced)
Centroids indicated by white o. (Cluster Size- 5)

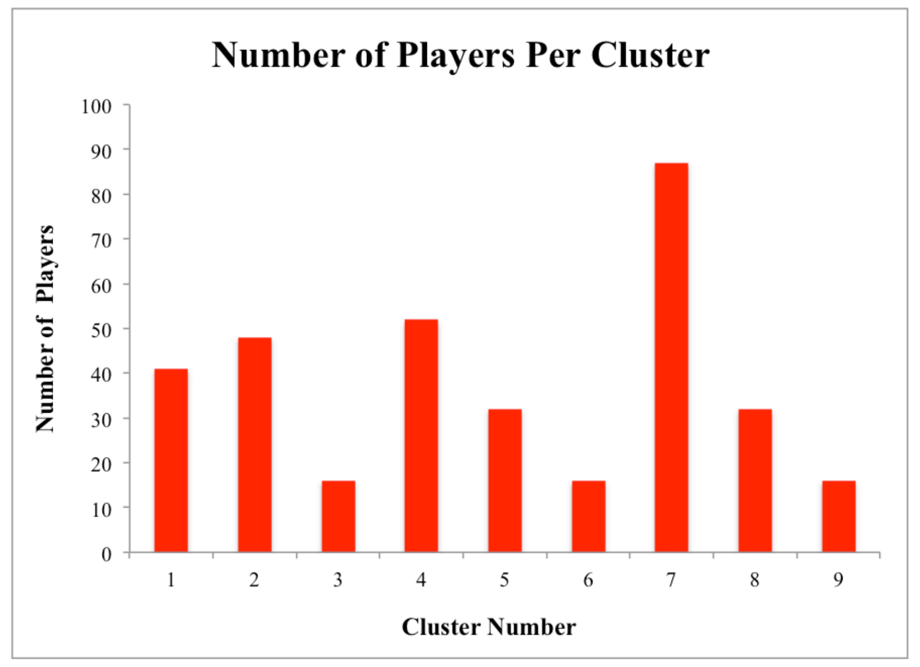

Elbow Graph- K-means Clustering


K-means clustering for NBA Players (PCA Reduced)
Centroids indicated by white o. (Cluster Size- 9)

Roster Analysis for NBA Teams-

Milwaukee Bucks
Current Lineup - 0.486 Win Share sum per 48 min
New Lineup (1-9 Clustering)- 0.62 Win Share sum per 48 min

Golden State Warriors
Current Lineup - 0.948 Win Share sum per game
New Lineup (1-9 Clustering)- 1.003 Win Share sum per 48 min



Prediction

Background- The NBA Playoffs is one of the sporting world's biggest tournaments, with bets on the postseason eclipsing a billion dollars. As a result, we tried to predict matchup outcomes and the overall winner through supervised learning techniques like regression and SVMs.

Algorithms-

In order to develop a comprehensive learner to predict basketball games, we experimented with Logistic Regression (L2 Regularization) and a Support Vector Machine.

$$\min_{w,c} \frac{1}{2}w^T w + C\sum_{i=1}^{n}\log(\exp(-y_i(X_i^T w + c)) + 1).$$

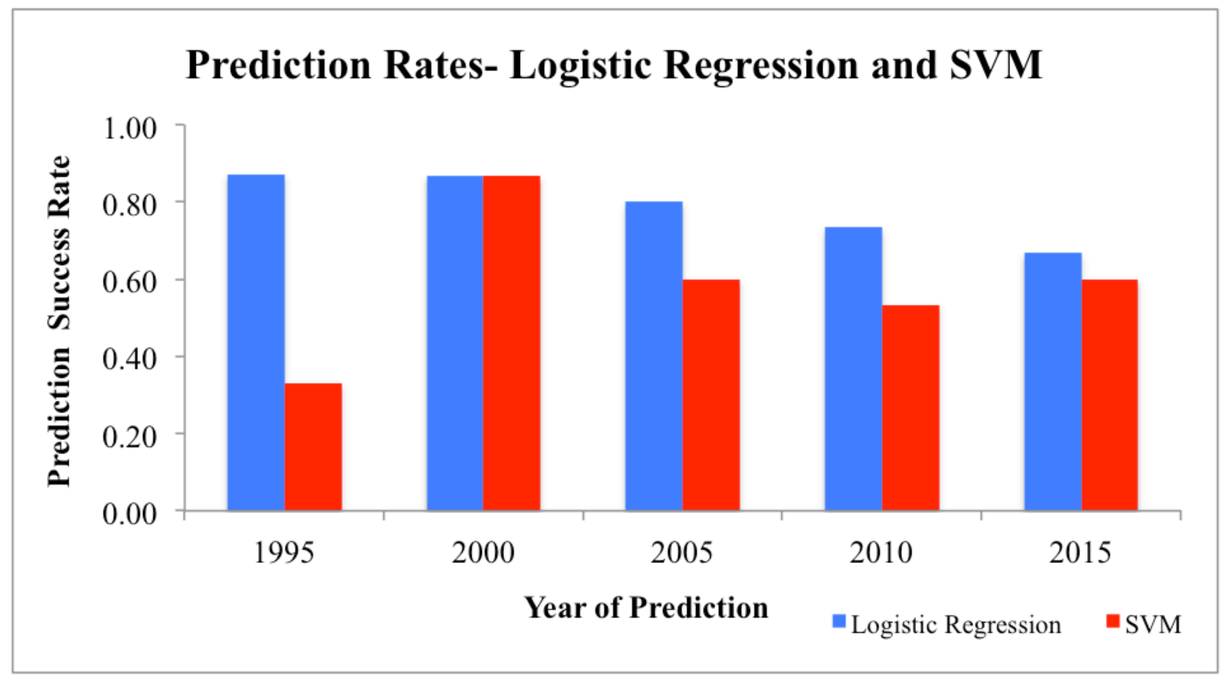
$$\min_{w,b,\zeta} \frac{1}{2}w^T w + C\sum_{i=1}^{n}\zeta_i$$
$$\text{subject to } y_i(w^T\phi(x_i) + b) \geq 1 - \zeta_i,$$
$$\zeta_i \geq 0, i = 1, ..., n$$

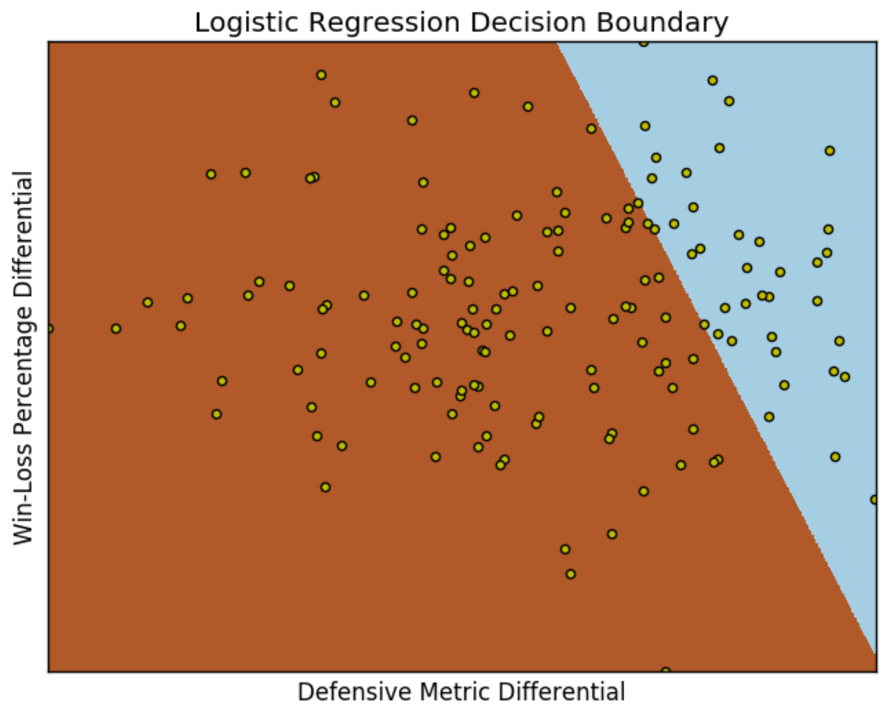
Input Data- There were many possible features to select from, but we picked our feature vector (Θ) to be-
(*Intercept, Win-Loss % _Team1, FG%_Team1, 3FG%_Team1, Turnovers_Team1, Defensive Metric_Team1, SOS_Team1, Win-Loss % _Team2, FG%_Team2, 3FG%_Team2, Turnovers_Team2, Defensive Metric_Team2, SOS_Team2)*

The feature vectors were calculated from a team's performance in that regular season. The algorithms were then trained and tested during the postseason.

NBA Finals Last Year- GS(0.817, 0.478, 0.398, 1185, 4072)
CLE(0.646, 0.458, 0.367, 1171, 3555)
Θ => (.817, 0.478, 0.398, 1185, 4072, 0.646, 0.458, 0.367, 1171, 3555)
OUTPUT: +1 (predicting a Warrior's win)

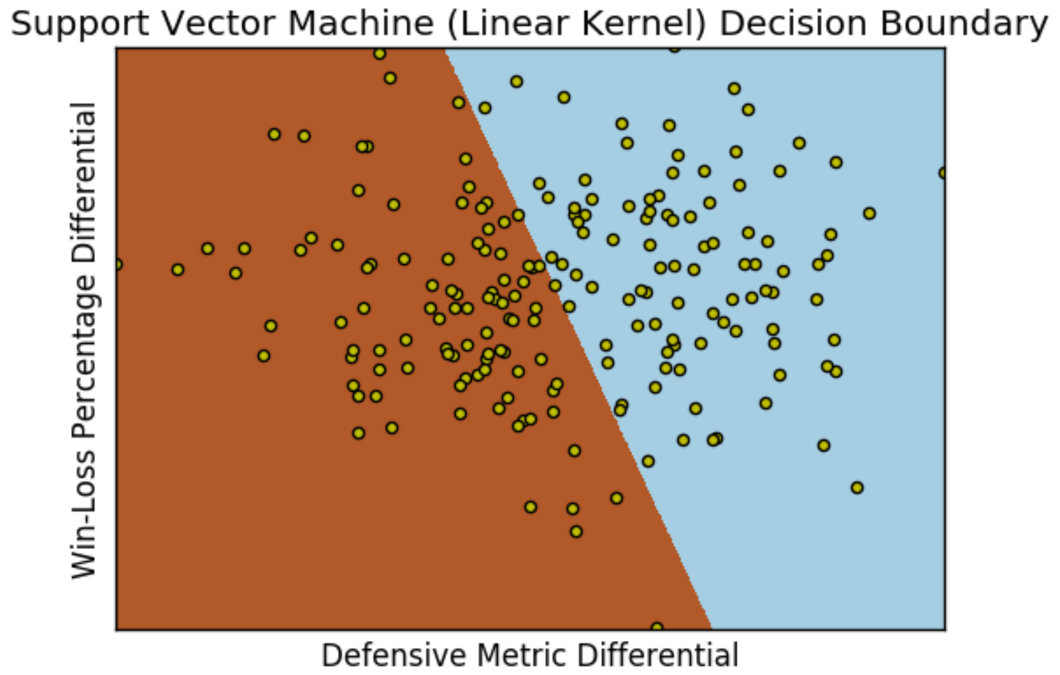
Data/Graphs-





As seen in the graphs, the logistic regression model performed better than the Support Vector Machine, indicating that the data may not be separable.
To evaluate our baseline, we compared our algorithm to ESPN pundits (data available only from 2010+), and the logistic regression model outperformed the mean analyst by about 5% percent.

To get a sense of how generalizable our algorithm was, we also tried to learn "March Madness"/college basketball.
In the 2015 tournament, our model placed in the 85th percentile of ESPN brackets after predicting 50/63 results correctly.
In the 2014 tournament, our model placed in the 73rd percentile after predicting 37/63 results correctly.


Support Vector Machine (Linear Kernel) Decision Boundary
Win-Loss Percentage Differential
Defensive Metric Differential