Prediction of NBA Game Outcomes and Player Outlier Detection using Machine Learning

Benjamin Naidoo (216063593), Kailin Reddy (218015230)

GitHub Repository: Repository for Project

Abstract:

In this paper, machine learning and outlier detection implements data surrounding the National Basketball Association (NBA). Multilayer perceptron and support vector machine (SVM) are two supervised regression approaches explored regarding NBA game outcome prediction. Thirty-one features are utilised for training each classifier. Multilayer perceptron outputted the most promising results. A maximum error of 0.168, explained variance score of 0.867, mean absolute error of 0.045, mean squared error of 0.003 and median absolute error of 0.039 and R2 score of 0.867. The model then produces the probability of a win for a particular team. The Principal Component Analysis (PCA) determines the remaining outlier players. A supervised K-Nearest Neighbour (KNN) approach is used to investigate a PCA, and an unsupervised K-Means approach is used for a PCA. After evaluating both approaches for outstanding player detection, the results show that the PCA with K-Means yielded better results as it determines more relevant outstanding outlier players in general.

Introduction:

The NBA is the leading basketball league in the world and one of the biggest sports leagues in the United States (US), with estimated revenues adding up to ~8 billion USD. NBA teams are. The average NBA team is valued at 2,2 billion USD, making them some of the most sought-after franchise sports brands. Matches are attended by ~twenty-one million people every year and maintain high TV ratings across households in the US. To understand how the NBA became such a huge international sports league, whose influence is felt throughout popular culture and media, emphasis should be made on understanding the circumstances behind its establishment.

The NBA established themselves in 1946, which is much later than most other large sports leagues. They had to spend loads of resources and time to catch up and root itself as a massive sports league. They used strategies that differentiate the NBA from the rest of the sports industry to a comparative edge over the other core sports organisations. Some of these innovations can include the NBA All-Star game (a three-day exhibition full of fan events that can drive the marketability of the league's best players and showcase the entertainment side of the sport). The NBA was the first sports league to invest highly in the merchandising part of professional sports, drawing a large sum of their overall profits from apparel and sportswear. These innovations helped the NBA develop some of the world's most marketable and famous sports superstars. Household names like Kobe Bryant and Michael Jordan were the main faces that greatly increased the league's popularity inside foreign markets. One of these foreign markets is China, which is currently in partnership with the NBA and generating an estimated five hundred million USD of revenue alone.

The NBA follows a format and schedule well-known within the sporting world. A regular season league system where all thirty teams play each other, and the top eight teams qualify for a single elimination knockout tournament that determines the victor. The NBA's scheduling system is where

they differ from other sports leagues. The NBA regular season schedule has had many minor changes, but the general format is as follows: Each of the thirty teams plays eighty-two games, forty-one home and away. The teams are split across two lines. Firstly conferences, east and west and secondly, divisions. For east Atlantic, Central, and Southeast. For West, Northwest, Pacific and Southwest. This categorises NBA teams by location across the USA and Canada. Teams play opponents in their division four times a year. Then plays six teams from the other two divisions in its conference four times. Then the remaining four teams were in the conference three times. Then every team in the other conference two times. This means the strength of the schedule varies. For the playoff format, the top eight teams in a conference are seeded and play each other in a single elimination best of seven, with the winner advancing. The two best teams in each conference play each other for the NBA championship. We can surmise that playoff games have higher stakes than regular season games.

From a machine learning (ML) standpoint, basketball, like every other layered multi-agent game, can produce a high volume of data which has become extremely valuable. A large amount of data is important for ML scientists to apply algorithms that can find patterns and make useful predictions. This makes the NBA a highly attractive platform for experimenting with ML techniques. Sports Analysis has been growing and gaining more popularity as the ability to predict game results and analyse a player's performance has become more valuable than ever. Using the machine learning approach on historical data, which contains data about the teams and individual player performances, it is possible to identify players that stand out from other players. It is also possible to predict the likely outcome of a game between two basketball teams.

In this paper, we use ML approaches to determine the chances of a team winning a game and which players fall under the category of "outstanding" using the NBA dataset, which contains data up to and including the 2004-2005 season. The structure of the paper is a review of related studies, followed by the methods and techniques used in this study, the results obtained from this study, the conclusion, and the list of references.

Related Works:

Nguyen et al. [1] researched whether a player will be chosen for All-star using Regression and Discrete Classification models. The results showed that scoring is the most crucial aspect for predicting the future performance of a player, whether they will be selected for All-star or not. A total of nineteen variables were used for training, and it was observed that there were problems with the class weightings. To overcome this challenge, under-sampling and over-sampling techniques were proposed as solutions. One of the disadvantages of this research was that other important factors that may affect a player's future performance were not included, such as team tactical style, coach decision, and team chemistry, to name a few. An RMSE of 2,1969 and MAE of 1,6465 was produced for Regression models, a Recall of 0.9368, and a ROC AUC of 0.9152 was produced for the Classification models. According to similar studies, Under-sampling was better than Over-sampling for solving the imbalanced data weighting issue. This is validated in this study.

Cheng et al. [2] conducted a study to predict the outcomes of NBA playoffs using an NBA Maximum Entropy (NBAME) model. The model managed to surpass other classical machine learning algorithms. The main advantage of using a Maximum Entropy (ME) model is that it makes use of little-known facts and no assumptions about the unknown. The ME model is more concerned about the construction of feature functions and the pre-processing of feature values of the data. The problem with Naive Bayes is that there exists a lack of independence between some features used in sports forecasting. Since a minimal training dataset was used, attributes that had continuous

numeric values were converted into discrete ones to train the ME model. This is implemented using K-means clustering for vector quantization. It is famous and one of the most effective unsupervised discretization algorithms in data mining. Each game was described by a vector consisting of twenty-nine features, fourteen features for each team and a label indicating a win or loss for the home team, of participating teams and the outcome of the game. To classify the outcome of a new game as a win or loss for the home team, the "Maximum A Posteriori" decision rule and the category with the highest probability are selected. It was observed that a higher threshold improves accuracy but will only be able to predict fewer games. NBAME outperformed the other ML algorithms (Naive Bayes, Logistic Regression, Back Propagation Neural Networks, Random Forest) on the same feature set for most seasons and some seasons are more challenging to predict than others (due to unforeseeable factors, such as injuries, moral, player attitudes, and events that occur outside of the game itself). The authors concluded that Random Forest was the second-best classifier and Naive Bayes was the worst classifier due to its low accuracy, possibly due to its independence assumption.

Kannan et al. [3] proposed a study to predict a player's future success using biometric data, college statistics, draft selection order, and positional breakdown features. The three binary models that were used and compared were the random forest classifier, logistic regression, and support vector machine, and they were trained separately and on two different datasets. The first dataset was a reduced dataset with only biometric data; the second dataset was the full dataset with biometric data, college statistics, draft selection order, and position breakdown. The goal was to determine if adding the extra features influenced the overall predictive power of a model. The Random Forest classifier outdid the other two models on the reduced dataset, with a 0.72 average recall, precision and f1 score. It was determined that the vertical jump and height of the player were the two biometric features with the highest predictive power of player performance. The Random Forest Classifier can then predict the future performance of players using the entire dataset based on the results. When trained on the entire dataset, the model has a higher recall on predicting no success and a higher precision on predicting future success. It was also shown that a player's draft pick was the most crucial attribute for predicting a player's future success. This research showed that a player's draft pick and past performance in college statistics were the best predictors of their future success in the NBA. It was concluded by the authors that biometrics did not have strong enough predictive power. This study could be improved by probing further into more important biometric features of players so that predictions of performance could be defined by biometrics alone. This study can only accurately predict a player's success once established athletes have been drafted. The methodology should be developed to determine a player's future success even before their career begins using biometric data alone.

Chen et al. [4] used a hybrid basketball game outcome prediction model to guess the final scores of NBA teams. Data Mining was performed using Stochastic Gradient Boosting, Extreme Machine Learning, K-Nearest Neighbours, Multivariate Adaptive Regression Splines, and eXtreme Gradient Boosting (XGBoost). The 2018-2019 NBA season dataset was used for this experiment, with a total of two thousand four hundred and sixty game points analyzed. The features chosen for training were generated from basketball statistics that are based on game-lag information. Predictions were made using ten prediction models: five are single-stage models, and the remainder are two-stage models. It was concluded that the T-XGBoost model using a game-lag of four had obtained the highest prediction accuracy from the ten prediction models that used between two and six game-lags' information, and a game-lag of four was the most suitable. The study also classified six statistics as necessary based on the four game-lags. These statistics are average defensive rebounds, offensive rebounds, two-point field goal percentage, assists, free throw percentage, and three-point field goal attempts. This study managed to produce a model with reasonable accuracy; however, it can be

improved. One approach is to use NBA data for multiple seasons instead of one season. This allows for other features to be selected. Another approach is using datasets consisting of matchups, the intensity of a team's game schedule, and low-scoring games can also be used to improve the performance or accuracy of the proposed model used in this study.

Methods and Techniques:

Game outcome predictions:

The game prediction part of the project aims to achieve the following goal: Given two teams, determine which team has a higher probability of winning a game between these teams. It aims to achieve this by comparing, contrasting, and analysing multiple machine learning algorithms using defined metrics and implementing the best-suited algorithm for the task.

The datasets used for NBA game outcome prediction were obtained from publicly available NBA statistics from [5]. The first step in applying a machine learning algorithm to any dataset is to get the features of interest and to ensure that the data is structured in a way that makes it accessible for the model. The data set being used is the "teamseason.txt". This text file contains offensive and defensive metrics for every NBA team since the league's inception in 2005. The dataset does contain wins and losses for each team. This represents the probability of a team winning a game in that season. For a supervised machine learning algorithm, the win rate of a team can be considered the label for the algorithm. The feature vector will be split between fifteen offensive and fifteen defensive metrics. These metrics, taken alongside a pace metric, give a size thirty-one feature vector for the supervised machine learning model.

Data pre-processing:

Data for teams before 1976 are removed due to the lack of relevant metrics and that the NBA and ABA were separate leagues. The first step of data pre-processing is to extract a win rate from each team in the dataset. This is done by taking the column of the total amount of wins of each team and dividing it by the total amount of wins and losses. This is done for all teams and is stored in a separate dataset. The next step involves calculating offensive and defensive "3 points made" for teams from 1976 to 1978. The average win rate, o3pm and d3pm of the entire dataset is calculated. The formula for the total offensive and defensive 3 points made is given as follows:

o3pm/d3pm = avgWinrate + (1 + (teamWinrate - avgWinrate)/avgWinrate)

This formula calculates the weight that the 3 pm score had on a team's win rate, adds the average 3 pm value, and ensures that every team has a 3 pm score. All the unnecessary columns are dropped from the dataset (win, lost, team, year, leag).

The dataset is split into an 80:20 train-test ratio. The data is shuffled before splitting, and the seed of the pseudorandom number generator remains fixed so that each algorithm can be compared in the same run. The dataset is then normalised, and scaling is done to ensure no mean and unit variance. By altering the range of the feature values to a lower scale, the range differences can still be maintained, and no initial feature has dominance. Scaling is done after the dataset is split into training and test sets, so scaling parameters are learnt from the training data. After that, it is applied to the test data.

A supervised regression algorithm will be applied to determine the probability of a team winning based on the 31-attribute feature vector. Linear Regression was considered initially, but since the algorithm is too simple, it would likely underfit the training data due to its large bias. Thus, we opted

not to use this algorithm. The two regression algorithms which were elected: were multi-layer perceptron [10] and Support vector machine [6]

The multi-layer perceptron regression model uses a multi-layer perceptron ANN that trains using backpropagation and no activation function in the last layer. Square error is used as the loss function. The neural network consists of three hidden layers with 100 nodes each. The activation function of the hidden layers is hyperbolic tan. And the lbfgs or stochastic gradient descent optimisers are used over the Adam solver due to the small size of the dataset. Mini batches are not used with the lbfgs solver. The SVM model uses an rbf kernel. The C value is set to 1e3 to minimise error, while the gamma value is set to 1e-8, so there is less curvature in the decision boundary.

Player outlier detection:

The datasets used for NBA player outlier detection were obtained from publicly available NBA statistics from [5]. Outlier detection was carried out on datasets containing player data for regular seasons, playoffs, and all-star games [5]. Most of these datasets had information over the different years and some on the total statistics throughout a player's entire career. To obtain the outstanding players over the previous years, information from each year was extracted and evaluated to determine outliers for that particular year. An outlier graph for each year was plotted. For datasets that had information over a player's entire career, a single extraction of outliers was done, and only one outlier graph was plotted.

The main aim here is to determine the outstanding players based on the NBA statistics for the 2004-2005 NBA season. These statistics include statistics captured from 1949 to 2004. Outstanding players form part of the outliers in the dataset. Our approach to discovering the outstanding players involved using a scatter plot to visually display outliers. This scatter plot is plotted in a 2-dimensional plane. Observing the data, it was discovered that the number of features was too great to be represented in a 2-dimensional plane. Thus, we required some form of dimensionality reduction. We chose to use Principal Component Analysis [7] to reduce dimensionality because it is the most common and simplest way to reduce the dimensionality of large datasets with many features.

Principle Component Analysis:

The numerical features of the dataset were extracted. We used a Standard Scaler to normalise all the features in the dataset so that they are within similar ranges. This ensured that features with larger numerical values would not be favoured over smaller ones. The Standard Scalar for each feature value is calculated as follows: $z = (x - u)/s$

x is the numerical value of the feature, u is the mean of the training samples, and s is the standard deviation of the training samples. The following steps [6] are taken afterwards to determine the principal components:

- A covariance matrix is created to determine the correlations
- Eigenvectors and eigenvalues within the covariance matrix are used to determine the principal components
- A feature vector is created to decide which principal components to retain
- 2 principal components required are returned in this investigation so that a 2-dimensional scatter plot graph can be plotted.

To obtain the scatter plot we place the principal component 1 on the x-axis and principal component 2 on the y-axis.

When determining an outstanding player, careful consideration must be taken depending on three different player positions in basketball (Forward, Centre and Guard). It was found that outstanding centres had a high value for the

While outstanding forwards, Principal Component 1 and Principal Component 2 had a high value for Principal Component 1 but not for Principal Component 2, and Guards had a high value for Principal Component 2 but not for Principal Component 1. To determine the outstanding players, we had to determine the players with the highest Principal Component 1 values, the highest

Principal Component 2 values and highest values for Principal Components 1 and 2. Two methods were explored to determine outstanding outlier players: one using K-Nearest Neighbours [8] and the other using K-Means [9].

K-Nearest Neighbours:

The Euclidean distance between the three nearest neighbours of each player plotted on a scatter plot was calculated and stored. The mean of the three distances was calculated for each player and if this mean was greater than a threshold value of 1, the player was an outstanding outlier player. These players were marked as blue on the scatter plot. Furthermore, a list of outlier indices from a dataset was recorded so that it would be possible to traverse the full dataset and obtain the actual names of the outstanding players.

K-Means Clustering:

K-Means Clustering [9] was used in correlation with Principal Component Analysis to determine outliers. The cluster size was selected as one and was used to group all the data into a single cluster from a central point of this cluster is determined. The approach used involved calculating the coordinates of the centre value of the clustered PCA component points, and then finding the individual distance from this point to every other point on the scatter plot. These distance values were then evaluated to determine if the corresponding points were outliers or not, using the evaluation that If (Distance < x), then x is an outlier. Where x is the maximum PC1 and Distance refers to the distance of the point from the centre value of the clustered PCA component points. Evaluation of the graph indicated that points to the right of the centre point, indicating a high PC1 value, gave us outstanding players. Points to the left of the centre point gave us players that did not perform well. Likewise, points above the centre point, indicating a high PC2 value, gave us outstanding players. Points below the centre point gave us players that did not perform well. Initially, the approach was to isolate the values to the right of the centre and above the centre to find players with high values for both these principal components. After evaluating the data and the players classified as outstanding and those who were not, it became apparent that players classified as outstanding were players who played in the 'Centre' position. This position involves 'attacking' and 'defensive' features. Therefore, we concluded that players with very high PC1 or PC2 values should be considered outliers. We considered the extremely high PC1 and PC2 values when evaluating the outliers. Each clustered PCA component point had its PC1 and PC2 values evaluated, with regards to the extremely high PC1 and PC2 values, to determine whether they are classified as an outlier or not.

Results:

Outstanding Player Detection:

It was discovered that the PCA and K-Means approach was more sensitive to the positions of the players, and it yielded more relevant information about the outstanding players. In contrast, the PCA and KNN approach detected the top performers as well as the poor performers as 'outstanding'. This approach did not evaluate each datapoint on its own, but rather it also took into account the datapoints in the vicinity of the main point, which is why these observations were made. The KNN approach tended to determine players who were above average as 'outstanding'. Given these observations, it was concluded that the PCA and K-Means approach was the more suitable algorithm for this dataset. Figures 1 and 2 are scatter plots of the PCA components for NBA Regular Seasons in general using the KNN approach and K-Means approach respectively.

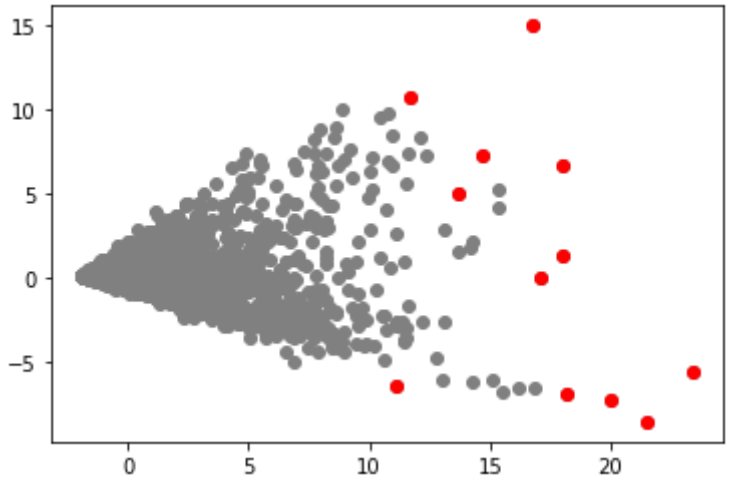

Fig. 1. Scatter plot of PCA components for NBA Regular Seasons in general, using KNN approach

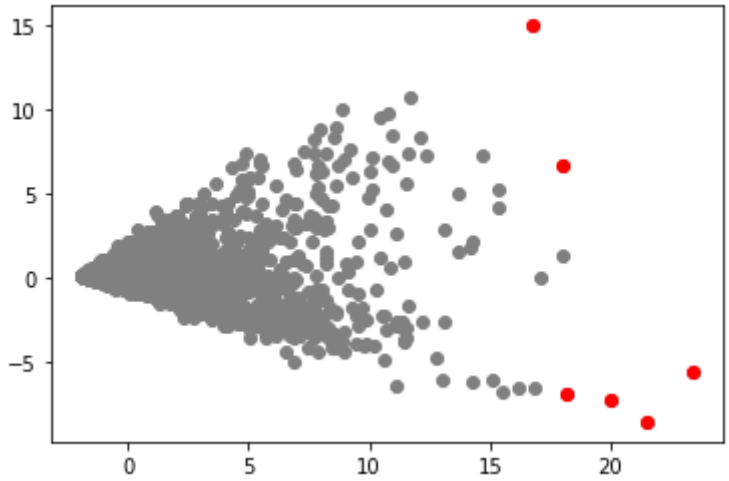

Fig. 2. Scatter plot of PCA components for NBA Regular Seasons in general, using K-Means approach

If we compare the scatter plots of the outstanding players for the NBA playoffs in general, a much better understanding of the conclusion is presented, as figures 3 and 4 show.

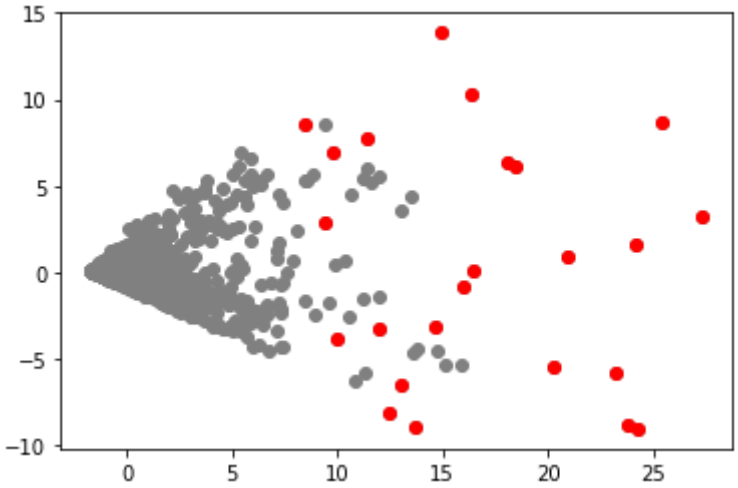

Fig. 3. Scatter plot of PCA components for NBA Playoffs in general, using KNN approach

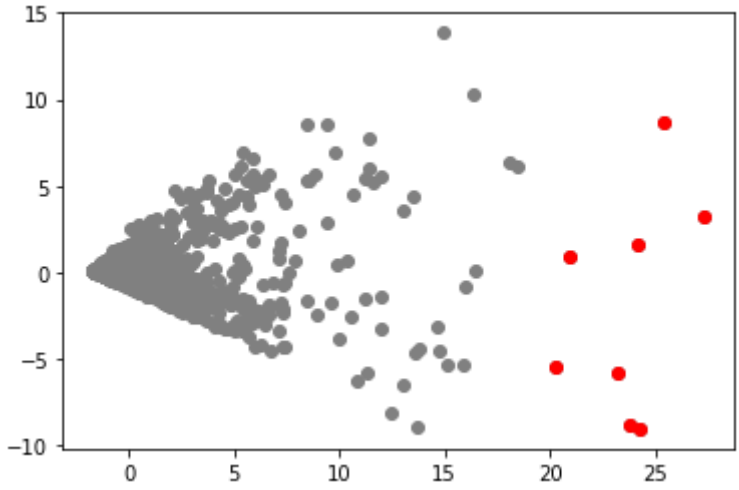

Fig. 4. Scatter plot of PCA components for NBA Playoffs in general, using K-Means approach

From figures 3 and 4, the operation of the nearest neighbour's algorithm can be clearly observed due to the spread and groupings of the datapoints.

Game Prediction Results:

The following metrics were used to determine the performance and accuracy of each model and were obtained via Scikit-Learn's metrics library.

- Max Error

  This is taken as the worst-case error between a predicted value and the correct value.

$$MaxError\ (y, \hat{y}\ ) = \max(\ |y_i - \hat{y}_i\ )$$

- Explained Variance Score

  This is the proportion of variance that is accounted for.

$$ExplainedVariance\ (y, \hat{y}\ ) = \ 1 - \frac{Var(\ y - \ \hat{y}\ )}{Var\ (\ y\ )}$$

- Mean Absolute Error

  This is a measure of observed errors between paired observations

$$MAE\ (\ y, \hat{y}\ ) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (\ |y_i - \ \hat{y}_i\ |)$$

- Mean Squared Error

  This is the average squared difference between predicted values and expected values.

$$MSE\ (\ y, \hat{y}\ ) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (\ |y_i - \ \hat{y}_i\ |)^2$$

- Median Absolute Error

  This is the median of absolute values between the predicted and expected values

$$MedAE(y, \hat{y}) = median(|y_1 - \hat{y}_n|, \ldots |y_n - \hat{y}_n|)$$

- R2 Score

  This measures the proportion of variance due to the independent variables of the model and is considered an indication of how well the model fits the dataset.

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

With all these metrics, a good understanding of each model's performance was gained, and the comparative metrics are shown in figure 5.

```
Algorithm: MLPRegressor

        Prediction Metrics:
                Max Error:  0.168
                Explained Variance Score:  0.867
                Mean Absolute Error:  0.045
                Mean Squared Error:  0.003
                Median Absolute Error:  0.039
                R2 Score:  0.867

Algorithm: SVR

        Prediction Metrics:
                Max Error:  0.21
                Explained Variance Score:  0.823
                Mean Absolute Error:  0.053
                Mean Squared Error:  0.004
                Median Absolute Error:  0.046
                R2 Score:  0.822
```

Fig. 5. Comparative metrics of MLPRegressor and SVRegressor

Observing these results, the conclusion becomes clear that the MLP model is the better model for this dataset. Some tests were done on random teams, predicting the outcome between them. Table 1 shows the randomly selected teams the predictions between them.

	Randomly Selected Teams	Prediction Results
First Test	Team 1 : Dallas Mavericks Year : 1995 Team 2 : Seattle Supersonics Year : 1997	Model - MLP Regression Win Probability of Team 1: 0.514285821096501 Win Probability of Team 2: 0.6036824030290366 The probability of Team 1 beating Team 2 is: 0.46001828137715606 The probability of Team 2 beating Team 1 is: 0.539981718622844 The winner is Team 2 : Seattle Supersonics
Second Test	Team 1 : Toronto Raptors Year : 1996 Team 2 : Seattle Supersonics Year : 1985	Model - MLP Regression Win Probability of Team 1: 0.514285821096501 Win Probability of Team 2: 0.6642197490165557 The probability of Team 1 beating Team 2 is: 0.43638811231682545 The probability of Team 2 beating Team 1 is: 0.5636118876831745 The winner is Team 2 : Seattle Supersonics
Third Test	Team 1 : Philadelphia 76ers Year : 1992 Team 2 : Phoenix Suns Year : 1990	Model - MLP Regression Win Probability of Team 1: 0.38472131235272844 Win Probability of Team 2: 0.7945188606069719 The probability of Team 1 beating Team 2 is: 0.3262450866027916 The probability of Team 2 beating Team 1 is: 0.6737549133972083 The winner is Team 2 : Phoenix Suns

Table 1. Results of 3 separate game predictions with randomly selected teams

Conclusion:

This project had the end goal of using machine learning techniques to determine outstanding players and predict the outcome of a hypothetical game between two teams. For the outstanding player detection, two approaches were used – KNN and K-Means algorithms. It was determined that the best method was the PCA and K-Means approach. It was more accurate and detected less noisy data in comparison to the KNN approach. For game outcome prediction, MLP and SVM methods were considered, and it was concluded that the MLP model was the better of the two. It had better model performance metrics and provided more logical results than the SVM. By observing these results and comparing these methods, it is clear to see that machine learning algorithms can perform well on real-life datasets in fields such as sports sciences.

References:

1. Nguyen, N.H., Nguyen, D.T.A., Ma, B. and Hu, J., 2021. The application of machine learning and deep learning in sport: predicting NBA players' performance and popularity. Journal of Information and Telecommunication, pp.1-19.
2. Cheng, G., Zhang, Z., Kyebambe, M.N. and Kimbugwe, N., 2016. Predicting the outcome of NBA playoffs based on the maximum entropy principle. Entropy, 18(12), p.450.
3. Kannan, A., Kolovich, B., Lawrence, B. and Rafiqi, S., 2018. Predicting National Basketball Association Success: A Machine Learning Approach. SMU Data Science Review, 1(3), p.7.
4. Chen, W.J., Jhou, M.J., Lee, T.S. and Lu, C.J., 2021. Hybrid Basketball Game Outcome Prediction Model by Integrating Data Mining Methods for the National Basketball Association. Entropy, 23(4), p.477.
5. Basketball Reference.Basketball Stats and History Statistics, scores, and history for the NBA, ABA, WNBA, and top European competition. Available: basketball-reference.com
6. Cortes, C. and Vapnik, V., 1995. Support-vector networks. Machine learning, 20(3), pp.273-297.
7. Jaadi, Z. (2019). A Step by Step Explanation of Principal Component Analysis. [online] Built In. Available at: https://builtin.com/data-science/step-step-explanation-principal-component-analysis.
8. Mucherino A., Papajorgji P.J., Pardalos P.M. (2009) k-Nearest Neighbor Classification. In: Data Mining in Agriculture. Springer Optimization and Its Applications, vol 34. Springer, New York, NY. https://doi.org/10.1007/978-0-387-88615-2_4
9. Lloyd, Stuart P. "Least squares quantization in PCM." Information Theory, IEEE Transactions on 28.2 (1982): 129-137.
10. Haykin, S., 1994. Neural networks: a comprehensive foundation, Prentice Hall PTR.