Article

Predicting the Outcome of NBA Playoffs Based on Maximum Entropy Principle

Ge Cheng [1], Zhenyu Zhang [1,*], Moses Ntanda Kyebambe [1] and Nasser Kimbugwe [2]

1 Department of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, China; chengge@xtu.edu.cn (G.C.); mntanda@cis.mak.ac.ug (K.M.N.)

2 Department of Information Engineering , Xiangtan University, Xiangtan 411105, China; nkimbugwe@cis.mak.ac.ug

* Correspondence: zhenyuzhang@smail.xtu.edu.cn; Tel.: +86-155-7328-6689

Abstract: Predicting the outcome of a future game between two National Basketball Association (NBA) teams poses a challenging problem of interest to statistical scientists as well as the general public. In this article, we formalize the problem of predicting the game results as a classification problem and apply the principle of maximum entropy to construct NBA maximum entropy (NBAME) model that fits to discrete statistics for NBA games, and then predict the outcomes of NBA playoffs by the NBAME model. The best NBAME model is able to correctly predict the winning team 74.4 percent of the time as compared to some other machine learning algorithms which is correct 69.3 percent of the time.

Keywords: Maximum entropy model; K-means clustering; accuracy; classification; sports forecasting

---

1. Introduction

National Basketball Association (NBA), the highest level Basketball league in the world, was founded in 1946, having a history of 70 years. Now it is among the most professional, marketed, attended games and one of the most popular leagues in the world. NBA enjoys a big following around the world, with many participants anticipating for results, in addition to a multitude of betting companies offering vast amounts of money to gamblers on odds of one team winning another [1,2]. Most participants often place their odds subjectively basing on their personal preference of teams without any scientific basis thus accuracy of the prediction is often very poor. With the rapid advance in science and technology, specifically using sophisticated data mining and machine learning algorithms, forecasting the outcome of a game before the game starts with high precision is highly feasible and of great economic significance to various players in the betting industry.

With the prevalence of global sports competition, experts have began to focus on the historical records of game statistics in a bid to turn the data into useful information since 1950. In the early days, most of forecasts of NBA games just applied simple principles of statistics, which simply combined the technical features of games, then calculating each team's attacking and defensive strength, thus determining the team's overall strength, and then sorting teams according to their overall strength, and finally using the sorted list to predict the outcome of the game [3,4]. However their accuracy is low compared to probabilistic based machine learning methods. As the data for statistical features became more ubiquitous, people began to look for more methods to apply to the large amounts of data thus a vast amount of articles related to the analysis and forecasting of results of sports encounters were published. With advances in statistics and processing power of personal computers, researchers leveraged this power to improve accuracy in prediction. Bhandari et al. [5] developed the Advanced Scout based on PC machine in 1996, which pushed NBA games' data into data mining and knowledge discovery technology field and enabled coaches to find some interesting patterns of the competition of basketball game based on data.

By the end of the 20th century, a variety of machine learning algorithms started to be used by scientists to forecast NBA games. Existing research that has used neural nets and decision trees has a

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

major limitation of limited datasets which leads of overfitting of both the neural network and decision tree models. Consequently the models will perform very well on the training data but a very low performance results on the test dataset [6–8]. The Maximum Entropy model overcomes this limitation by making use of the little known facts and making no assumptions about the unknown. Similarly, support vector machine is limited by it's failure to output a probability value but just a win or loss which makes the results difficult to explain [9]. Lack of independence between some features used in sports forecasting is a major limitation to researches such as [10], that used Naive Bayes method.

Of recent, many scholars have used a variety of probability graph models to simulate games [11–14], and their results are promising. However their major focus is on the difference between the simulation and the real game but do not predict the final outcome of the game and neither do they compute their prediction accuracy. Stekler et al. [15] examined some different evaluation procedures and compared prediction accuracy of some forecasting methods. Haghighat et al. [16] reviewed the use of data mining technology (neural nets, support vector machines, bayesian method, decision trees and fuzzy system) to forecast the results of sports events and evaluated the advantages and disadvantages of each method. However, they did not evaluate the Maximum Entropy method because to the best of our knowledge, this is the first research to apply the Maximum Entropy model to sports forecasting.

The Maximum entropy model is more concerned about the construction of feature functions and the preprocessing of feature values of the data. In this paper, using the maximum entropy principle, we attempt to overcome the feature independence assumption that limits the Naive Bayesian model. We applied the maximum entropy principle to a set of features and established the NBAME model. Then we used the model to calculate the probability of the home team's win of an upcoming game and made predictions based on this probability. Our results show that the prediction accuracy is pretty high when compared with others machine learning algorithms.

The rest of this paper is arranged as follows: In the following sections we describe the Maximum entropy model and K-means clustering. Section 3 gives overview of NBAME model. Section 4 presents the experiment results and compare with other models. Finally, concluding remarks and suggestions for future work are given in Section 5.

2. Methods

Before exploring the use of the entropy-based scheme in NBA predication, we discuss the Maximum entropy model, and the K-means clustering algorithm used to discretize continuous valued attributes.

2.1. Maximum entropy model

The concept of "Information entropy" dates way back since Shannon [17] first put forward the concept of information entropy in 1948, which is the expected value of the information contained in each message. As a measure of random events' uncertainty or the amount of information, the information entropy can explicitly be written as

$$H(p) = -\sum_{i=1}^{n} p_i log_2(p_i) \tag{1}$$

Where H(p) is the information entropy, and p_i is the probability of a random event. Jayne [18] proposed a criterion when reasoning according to some information, we must choose such a set of probability distribution that had maximum entropy and subjected to all the known information; this criterion is known as the "Maximum entropy principle". The maximum entropy principle points out that when we need to estimate the probability distribution of a random event, it must meet all the known conditions, and make no subjective assumptions about unknown conditions. In this case, the probability distribution is most uniform, and the risk of making a wrong prediction is the lowest, and

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

called this distribution as "Maximum entropy model". If we assume that all the models are constraint to the set C, as

$$C = \{p \in P | E_{\tilde{p}} f_k = E_p f_k, i = 1, 2, \cdots, N\} \tag{2}$$

The condition entropy that defined in the conditional distribution $p(y|x)$ is

$$H(p) = -\sum_{x,y} \tilde{p}(x) p(y|x) log p(y|x) \tag{3}$$

Then the model that has the maximum entropy $H(p)$ is the maximum entropy model.

Maximum Entropy model is a good-performance adaptability and flexibility excellence statistical model, making probability estimates to the problem, which is suited to solve the problem of classification. Unlike other models, the Maximum Entropy Model does not suffer from the effect of related features since it has an internal mechanism of dealing with features that are related to each other, thus it gives a higher accuracy when dealing with data that has several related features. The Maximum entropy principle has been widely applied to all kind of areas at the present time. Tseng and Tuszynski [19] gave several examples of applications of maximum entropy in different stages of drug discovery. Xu et al. [20] proposed a continuous maximum entropy method to investigate the robust optimal portfolio selection problem for the market with transaction costs and dividends. Berger et al. [21] described statistical modeling based on maximum entropy and used the model to solve natural language processing problems. Nigam et al. [22] proposed the use of maximum entropy techniques for text classification. Phillips et al. [23] studied the problem of modeling the geographic distribution of a given animal or plant species by maximum-entropy techniques.

2.2. *K-means clustering*

Clustering is an unsupervised learning method, the algorithm need not be given labeled data. Jain [24] provided an overview of clustering algorithm development and application. K-means clustering is a method of vector quantization and originally from signal processing. The standard algorithm was first proposed by Stuart Lloyd in 1982 [25], the central concepts of it was that portion n observations $\{x_1, x_2, \cdots, x_n\}$ into k clusters in which each observation belongs to the cluster with the nearest mean. Nowadays it is more and more popular as a kind of clustering method in data mining field. Münz et. al [26] presented a novel flow-based anomaly detection scheme based on the K-means clustering algorithm. Kanungo et. al [27] presented a simple and efficient implementation of K-means clustering algorithm and so on.

Algorithmic steps for k-means clustering:

1. Let $\{x_1, x_2, \cdots, x_n\}$ be the set of data points and $V = \{v_1, v_2, \cdots, v_c\}$ be the set of centers.
2. Randomly select "c" cluster centers and calculate the distance between each data point and cluster centers.
3. Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.
4. Recalculate the new cluster center using: $v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$ where, c_i represents the number of data points in ith cluster.
5. Recalculate the distance between each data point and new obtained cluster centers.
6. If no data point was reassigned then stop, otherwise repeat from step 3.

As K-means clustering is one of the most effective methods to discretize features [28], so we used K-means clustering to discretize values of each feature.

3. Maximum Entropy principle in NBA playoffs' prediction

In this section, we describe basic technical features described in each game and apply the maximum entropy principle to build the National Basketball Association Maximum Entropy (NBAME) model

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 September 2016 doi:10.20944/preprints201609.0103.v1

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

3.1. Basic technical features

The outcome predicting problem is formalized as a classification problem, where the game outcomes belong to exactly one of two categories. Each game is described with record consisting of 29 features, which are related to the participating teams and the outcome of the game. Complete statistics feature set with abbreviations can be seen in Table 1.

Table 1. Basic technical features used by our model.

Feature	Abbreviation	Feature	Abbreviation
Field Goal Made	FGM	Field Goal Attempt	FGA
Three Point Made	3PM	Three Point Attempt	3PA
Free Throw Made	FTM	Free Throw Attempt	FTA
Offensive Rebounds	Oreb	Defensive Rebounds	Dreb
Assists	Ast	Steals	Stl
Blocks	Blk	Turnover	TO
Personal Fouls	PF	Points	PTS

The statistics shown in Table 1, basic technical features, were used since they are common to basketball and the typical fan understands what statistic represents.

3.2. NBAME model overview

Before building the NBAME model, we should construct the feature function. Choice of the feature function is vital for performance of the maximum entropy model, which affects the structure of the optimal probability model directly, and it is also gives the maximum entropy model the most superiority over other models. The choice of feature function is flexible, which enables the designer to make full use of the known facts of all kinds of information to improve the performance of the model. In general, a feature function is a binary function of the form $f(x,y) \in (0,1)$. Considering the NBAME model, we consider the NBA statistical probability model of training data set$(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N), x_i = (x_i^{(1)},x_i^{(2)},\cdots,x_i^{(28)}) \in R^{28}$ and is the features of the data sets for each match, y_i=0 or 1 is corresponding to the result of the match whether home team wins. So according to dataset features x_i and the outcome of the game result y_i for each game in training data set, we can define the feature function as:

$$\mathrm{f_k(x,y)} = \begin{cases} 1, & (x_i = (x_i^{(1)},x_i^{(2)},\cdots,x_i^{(28)})) \wedge (y = y_i) \\ 0, & otherwise \end{cases} \tag{4}$$

Where the $k \in K$, $K = |x^{(1)}| * |x^{(2)}| * \cdots * |x^{28}|$.

After constructing the feature functions, we can build the NBAME model by maximum entropy model. We count the games with the same features x_i, and the same outcome of the game y_i in the training data set, and then divide them by total number of training data set N. We get the empirical distribution of joint probability distribution $\tilde{p}(x,y)$.

$$\tilde{p}(x,y) = \frac{1}{N} * \text{number of times that } (x,y) \text{ occurs in the sample} \tag{5}$$

For each feature function f_k, the expectation with the empirical probability distribution of joint probability distribution $\tilde{p}(x,y)$ is:

$$E_{\tilde{p}} f_k = \sum_{(x,y)} \tilde{p}(x,y) f_k(x,y) \tag{6}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 September 2016 doi:10.20944/preprints201609.0103.v1
Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

We make calculation of the same feature attribute data feature x, and then divided them by the total number of training dataset to get the empirical distribution of marginal probability distribution $\tilde{p}(x)$,

$$\tilde{p}(x) = \frac{1}{N} * \text{number of times that } (x) \text{ occurs in the sample} \tag{7}$$

The expectations of feature function f_k relative to the model $p(y|x)$ and empirical distribution of marginal probability distribution $\tilde{p}(x)$ is:

$$E_p f_k = \sum_{(x,y)} \tilde{p}(x) p(y|x) f_k(x,y) \tag{8}$$

According to maximum entropy model, the classification problem becomes the optimal solution problem that meets a set of constraint conditions now, namely:

$$P = \{p | E_{\tilde{p}} f_k = E_p f_k, i = 1, 2, \cdots, N\} \tag{9}$$

$$p^* = \arg\max_{p \in P}(H(p)) \tag{10}$$

where

$$H(p) = -\sum_{x,y} \tilde{p}(x) p(y|x) logp(y|x) \tag{11}$$

We transformed the problem of constrained optimization into unconstrained optimization of its dual problem by Lagrange multiplier method:

$$p^*(y|x) = \frac{1}{\pi(x)} \exp(\sum_{i=1}^{N} \lambda_k f_k(x,y)) \tag{12}$$

where the $\pi(x)$ is a normalization factor:

$$\pi(x) = \Sigma_y \exp(\sum_{i=1}^{N} \lambda_k f_k(x,y)) \tag{13}$$

Parameter λ_k can be perceived as the weight of feature function $f_k(x,y)$ the process of maximum entropy algorithm's learning is the process of adjusting the λ_k. When solving the parameter λ_k, we could not obtain it in an analytical way but by a numerical calculation method , the most popular being the Generalized Iterative Scaling(GIS) [29]. In this paper, we use the Generalized Iterative Scaling method to calculate parameter λ_k.

4. Results

In order to test the performance of NBAME model, after collecting and preprocessing the games' statistics, we turn to the problem of predicting the outcomes of NBA playoff games for each season individually during 2007-08 season and 2014-15 season. We make experiments on dataset by NBAME model and some other machine learning algorithm.

4.1. Data Collection and Preprocessing

We created a crawler program to extract the 14 basic technical features of both teams and home team's win or lose from http://www.stat-nba.com/ , collected a total of 10271 records for all games for seasons ranging from 2007-08 season to 2014-15 season, and stored them into a MySQL database.

After the original data set was obtained, we cleaned it using Java. First, we combined the two teams' 14 basic technical features of the same game into a single record for the game. The features of a game therefore contained 28 basic technical features and win or lose of the home team. Secondly,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 September 2016 doi:10.20944/preprints201609.0103.v1

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

we calculated the mean of each basic technical features respectively from recent six games for both sides before the game started (In every season, if teams didn't have 6 games before the game started, then we took the mean of the feature data that had happened before the game. The features of the first game of each season could not be predicted, so we removed the record of the day contains teams' first game for each season), as the basis of predicted to each technical features for the coming game, and used to predict the outcome of it.

Table 2. Sample features raw values obtained from stat-nba.com website

Features	FGM	FGA	3PM	3PA	FM	FTA	Oreb	Dreb	Ast	Stl	Blk	TO	PF	PTS
Features'	32	79	6	24	18	24	8	28	17	10	2	18	15	88
values of	45	87	9	24	8	11	5	32	32	8	3	14	23	107
last	33	85	7	23	22	29	9	36	22	10	4	12	21	95
six games	33	83	6	23	12	15	14	28	22	6	4	15	18	84
for	48	85	8	23	10	14	12	31	29	9	6	13	20	114
home team	44	80	7	19	14	18	7	35	25	9	8	14	16	109
Average	39.17	83.17	7.17	22.67	14.00	18.50	9.17	31.67	24.50	8.67	4.50	14.33	18.83	99.50

Table 2 shows home team's least six games' basic technical features obtained from website and the average values for coming game. We could calculate the basic technical features for every coming game in the same way. Table 3 shows a sample record of the features and the true outcome for the first game in 2014-12-31.

Table 3. A sample record experimental dataset obtained by getting averages of previous six games.

Features	Values	Features	Values	Features	Values	Features	Values	Features	Values
FGM_h	39.17	$Dreb_h$	31.67	$FGMa$	41.00	$Dreba$	31.17	Win_h	1
FGA_h	83.17	Ast_h	24.50	$FGAa$	82.33	$Asta$	22.17		
$3PM_h$	7.17	Stl_h	8.67	$3PMa$	7.50	$Stla$	6.67		
$3PA_h$	22.67	Blk_h	4.50	$3PAa$	18.33	$Blka$	4.00		
FTM_h	14.00	TO_h	14.33	$FTMa$	21.00	TOa	16.33		
FTA_h	18.50	PF_h	18.83	$FTAa$	27.83	PFa	22.17		
$Oreb_h$	9.17	PTS_h	99.50	$Oreba$	10.33	$PTSa$	110.50		

As shown in Table 3, each training example is of the form (x_i, y_i), which corresponds to the statistics and output of a game in a particular match. x_i is an 28 dimensional vector contains the input variables and y_i indicates whether the home team won $(y_i = 1)$ or lost $(y_i = 0)$ in that game. The first 28 columns indicate the basic technical features for each team as obtained by computing an average of the previous six games played by the corresponding team. The 29th column is the actual outcome of the game corresponds to the predicted game, labeled as "Win_h" takes on only two values: 1 or 0; a value of 1 indicates that the home team won and 0 otherwise. We use these basic technical features dataset to train NBAME model by the principle of the maximum entropy and predict the result of the coming game during NBA playoffs for each season.

According to the maximum entropy principle, the NBA maximum entropy (NBAME) model needs to be trained on sufficient amount of training data. However, training in each season is limited thus a possible threat of over-fitting; if there are too many feature functions such that the number of training samples is smaller than the feature functions, the probability distribution model will over-fit resulting into high variance. Consequently, we get a better performance on the training data but low accuracy on testing data. So we needed to discretize each feature's value to reduce variance and avoid over-fitting.

We used K-means clustering for data discretization by the software R. We applied the clustering software package [30], using the Partitioning Around Medoids (PAM) function to cluster the data of each feature. The number of clusters is input parameters, and its value often involves clustering's effect. A crucial choice to make was the number of clusters to be used; Silhouette Coefficient (SC) [31] can be used to solve this problem, which combines condensation degree and degree of separation.

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

It indicated the effectiveness of clustering with a value between -1 and + 1. The greater the value is the better result of clustering. According to this principle, we could try to use some parameters of number of clustering, calculating the SC repeatedly under the condition of different cluster number, we chose the one when the SC is the highest, which corresponded the number of best clusters.

We calculate the SC of the away teams' score when k ranges from 3 to 10 (2 clusters are not enough to distinguish a lot of data obviously). Figure 1 shows the relationship between k value and SC by the K-means clustering to discrete the away teams' score.

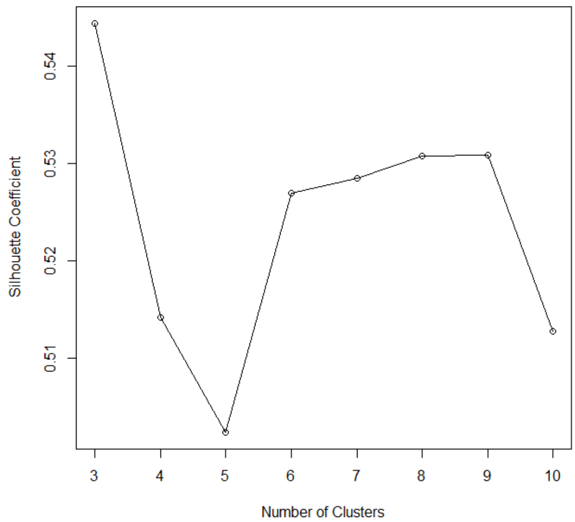

Figure 1. Silhouette Coefficient (SC) with the change of clusters.

As shown in the Figure 1, 2014-15 season after getting rid of the first three days (some of the teams had not played the first game of the season in the first three days of this season, it was not possible to forecast its features' values, here we did not use it as an experimental dataset), the SC value of away teams' score changes as the change of the number of clusters k value from 3 to 10 in 2014-15 season. It is shown that when we take k as 3, the SC 0.545 is greater than when k takes any other values. Thus the cluster number of away teams' score is assumed to be 3. Figure 2 shows the results of discretization of away teams' score, when the SC is 0.545

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 September 2016 doi:10.20944/preprints201609.0103.v1

Peer-reviewed version available at *Entropy* **2016**, *18*, 450; doi:10.3390/e18120450

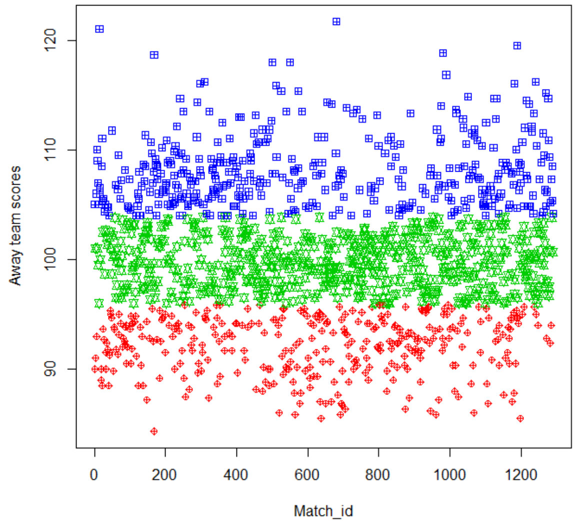

Figure 2. Three clusters for away teams' score

As shown in the Figure 2, away teams' score discrete values after 3 clusters of K-means, the distribution in each cluster is shown. We use k-means clustering to discretize home teams' score values and other basic technical features for each game in the same way. Some samples of the experimental data set can be seen in the Table 4.

Table 4. A sample record experimental dataset obtained by getting averages of previous six games.

Home teams' features													
FGM_h	FGA_h	$3PM_h$	$3PA_h$	FTM_h	FTA_h	$Oreb_h$	$Dreb_h$	Ast_h	Stl_h	Blk_h	TO_h	PF_h	PTS_h
37.63	83.31	7.85	22.66	14.35	19.02	9.50	31.68	24.26	8.82	4.18	14.49	18.65	98.6
37.63	83.31	7.85	18.94	14.35	19.02	7.17	35.01	24.26	6.94	6.50	12.42	22.07	94.48
37.63	84.75	10.74	26.11	17.80	25.44	11.27	32.59	24.26	8.82	6.50	12.42	19.87	106.59
37.63	80.20	7.85	26.11	17.80	22.35	9.50	29.60	22.97	7.90	4.18	15.14	18.65	98.60
37.63	86.12	10.74	34.05	17.80	23.80	12.30	31.68	21.97	9.64	3.36	16.09	22.07	102.39
37.63	77.95	7.85	20.87	17.80	20.76	7.17	30.77	24.26	6.94	4.18	16.09	20.98	102.39
40.85	87.78	7.85	18.94	17.80	25.44	13.39	36.94	19.96	6.94	5.66	12.42	20.98	106.59

Away teams' features														**Home team win**
FGM_a	FGA_a	$3PM_a$	$3PA_a$	FTM_a	FTA_a	$Oreb_a$	$Dreb_a$	Ast_a	Stl_a	Blk_a	TO_a	PF_a	PTS_a	
40.36	82.6	7.77	18.40	20.61	26.54	10.34	32.16	22.40	6.59	4.08	16.19	21.65	108.43	1
36.76	73.84	7.77	21.63	16.85	24.42	8.26	28.68	19.34	9.00	3.76	13.17	21.65	100.06	1
37.73	86.81	5.39	18.40	16.85	20.34	12.48	28.68	21.66	7.76	5.15	12.42	20.45	100.06	1
38.88	79.5	10.49	21.63	15.58	20.34	7.59	32.16	24.13	7.76	5.71	17.36	22.79	100.06	0
35.66	85.13	7.77	21.63	17.90	24.42	10.83	35.50	19.34	7.18	5.71	11.74	19.09	100.06	1
37.73	85.13	5.39	18.40	16.85	20.34	11.48	32.16	20.70	7.18	5.15	12.42	16.32	100.06	1
42.48	86.81	10.49	24.73	16.85	22.41	12.48	32.16	20.70	8.40	5.71	12.42	19.09	108.43	1

As shown in Table 4, the first 14 columns represent the home teams' basic technical features values after k-means clustering discretization. The last column is the home teams' actual win or loses of the game. Others represent the away home teams' basic technical features values after k-means clustering discretization. It is also the final dataset that is applied to train NBAME model and make prediction for the NBA playoffs. We sort them by the date, separate them according to the seasons, and then saved the data for each season to a file, and then train and test NBAME model for each of the season saved in the file repeatedly.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 September 2016 doi:10.20944/preprints201609.0103.v1

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

4.2. The results of NBAME model for predicting the NBA playoffs

We accumulated 14 basic technical features, which are discretized by K means cluster, of both sides and the victory of home from the first game of the season to the coming game to construct NBAME model by the maximum entropy principle, and trained the parameter λ_k by the Generalized Iterative Scaling (GIS) algorithm. Then we applied 28 basic technical features of the coming game into the NBAME model; calculated the probability of the home team's victory in the game $p(y|x)$. As the results for $p(y|x)$ are continuous number, we set up a threshold at 0.5 (meaning that if our model outputs a probability not less than 0.5, we decide that the home team wins, else if the probability is less than 0.5, we decide the home team loses) to decide the winner of the game, meaning:

$$\mathrm{f_k}(\mathrm{x},\mathrm{y}) = \begin{cases} 1(win), & p(y|x) \geq 0.5 \\ 0(lose), & p(y|x) < 0.5 \end{cases} \tag{14}$$

Finally, we compared the decision of the home team to win or lose to the corresponding real competition. If it was the same, then we said the prediction of the NBAME model was right, and we added 1 to the count of correct prediction. Eventually we would get the total number of predicting correctly, and we divided it by the number the data set that we used to test it, which is our model's forecast accuracy. Accuracy was used as performance measure, and it was calculated by the following formula:

$$accuracy = \frac{\text{number of correct predictions}}{\text{total number of predictions}} \tag{15}$$

The accuracy of predicting eight seasons NBA playoff games by NBAME model with 0.5 threshold judgment is shown in the first row of Table 5.

Table 5. Prediction accuracy (in percentages) of NBAME model with different thresholds.

Threshold	2007-08	2008-09	2009-10	2010-11	2011-12	2012-13	2013-14	2014-15
0.5	74.4	68.2	68.3	66.7	69.0	67.1	65.2	69.1
0.6	77.1	74.5	75.0	69.8	73.0	71.4	66.7	70.4
0.8	100.0	80.0	100.0	100.0	100.0	75.0	100.0	100.0

As can be seen from Table 5, we used entire discrete feature set (all 28 game statistic as features) to predict the probability of the home team's victory for the playoffs of each season between the 2007-08 season and 2014-15 season. The prediction accuracy reached as high as 74.4% in 2007-08 season. This shows that the NBAME model by the maximum entropy principle is suitable to forecast the outcome of NBA game, and can achieve the best prediction accuracy.

NBAME model outputs the probability of the home team's winning in the coming game given the coming game's features. The home team would be more likely to win if the model outputted a probability greater than the threshold value. At this point it is important to note setting a high confidence improves the accuracy of our model predictions with a drawback of predicting fewer games. If we set the threshold to 0.6, we shall not be able to make a decision on all the games with output probabilities between 0.4 and 0.6,

$$\mathrm{f_k}(\mathrm{x},\mathrm{y}) = \begin{cases} 1(win), & p(y|x) \geq 0.6 \\ 0(lose), & p(y|x) \leq 0.4 \end{cases} \tag{16}$$

However, predictions based on the few decisions made would be more accurate. To study the effect of varying the threshold values, we used the same training dataset to train the NBAME model, and calculated the probability of the home team's victory for each season, and used the threshold of 0.6 and 0.7 to decide the outcome of coming game, the experimental results are shown in Table 5. We

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 September 2016 doi:10.20944/preprints201609.0103.v1

Peer-reviewed version available at Entropy 2016, 18, 450; doi:10.3390/e18120450

also made Figure 3 for a graphical representation of the number of predicted games and predicted accuracy during 2007-08 season and 2014-15 season with the threshold of 0.5, 0.6 and 0.7

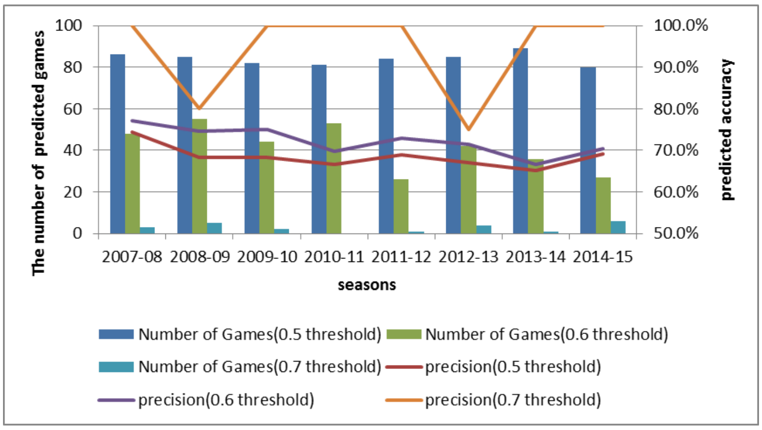

Figure 3. The number and accuracy of prediction with different confidence by NBAME model from 2007-08 season to 2014-15 season playoffs

As shown from Figure 3, when adjusted the confidence, the games for which we could make a decision are less than the total of playoff games of each season. For example, the number of prediction decreased from 86 to 48 when we increased the threshold to 0.6 in 2007-08 season, however, the correct rate of prediction improved to 77.1 percent. While used 0.7 as the threshold, we got the 100 percent for many season payoffs, even though we just could make decision for few games. It shows that we could trade the number of games for which we can make a prediction for the benefit of improved prediction accuracy which can be of great commercial value.

4.3. Comparison of NBAME model with some selected existing machine learning algorithms.

To evaluate NBAME model by maximum entropy we made prediction for NBA playoffs using selected other machine learning algorithms and compared the results with those of NBAME model. In Table 6 we present results obtained when the features in Table 1 were used together with the algorithms in Table 6 to predict outcomes of NBA playoffs between 2007 and 2015.

Table 6. Prediction accuracy (in percentages) of selected algorithms for NBA playoffs for seasons between 2007 and 2015.

Algorithm	2007-08	2008-09	2009-10	2010-11	2011-12	2012-13	2013-14	2014-15
Naive Bayes	65.0	61.9	59.1	59.3	53.6	58.8	56.5	55.0
Decision Tree	67.0	56.0	51.2	58.0	54.8	61.2	56.5	60.0
(BP)Neural Networks	69.3	63.1	52.4	67.9	56.0	63.5	50.6	57.4
Support Vector Machine	65.1	60.7	62.2	60.5	60.7	67.1	63.5	60.0
NBAME	74.4	68.2	68.3	66.7	69.0	67.1	65.2	69.1

From Table 6 we realize that our model out-performed all the other classifiers for all seasons under consideration except for the 2010-11 season where our model was out-performed by Neural Networks. The SVM model follows closely in the second position and then Neural Networks finished third. The Naïve Bayes had the lowest accuracy with an average of about 60%, this may have been caused by it's assumption that all the features were independent which was not the case. Accuracy results from the Neural Networks and Decision Tree suffer adverse variations between seasons, for example in the 2007-2008 season, the Neural Network registered impressive prediction accuracy at 69.3% but drastically reduced to 52.4% and 50.6% in 2009-2010 and 2013-2014 seasons respectively. These variations could be explained by insufficiently small size of the training dataset that may have

Peer-reviewed version available at *Entropy* **2016**, *18*, 450; doi:10.3390/e18120450

caused the models to over-fit the data. Another interesting feature of the results is that some seasons appear to be inherently more difficult to predict, such as 2013-14 season , which showed lower overall prediction accuracies for all methods. This could have been caused by some significant changes for example: Zach Randolph was suspended for punching Steven Adams while jogging back in transition in Game 6. The controversy regarding Clippers' owner Donald Sterling's racist comments arose, which led the Clippers and all NBA teams' players to protest strongly against such remark of his. The Oklahoma City Thunder failed to cover for the absence of Ibaka Serge for his injury, and so on. All of these unpredictable incidences, both on and outside the court, may have caused NBA games in this season to be highly unpredictable. Figure 4 presents a graphical representation of this data.

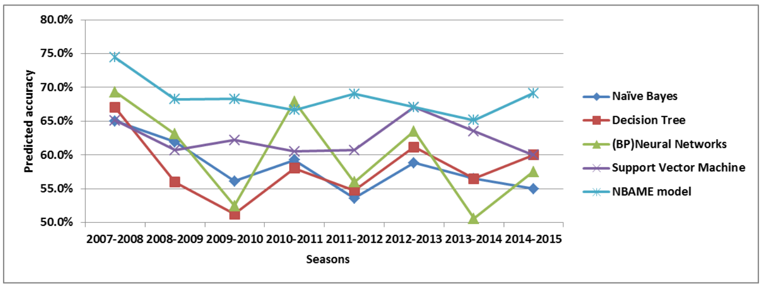

Figure 4. Comparison of performance of NBAME model against some machine learning algorithms

5. Conclusions and Future work

We applied the maximum entropy principle to construct the NBAME model and used the model to predict the outcome of the NBA playoffs from 2007-08 season to 2014-15 season, as seen in the results section, NBAME model is a good probability model for prediction of NBA games. The prediction of NBA games results is a very hard problem because there are many random factors such as the relative strengths of either teams, presence of injured players, players' attitude, and team manager's operations, that determine the winner or loser. Overall the NBAME model is able to match or perform better than other machine learning algorithms.

The predictive model in this research was able to use the mean of each basic technical features respectively from recent six games for both sides before the game started to accurately predict the outcome of an "un-played" game. Possible extensions to this research would include exploring better methods to calculate the value of the features for the coming game, such as using more effective algorithms to preprocess the features of NBA dataset, or looking for some comprehensive strengths as features.

Acknowledgments: This work was a result of inputs from various individuals and groups whose invaluable contribution we wish to acknowledge. We are deeply grateful to Mr. Seth Trachtman for interpreting some enquiries about predicting results posted on the TeamRankings.com website, and this work wouldn't have been complete without support from the research Fund for the Doctoral Program of Higher Education of China (research grant No. 20114301120001).

Author Contributions: Ge Cheng proposed an approach to solve the problem. Ge Cheng, Zhenyu Zhang and Kyebambe Moses Ntanda provided the theoretical analysis and and feasibility explanations. Zhenyu Zhang established the NBAME model, and did numerical experiments and wrote the manuscript. Ge Cheng, Kyebambe Moses Ntanda and Kimbugwe Nasser reviewed the manuscript and contributed to the final version. All authors have read and approved the final published manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Andrews, David. The (Trans) National Basketball Association: American commodity-sign culture and global-local conjuncturalism. *Articulating the Global and the Local. Boulder, CO: Westview.* **1997**, 72-101.
2. Berri D J. National Basketball Association. In *Handbook of Sports Economics Research*; 2006; pp. 21-48.
3. Zak, T. A.; Huang, C. J.; Siegfried, J. J. Production Efficiency: The Case of Professional Basketball. *J. Bus.* **1979**, *52*, 379.
4. Harville, D. a The Selection or Seeding of College Basketball or Football Teams for Postseason Competition. *J. Am. Stat. Assoc.* **2003**, *98*, 17-27.
5. Bhandari, I.; Colet, E.; Parker, J.; Pines, Z.; Pratap, R.; Ramanujam, K. Advanced Scout: Data Mining and Knowledge Discovery in NBA Data. *Data Min. Knowl. Discov.* **1997**, *1*, 121-125.
6. Beckler M, Wang H, Papamichael M. Nba oracle[J]. *Zuletzt besucht am*, **2013**, *17*.
7. Ivankovi, Z.; Rackovi, M.; Markoski, B.; Radosav, D.; Ivkovi, M. Analysis of basketball games using neural networks.*Comput. Intell. Informatics (CINTI), 2010 11th Int. Symp. 2010 11th Int. Symp.* **2010**, 251-256.
8. Loeffelholz B, Bednar E, Bauer K W. Predicting NBA games using neural networks[J]. *Journal of Quantitative Analysis in Sports*, **2009**, *5(1)* : 1-15.
9. Delen D, Cogdell D, Kasap N. A comparative analysis of data mining methods in predicting NCAA bowl outcomes[J]. *International Journal of Forecasting*, **2012**, *28(2)*: 543-552.
10. Miljković, D.; Gajić, L.; Kovačević, A.; Konjović, Z. The use of data mining for basketball matches outcomes prediction. *In SIISY 2010 - 8th IEEE International Symposium on Intelligent Systems and Informatics;* **2010**; pp. 309-312.
11. Shirley, K. A Markov model for basketball.*New Engl. Symp. Stat. Sport.* **2007**, 82-82.
12. Strumbelj, E.; Vracar, P. Simulating a basketball match with a homogeneous Markov model and forecasting the outcome. *Int. J. Forecast.* **2012**, *28*, 532-542.
13. Vracar, P.; Strumbelj, E.; Kononenko, I. Modeling basketball play-by-play data.*Expert Syst. Appl.* **2016**, *44*, 58-66.
14. Oh, M.; Keshri, S.; Iyengar, G. Graphical Model for Baskeball Match Simulation. *In MIT Sloan Sports Analytics Conference*; **2015**; pp. 1-8.
15. Stekler, H. O.; Sendor, D.; Verlander, R. Issues in sports forecasting.*Int. J. Forecast.* **2010**, *26*, 606-621.
16. Haghighat, M.; Rastegari, H.; Nourafza, N. A Review of Data Mining Techniques for Result Prediction in Sports.*Adv. Comput. Sci.* **2013**, *2*, 7-12.
17. Shannon, C. E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379-423.
18. Jaynes, E. T. Information theory and statistical mechanics. *Phys. Rev.* **1957**, *106*, 620-630.
19. Tseng, C.-Y.; Tuszynski, J. Maximum Entropy in Drug Discovery.*Entropy* **2014**, *16*, 3754-3768.
20. Xu, Y.; Wu, Z.; Jiang, L.; Song, X. A Maximum Entropy Method for a Robust Portfolio Problem.*Entropy* **2014**, *16*, 3401-3415.
21. Berger, a. L.; Pietra, V. J. D.; Pietra, S. a. D. A maximum entropy approach to natural language processing. *Comput. Linguist.* **1996**, *22*, 39-71.
22. Nigam, K.; Lafferty, J.; Mccallum, A. Using Maximum Entropy for Text Classification.*IJCAI-99 Work. Mach. Learn. Inf. Filter.* **1999**, 61-67.
23. Phillips S J; Dudík M; Schapire R E. A maximum entropy approach to species distribution modeling.*Proceedings of the twenty-first international conference on Machine learning. ACM.* **2004**, *83*.
24. Jain, A. K. Data clustering: 50 years beyond K-means.*Pattern Recognit. Lett.* **2010**, *31*, 651-666.
25. Lloyd, S. P. Least Squares Quantization in PCM. IEEE Trans. *Inf. Theory* **1982**, *28*, 129-137.
26. Münz, G.; Li, S.; Carle, G. Traffic Anomaly Detection Using K-Means Clustering.*GI/ITG Work. MMBnet* **2007**.
27. Kanungo, T.; Mount, D. M.; Netanyahu, N. S.; Piatko, C. D.; Silverman, R.; Wu, A. Y. An efficient k-means clustering algorithm: analysis and implementation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 881-892.
28. T. Pang-Ning, M. Steinbach, and V. Kumar, *Introduction to data mining*; Libr. Congr.; 2006; pp.796.

Peer-reviewed version available at *Entropy* **2016**, *18*, 450; doi:10.3390/e18120450

29. Darroch, J. N.; Ratcliff, D. Generalized Iterative Scaling for Log-Linear Models.*Ann. Math. Stat.* **1972**, *43*, 1470-1480.
30. Maechler, M.; Struyf, A.; Hubert, M.; Hornick, K. cluster: cluster analysis extended Rousseeuw et al.*R Libr.* **2009**.
31. Kaufman L, Rousseeuw P J. Finding groups in data. an introduction to cluster analysis[J]. *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics* New York: Wiley, 1990, 1990, 1.