Prediction of NBA All-Stars using supervised learning

Mario Medel & Joseph Coppeta

NC State: Department of Computer Science
{mamedelp, jncoppet}@ncsu.edu

1 Background

1.1 Problem

The goal of this project is to to create an algorithm/model that is proficient at predicting NBA All-Stars based on statistical data from their first three (or more/less depending on our model) seasons played. In order to do this, we took into consideration a vast variety of variables such as draft position, offensive statistics such as points, offensive rebounds, assists, turnovers, field goal percentage and free throw percentage, defensive statistics such as defensive rebounds, blocks and personal fouls committed, team coach, and team record. Given that steals, blocks, and turnovers were not official NBA statistics until the 1973-74 season, we elected to restrict our dataset from 1973-2004. At first, we planned to predict an MVP but due to the low probability of that happening (only one MVP for all available players), we instead look to predict all of the NBA all stars (14 total per season). Our goal is to have a model that improves as more data is added, or in other words; as the players complete more games. Another method which we are considering was to turn the MVP voting into a regression problem as mentioned in [2]. MVP voting occurs through a committee of 100 voters. A vote for first place gives the player 10, a vote for second place gives the player 7 points, and a vote for third to fifth gives one point. In the end, the player with the most points is the selected MVP. We could use the vote distributions in a regression problem in order to predict the MVPs for a given season. This could be used as a means to rank the players based on their predicted scoring for MVP voting.

1.2 Literature Survey

Modelling the NBA to make better predictions

In 'Modeling the NBA to make better predictions' [1], we see an interesting approach to a similar/related problem statement. The problem statement for this thesis was to predict the individual outcomes of games rather than making predictions that rarely occur, such as MVP award (one out of many players) or championship winners (one out of 30 teams). One of the main reasons the author decided to tackle this problem statement instead is because of amount of data that will be available to the prediction model, with more data there will most likely be better predictions. There are also more games played associated with their outcomes which may be easier to analyze (win or loss). We plan to use a similar approach to tackle the All-Star prediction problem. By taking the outcomes of individual games and building a model that is able to predict the outcomes of future games, we can tie these predictions and map them to individual players and give them a general ranking. Ranking players based on seasonal performance and predicted team wins may be the best approach in building the final All-Star teams. In doing so, we are not only making individual player predictions but also team-wide predictions.

The clutch performance metric as well as the team-to-team synergies were the two features and while they added value, it is also stated that their additions are marginal when combined with simpler features. Despite this, it was found out that these team-to-team synergies, a measure of their tendencies and specific patterns of play, were quite useful in making predictions. For example, a team that is prone to 3pt shooting versus a team with poor 3pt shooting defense, in his model, heavily favors the team that is more prone to shoot. He concludes this thought by offering that it appears "very likely that even more features could be created that better measure these synergies," [1]. Evidently, there is still more work that can be done and still many more team-to-team explorations that are promising insofar as yielding positive results.

A Hybrid Machine Learning Model for Predicting USA NBA All-Stars

In this academic journal [2], the authors are attempting to solve the exact same problem statement as we are. It goes on to present the current prediction models that are used within the realm of sports predictions and mentions Monte Carlo Simulation. Monte Carlo Simulation is a prediction algorithm which takes into account the outcomes of random variables and then simulating those outcomes in order to make a predictions (or an expected outcome). One of the downsides to using Monte Carlo Simulation is the computational expense on a system. Although this model can be accurate, it is inaccessible to the average person. Our goal is build a model that can run on most devices and make a close to accurate prediction as Monte Carlo Simulation.

One sub problem pointed out by this academic journal is how to measure accuracy in this sort of model. Using the traditional measure of accuracy (correct / error + correct), many of the models implemented within this context of predicting all-stars will be misleadingly accurate. This accuracy score is not a true indicator of accuracy because of the amount of non-All-Stars in the data. It is easier to predict non-all stars and we will predict them with high accuracy since we only want a team of about 14 all-stars and thus those not in this team will be most likely correctly classified as non-all-stars.

The authors developed a hybrid machine learning model that reached raw accuracy levels nearing 0.9, and specificity results that were even higher. Before constructing this hybrid model, they individually tested with SVC, kNN, Decision Trees, Gaussian Process Classifiers, Random Forest Classifiers, AdaBoost Classifiers, and Multilayer Perception Classifiers, each which performed admirably well. Their combined hybrid model significantly outperformed these however, and turned out to be a very reliable but not yet perfect model. Their discussion about potential improvements mentions that .there are a few other statistical measures that should be included such as popularity and team-related statistics. It was specifically mentioned that this could be done through looking at one's social media following or analyzing their win/loss record or total win shares. Additionally, combining the current model with even more models may potentially increase their true positive percentage, and yield more accurate results.

Application of Distributed Probability Model in Sports Based on Deep Learning: Deep Belief Network (DL-DBN) Algorithm for Human Behavior Analysis

Although we are not going to be using the algorithms mentioned in this article, it is interesting to see another approach to a similar problem statement, particularly, a Deep Learning Deep Belief Network model. This model uses video data in order to detect and extract features to be used within analysis. This model is highly accurate but is not highly performant in practice because of its immense amount of required data/features. Due to this high volume of data, the model may not be accessible to low-performing machines.

An empirical comparison of supervised learning algorithms

In this article, the authors present us with a general comparison of supervised learning algorithms which we will consider to use in our project. The last study done to compare machine learning models at a large scale was STATLOG which was during the 90s, this article will showcase newer models as well refined models (bagging, boosting, SVMs, random forests)[4]. The metric we are interested in is accuracy scores amongst the presented models and the F-score (we want to know how precise and how well the model can recall). It was surprising to see that once Boosted trees undergo Platts' method of calibration, we see higher accuracies than Neural-Networks and random forests which have high accuracies without the calibrations. The boosted trees shows much higher F-scores and accuracy scores amongst all other models. This is a huge bonus because of the simplicity of implementation and the performance of trees. We will consider using Decision tree classifiers and AdaBoosting for our prediction model.

2 Methods

2.1 Approach & Rationale

Measures of Accuracy: $Sensitivity = \frac{(TruePositive)}{TruePositive+FalseNegative}$
Sensitivity scores are a better measure of accuracy for this model than the traditional accuracy scores due to the low probability of the variable we are attempting to predict. Sensitivity in this context will be used to measure the correctly predicted all stars over all the actual all stars. In other words, this is our accuracy score for our All-star predictions.

K-fold cross validation:
It is a great idea to use three seasons of data in our model for training the rest of the seasons as testing models. We can achieve this efficiently by using K-fold cross validation to train the model with all the data and using all the data to validate it by cross folding it.

Models Our first three models will be the primary model that makes a prediction of whether a player will be an all-star or not.

K-Nearest Neighbors This classifier is of interest due to past data that we have available. If we are able identify past players that were identified as All-Stars, then once our model receives a new data point, the model will find the nearest players to their performance and make a prediction based on the nearest distance.

Decision Tree Classifier With AdaBoost [4] Decision Trees with Boosting can provide high accuracy and computational performance. With this knowledge I believe this model will be based at making supervised predictions.

Naive-Bayes Classifier Given all the random variables that may occur throughout an individual game or season, we consider using Naive-Bayes classifier although we may be limited by the assumption of independency amongst random variables. Naive-Bayes classifier may not perform well in this context but we may find surprising results. We may also use this "subpar" classifier to compare against the "better" models.

Regression If we are able to find the distributions of the previous votes for MVP players, then we will use regression to predict the votes for the current season which will be used in conjunction with our other models or may be used independently as a ranking metric.

3 Plan & Experiment

==>UNDERCONSTRUCTION<==

3.1 DataSet(s)

The dataset for our model is quite extensive and covers almost every non-advanced statistic that is tracked in the NBA today. There will be differing groups of statistics measured, which will be player-specific such as individual player statistics, and team-specific statistics relevant to a player's current team. As briefly touched upon in the earlier sections of this report, our dataset includes data in the NBA from the 1973-2004 season, and measures a player's first three years in order to make the prediction. The player-specific statistics will be analyzed through looking at both offensive and defensive data points. The offensive data consists of total minutes played, total points scored, total offensive rebounds, total assists, total turnovers, and field goal percentage. The defensive data consists of total defensive rebounds, total steals, total blocks, and total personal fouls. Over the course of the three seasons, not the total is going to be accounted for but the trend of improvement or disimprovement will be as well. The other portion of our data set consists of the team specific statistics that includes total wins, total loses, average field goal percentage and average rebounding percentage. Our initial dataset contained nearly 20,000 individual seasons worth of data which has not yet been trimmed down. This is still slightly under construction as we have not definitively chosen our total number of data entries, but it is expected that it will probably be in the 1,000 - 10,000 range.

3.2 Hypotheses

3.3 Experimental Design

4 Results

==>UNDERCONSTRUCTION<==

4.1 Results

We expect to see high performance in the AdaBoosted decision tree classifier based on results seen on another study. High performance will be measured based on several metrics. We will compare Accuracy Score (Sensitivity), F-score, and theoretical and experimental runtime. Accuracy Score and F-Score will be highest amongst Decision Tree with AdaBoost and KNN. The best runtime performance will most likely be Naive-Bayes classifier. We consider runtime to be of minor importance when it comes to measuring performance but it should still be considered because we are trying to implement an algorithm that can be access by most devices.

4.2 Critical Evaluation

5 Conclusions

As our experiment has not yet concluded and we have not yet reached a point where we have begun attaining results, thus there is nothing conclusive for us to add. However, based on our literature review, we can make some predictions about what we expect our results to be. Given that our tests will be quite similar to some of the ones in those papers, it is reasonable to assume that our model should finish with somewhere between 0.6-0.9 accuracy. The margin is so large due to the fact that our method is not yet complete, so we are not entirely positive about what it will look like at this point.

5.1 Overall Prediction Accuracy Reflections

Best Model Accuracy (%):
KNN Accuracy Score (%):
Naive-Bayes Accuracy Score (%):
Decision Tree with AdaBoost Accuracy Score (%):

Reflection:

5.2 Lesson Learned & Improvements

References

[1] Albert, A.A.; de Mingo López, L.F.; Allbright, K.; Gomez Blas, N. A Hybrid Machine Learning Model for Predicting USA NBA All-Stars. *Electronics* 2022, 11, 97. https:// doi.org/10.3390/electronics11010097

[2] Keshav Puranmalka. 2013. Modelling the NBA to make better predictions Retrieved October 19, 2022 from https://dspace.mit.edu/handle/1721.1/85464

[3] Rich Caruana and Alexandru Niculescu-Mizil. 2006. An empirical comparison of supervised learning algorithms. *Proceedings of the 23rd international conference on Machine learning - ICML '06* (June 2006). DOI:http://dx.doi.org/10.1145/1143844.1143865

[4] Tianyang Liu, Qizhe Zheng, and Ling Tian. 2022. Application of distributed probability model in sports based on Deep Learning: Deep Belief Network (DL-DBN) algorithm for human behaviour analysis. *Computational Intelligence and Neuroscience 2022* (December 2022), 1-8. DOI:http://dx.doi.org/10.1155/2022/7988844