Midway Report

Malik Majette
Department of Computer Science
North Carolina State University
mamajett@ncsu.edu

Qua Jones
Department of Computer Science
North Carolina State University
qyjones@ncsu.edu

Wenting Zheng
Department of Computer Science
North Carolina State University
wzheng8@ncsu.edu

1 Background

Each year the National Basketball Association (NBA), media, and fans of the sport vote to select the season's 24 All-Stars. 5 players from the Eastern and Western conference are chosen based on the highest number of votes, with fan votes weighing 50%, media votes weighing 25%, and NBA player votes weighing the remaining 25%. These 10 players will represent the starting line-up in the All-Star game for their respective conferences. The remaining 14 players are picked by NBA head coaches based on statistics and player abilities. Although some players, like LeBron James, are chosen year-to-year, for many NBA players there is an amount of uncertainty on whether or not they will be picked. We hypothesize that by using NBA statistics from previous years we can develop a classification-based determination of future All-Stars.

2 Method

The classification-based approach incorporates a decision tree, k-Nearest Neighbors, and Neural Network to determine the 2017-2018 All-Stars. Each model uses All-Star player data from 2014-2017 to a construct a classification for all 2017-2018 players. The misclassification rate for each model is compared to determine the best classifier for this problem. All models will be built on the same attributes (listed in Table 1) and supervised anomaly detection is implemented as well to determine which attributes weigh heaviest in correctly classifying an All-Star. Based on our results data scientists will have more information on the strongest classifier for this domain and NBA players will have a better understanding of if they will become an All-Star and which statistic lines impacted their chances the most. See Table 1.

2.1 Artificial Neural Network

The Artificial Neural Network (ANN) model is an assembly of inter-connected nodes and weighted links inspired by the biological neural networks of the animal brain. The input layer, which consists of a variable number of input nodes, is linked to a hidden layer based on a weight. Each node in the hidden layers applies an activation function to the summation of these inputs that determines a normalized output value to be passed to the next layer. When the algorithm reaches the output layer, the output node is compared to a threshold to classify the result. In supervised ANN training, the network is providing matching inputs and outputs with the intention of the network eventually producing the desired output by adjusting the weights of the connections over time. This technique will be applied to the NBA All-Star data to create a model that can classify NBA All-Stars by using data from previous year All-Stars as a training set.

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

Table 1: Attribute Table

Attribute	Description	Attribute	Description
AGE	Age	3P%	Three-Pointer Percentage
GP	Games Played	FTM	Free-Throws Made
W	Win	FTA	Free-Throws Attempted
L	Loss	FT%	Free-Throw Percentage
MIN	Minutes Played	OREB	Offensive Rebounds
PTS	Points per Game	DREB	Defensive Rebounds
FGM	Field Goals Made	REB	Rebounds per Game
FGA	Field Goals Attempted	AST	Assists per Game
FG%	Field Goal Percentage	TOV	Turnovers per Game
3PM	Three-Pointers Made	STL	Steals per Game
3PA	Three-Pointers Attempted	BLK	Blocks per Game

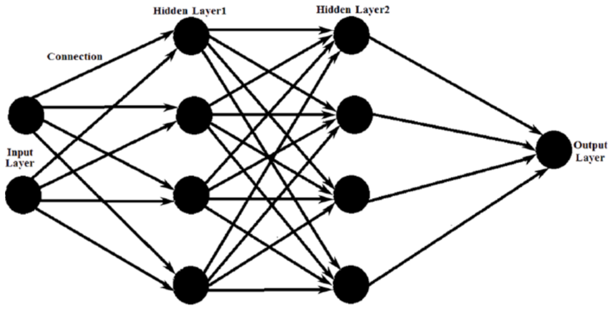

Figure 1: Multi-Layer ANN

2.2 Decision Tree

Decision Trees are a nonparametric supervised learning method for classification and regression. It builds classification models in the form of a tree structure. It breaks down a data set into smaller subsets while simultaneously incrementally developing an associated decision tree. Cross validation is a technique for evaluating models by training models on subsets of data and evaluating on the complementary of the same subset. With k-fold cross validation, data is split into k subsets and the model is trained on k-1 subsets. The model is evaluated on the subset that was not used for training, this process is repeated k times.

2.3 K-Nearest Neighbors

K-nearest neighbors algorithm(KNN) is a nonparametric method used for classification. It is preferred if the input is continuous. The input consists of training dataset in multidimensional feature space. A defined constant(K) number of the nearest neighbors are selected based on the unknown testing input. Classification is obtained by the majority vote of the K nearest points. Generally, larger K reduces the influence of outliers and noise but also increase the risk of underfitting.

The misclassification rate depends on the training data, the K value, the distance type, and if the distance is weighted. We will use 2014-2017 NBA statistics as training data, and use Manhattan and Euclidean distance to measure the distance. Our goal is to find the best combination that minimizes the misclassification rate.

3 Experiment

3.1 Artificial Neural Network

In order to construct a training dataset the NBA statistics from 2014-2017 are pre-processed and concatenated into a multi-dimensional array with each column representing an attribute from the attribute table and each row representing a player. The expected output training dataset is represented

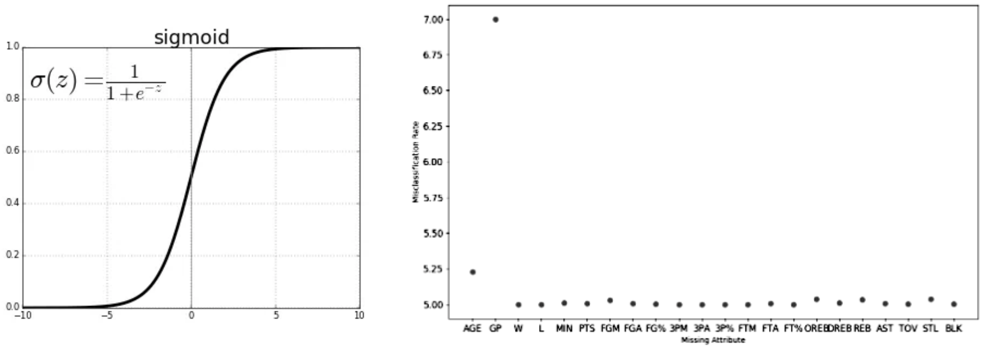

Figure 2: [left]Sigmoid Function and [right]NN Classification Dependency on Attributes

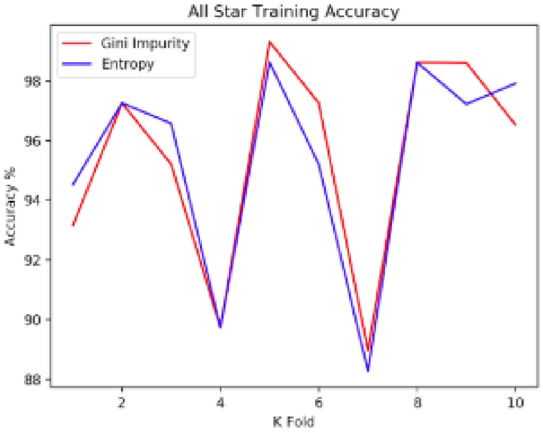

Figure 3: Decision Tree Accuracy

as [0, 1] for non-All-Stars and All-Stars. The neural network model is then constructed with 22 input nodes, a hidden layer of 24 nodes, and an output node. Rectified linear units is chosen as the step activation function for the hidden layer and sigmoid is used as the output activation, its mathematical form is shown below.

Next the model is fitted with the training data to update the weights in the model and learn a classification of an All-Star. To test the accuracy of the neural network the model is evaluated with the 2017-2018 NBA dataset. The resulting misclassification for the base model is 5.00%. With a large attribute set, in order to evaluate the attributes that impact the misclassification rate the most that construction of the model was repeated 22 times with a missing attribute. As seen in the plot below the misclassification rate increases most drastically when the number of games played is removed followed by the age of the player.

3.2 Decision Tree

In order to construct the appropriate training set, statistics from seasons 2014-2017 were combined into a larger set of data to be processed. For classifying All-Stars [0,1] is used to represent non-All-Stars and All-Stars respectively. Two decision tree classifiers were constructed, one based on a Gini Impurity classification and one based on a Entropy classification. 10 fold cross validation was applied to the 2014-2017 dataset for model evaluation. Data from the 2017-2018 season were used for testing and to determine the accuracy of the models. The mean accuracy for the classification models were 96.67% for Gini Impurity and 97.22% for Entropy.

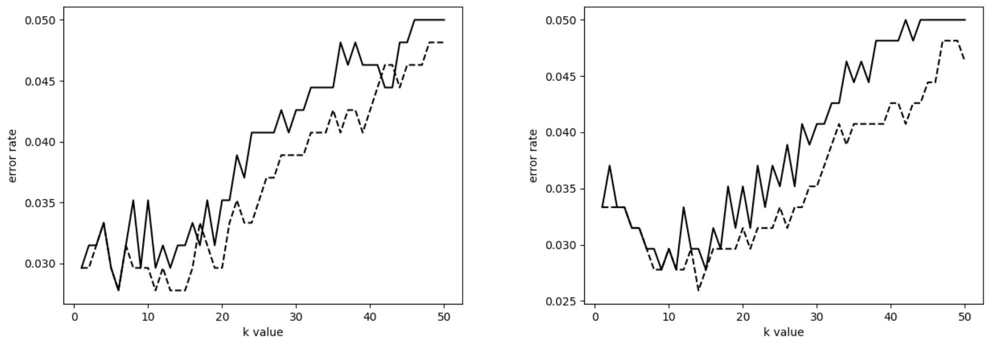

Figure 4: Error rate with 2014-2015[left] and 2015-2016[right]

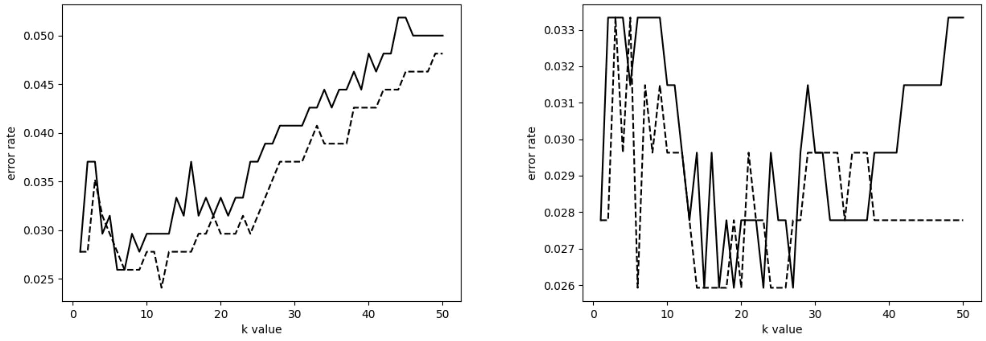

Figure 5: Error rate with 2016-2017[left] and 2014-2017[right]

3.3 K-Nearest Neighbors

There are three-year NBA player data. One dataset consists of all three-year data. Then, we group the datasets by year, so we have the other three training datasets. We will use four training datasets in total. We apply normalization to obtain normalized training and testing datasets in the pre-processing stage. Since the testing dataset is treated as unknown data, the maximum and minimum of the training dataset are used to normalize the testing dataset. This step rescales the range of the attributes in the range of [0, 1]. Then, we use four different datasets to train the KNN classifier with the K value in the range of [1, 50]. The predicted values are in the range of [0, 1] and [0, 0.5] belongs to 'No' class while [0.5, 1] belongs to 'Yes' class. Next, we calculate the error rate by misclassified objects/total number of objects. The following plots represent the error rate based on the K value and distance weight. There are two lines in the graph. The dashed line is the error rate with weighted distance while the solid line is with uniform weigh. See Table 2 to get the minimum misclassification rate in the graph and its K value.

Table 2: Misclassification Rate Table

Error Rate	14-15	15-16	16-17	14-17
Uniform	0.0278, k = 6	0.0278, k = 9	025926, k = 6	025926, k = 15
Weighted	0.0278, k = 6	025926, k = 14	0.024, k = 12	025926, k = 6

3.4 Upcoming Experiments

We used Euclidean distance in the KNN classifier and calculated the error rate based on different training data and distance weight. We will calculate the misclassification rate using the KNN classifier with Manhattan distance and compare them to get the best combination.

4 Conclusion

Under construction.

References

[1] Kim, Keon. (2017) Deep Q-Learning with Keras and Gym. *Deep Q-Learning with Keras and Gym.* keon.io/deep-q-learning/.

[2] Staff, NBA.com (2018) How the NBA All-Star Draft Works. *How the NBA All-Star Draft Works.* www.nba.com/article/2018/01/18/2018-nba-all-star-draft-rules.

Appendix

Previous Plan

Malik Majette: Reinforcement learning algorithm for predicting shooting percentages

Qua Jones: Decision tree algorithm for All-Star classification

Wenting Zheng: kNN algorithm for All-Star classification

Revised Plan

Malik Majette: Finding the best combination of activation function, hidden layer neural nodes, and optimizer that minimizes the misclassification rate.

Qua Jones: Decision tree algorithm with cross validation for All-Star classification.

Wenting Zheng: Finding the best combination of data, k value, distance type, and distance weight that minimizes the misclassification rate.

Overall: Compare research findings to conclude the strongest classifier and why.