Outlier Detection and Game Outcome Prediction of NBA Game

Aravind Anantha
aananth3@ncsu.edu

Abinav Pothuganti
apothug@ncsu.edu

Chethan Thipperudrappa
cnanniv@ncsu.edu

Huy Tu
hqtu@ncsu.edu

Abstract

The plausibility of predicting sporting events, simulating games, and estimating the performances of athletes is naturally attractive to businesses, coaches, and researchers in this data driven era. Especially, that plausibility for such major and professional sport league as National Basketball Association (NBA) has influences and fame across countries all around the world. For this milestone of the project, the main objective is to report our current progress and experiment results from using various Machine Learning methods.

1 Introduction

Thirty teams comprise two conferences in the NBA. Throughout the regular season, there are 1230 NBA games in total. It brings in significantly monetary opportunities for businesses, enterprising sport gamblers. The huge fan following and increasing interest motivates us to predict the MVP (Most Valuable Players) and game outcome. This milestone project report would explore prior researches that have been done relatively similar to NBA game outcome and outstanding players prediction.

There are 34 features for our NBA dataset. We use data beginning from the 1980 season since that is when 3 pointers were first introduced. The input is a team feature vector containing the 31 features for all NBA teams for several seasons. Each data point is the performance of each team in a season [1]. Various approaches for feature selections (BIC, PCA, and stepAIC) would be applied along with linear regression model and SVM Regression for team win ratio of per year prediction. Outstanding players are predicted based on the various attributes such as rebounds, steals, three points etc. We are making use of multivariate outliers library in r to perform outlier detection.

Our report is constructed as follows: (2) report on related work that has been done in the past, (3) proposed methodologies for feature selection, model selection and results from our experiments with those methodologies, (4) takeaways and conclusion.

2 Related Work

There were many works on predicting NBA team wins on per-game level and predicting the win percentage of that team in that NBA season with or without basing on the per-game level prediction. One paper by Evan Giarta and Nattapoom Asavareongchai predicts the win percentage of a team in a season using the per season cumulative stats of all players with PCA for feature selection [2]. The learning model used was a second order polynomial regression model [2]. Moreover, Bernard, L et al.'s paper in 2009 and Pedro et al.'s paper in 2012 used network based algorithm to predict NBA wins with very high success rate [3] [4]. Kevin Wheeler in 2012 used Linear Regression for player performance prediction along with SVM and Naive Bayes for outcome per-game prediction with PCA for feature reduction [6].

Mahalanobis distance is a multi-dimensional generalization of the idea of measuring how many standard deviations away P is from the mean of D. It measures the number of standard deviations

from P to the mean of D thus it is unitless and scale invariant.As per the paper "An Overview of Multiple Outliers in Multidimensional Data"[5], Mahalanobis distance of the points always falls in Chi Squared distribution and if a data point is not in Chi Squared distribution then it can be possible outlier. Calculation of Mahalanobis distance of all the observations can be done using the MCD(Minimum Covariance Determinant) estimates. Calculation of MCD(Minimum Covariance Determinant) [3] is a very computationally expensive operation. MCD estimates are then used to compute the Mahalanobis distance of all the observations to detect outliers.

3 Proposed Methods and Experimental Results

3.1 NBA Dataset

A conventional framework of data mining is applied for this project. We begin by analyzing the data available for our experiment. The given NBA dataset was collected from 1946. However, with a lot of missing data and rules changes between 1946 to 1980, especially the introduction of the three-points rule that lead to significant differences in how the team performing and strategies for playing the game, we have determined to only consider data from 1980.

3.2 Game Outcome Prediction

3.2.1 Data Pre-Processing

With the abundant availability of 31 features, it is important to reduce those to only features that have the most impact or are most relevant to the problem. Consequently, it would help to lighten the cost in term of time and space for algorithms to analyze a big scale of data set, find fitted model, and evaluate the model. Therefore, some best feature selection models were applied including:

- BIC is an estimate of a function of the probability of a model being true, under a certain Bayesian set among to the sets of candidates in which indicates a lower BIC means that a model is considered to be more likely to be the true model. There are forward and backward selection, we applied backward selection. In our experiment, we got 7 features being selected in the end for the lowest BIC(o_fgm, o_ftm, o_3pm, d_fga, d_ftm, d_fta, and d_3pa).
- StepAIC is an estimate of a constant plus the relative distance selecting a model that describe the unknown true likelihood function of the data and the fitted likelihood function of the model which indicates that a lower AIC means a model is considered to be closer to the truth. In our experiment, we got 13 features being selected in the end for the lowest stepAIC (o_fgm, o_fga, o_ftm, o_oreb, o_asts, o_to, o_3pm, d_fgm, d_fga, d_ftm, d_to, d_blk, and d_3pm)
- PCA uses orthogonal transformation to transform a set of possibly correlated variables into a set of linearly uncorrelated variables called principal components. We used the preProcess function from the caret library for the task of PCA. In our experiment we choose principal components that explain 95% of the variance in the data. This came out to choosing the first ten principal components. The Table 1 below shows the results of our PCA.

Table 1: PCA for Team Performance Prediction

PC's	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
Variance	0.69	0.14	0.05	0.03	0.02	0.016	0.01	0.006	0.005	0.004
Cumulative Variance	0.69	0.84	0.89	0.92	0.94	0.956	0.966	0.972	0.977	0.981

After reducing the number of dimensions and unnecessary features through feature selection, for predicting game outcome through predicting team performance part of the project was taken in two different approaches, regression and classification approaches.

3.2.2 Team Performance - Regression

Different machine learning regression algorithms were applied to approach this problem including linear regression (LR), Ridge Regression, Least Absolute Shrinkage and Selection Operator (LASSO) Regression, and Support Vector Regression(SVR) for the task of predicting the wins per season. Our baseline method is the linear regression model that runs on all of the features.

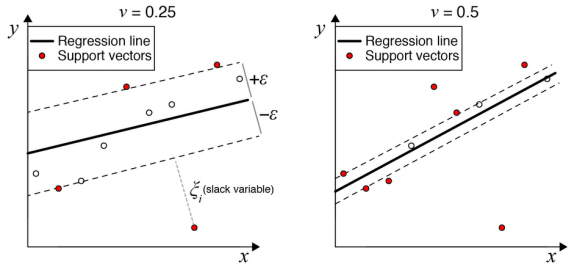
- Linear regression is used to model the relationship between a dependent variable and one or independent variables. For our analysis we use the features that were selected from the dimensionality methods described above. We then try to improve our model by using more advanced regression models such as Ridge Regression and Lasso regression[10].
- Ridge regression imposes constraints on the coefficients which shrink them towards zero. The coefficients are calculated by minimizing the function given below[10].

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}\beta_j^2=\mathrm{RSS}+\lambda\sum_{j=1}^{p}\beta_j^2$$

- Lasso regression also shrinks the coefficients towards zero. The function that lasso regression minimizes to obtain coefficients is given below[10].

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|\ =\mathrm{RSS}+\lambda\sum_{j=1}^{p}|\beta_j|$$

- Support Vector Regression gives coefficients that minimize a function where only residuals larger than some positive constant (epsilon) contribute to the loss function. We use the default value of 0.1 for epsilon from the svm function in R.



Using 75% of the dataset as the training data and the remaining 25% as test data. The data was split into training and test randomly using the sampling method without replacement. The mean of the original data is 0.50 and standard deviation is 0.156. The results of all the regression functions are given below.

Table 2: Team Performance Evaluation

			Root Mean Square Error in %	
Regression Methods	Mean	SD	Train error	Test error
LR with PCA	0.504	0.134	6.785	6.85
LR with stepAIC	0.503	0.156	3.828	4.056
SVR with stepAIC	0.502	0.133	4.539	5.445
SVR with PCA	0.503	0.126	5.03	7.24
SVR with BIC	0.509	0.100	9.592	10.631
Lasso	0.501	0.121	7.605	14.754
Ridge	0.500	0.052	13.365	14.754
Baseline Method	0.004	0.684	77.566	67.316

Multivariate linear regression with stepAIC for feature selection gave us the best results with a root mean square error on the testing data being 4.05.

From previous season performance of that team, it is plausible to predict the outcome of individual games in the future. Out of the two competing teams, predicting the team that would win by choosing the team which has a better performance in previous season is a good bet. There have been previous similar researches on this method such as Yuanhao Yang[7]. They were able to predict the outcome of games with a 63% accuracy.

3.2.3 Team Performance - Classification

For the second approach, work has been done to predict team performance by classification methods. From win ratio, to give audience a holistic idea of how well the team perform per season, each team was labelled according to their performance as mentioned in the table below.

Table 3: Classification of teams

Class	Win percentage	Number of instances
Class 1 (poor)	0%-35%	135
Class 2 (average)	35%-50%	190
Class 3 (good)	50%-65%	227
Class 4 (outstanding)	65%-100%	132

We have attempted different classifier models to approach this team performance classification problem. They are:

- Support Vector Machine (SVM) is primarily a classier method that performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels [10].
- Random Forest is a tree-based methods for regression and classification. It is meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting [10].
- Naive Bay Classifier is one of the most used algorithm. It is a design learning algorithms based on the rule of probabilistic model. Learning $f : X -> Y$ and $f : X_1, X_2, ..., X_n -> Y$ or specifically $P(X|Y)$ and $P(X_1...X_n|Y)$ [10].

Table 4: Overall Accuracy and Majority Class for different classifiers models

Classifying Methods	Overall Accuracy
SVM	60.8%
Random Forest	61.4%
Naive Bayes	57.3%

From the table above, we can see the performance of our classification methods in term of overall accuracy. However, classification accuracy alone is typically not enough information to make the decision of how strong and good our models are. Therefore, performance of statistical measurements of f-measure, precision, and recall were analyzed for each individual classes as our graphs below shown from figure 1-4.

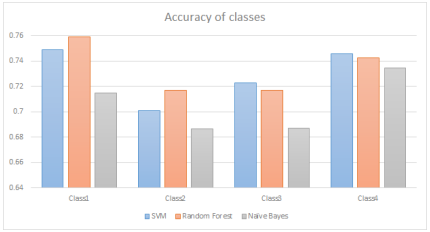


Figure 1: Accuracy in Different Models

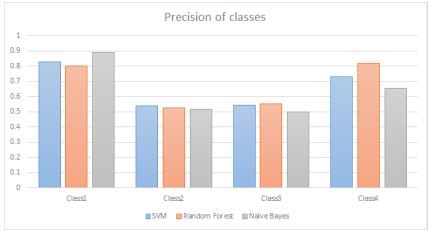


Figure 2: Precision in Different Models

Figure 3: Recall in Different Models

Figure 4: F-measure in Different Models

There is no explicit majority class in our predicition, classes 2 and 3 have higher instances than classes 1 and 4. From analyzing both the overall accuracy performance and individual statistical measurements per classes of each classification methods, random forest consistently perform better than Naive Bayes and SVM methods.

3.3 Outstanding Player Detection

3.3.1 Data Pre-Processing

For our outstanding player detection, we considered 5 parts for our dataset i,e (i) Statistics of players in regular season (19113 records) (ii) Statistics in playoff season (7544) (iii) Statistics in all star season (1463 records) (iv) Statistics in playoff career (2056 records) (v) Statistic in regular season career (3760 records).

Though the data corresponds to different seasons the attributes are common among all of them. There are 23 attributes in each dataset and 6 of the attributes are nominal and they are not used in the analysis. Remaining 17 attributes are numerical attributes and they are scaled to lie between 0 and 1 before performing the analysis. We have considered the dataset from the year 1980 as some of the attributes have missing values prior to 1980.

In case of player_regular_season data, we are aggregating the player statistics based on the player name and the year and then applying the outlier detection algorithm to find the outstanding players in a season. For dimensionality reduction we used PCA, in particular we used prcomp method in r to find the PCs. The PCA results of the dataset for the season 2004-05 are as below:

Table 5: PCA for Outlier Detection

PC's	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
Variance	0.59	0.12	0.04	0.03	0.03	0.02	0.02	0.02	0.01	0.01
Cumulative Variance	0.59	0.72	0.77	0.8	0.83	0.86	0.89	0.91	0.93	0.95

From the above list of PCs, 5 Principle Components were considered for the analysis by keeping 95% of the variance in the data.

Outliers are the ones which deviate from the other data points the most. Chi-squared plot function is used to detect the outliers in the data. The function chisq.plot plots the ordered robust mahalanobis distances of the data against the quantiles of the Chi-squared distribution.

3.3.2 Pseudo Ground Truth

We considered the MVP shortlisted players from the NBA official website for the year 2004 and compare against the results of our methods.

By using unsupervised learning to find the results, it is notable that ground truth is not available to compare with. The official shortlisted MVP candidates are formed by considering public voting, performance in critical matches and other factors. As we don't have such information, we cannot consider this list as ground truth.

3.3.3 MV outlier detecting based on chi-square distrubution

The Mahalanobis distance of the normally distributed data follows a Chi Square distribution. Chisq.plot uses this information to identify outliers present in the data.

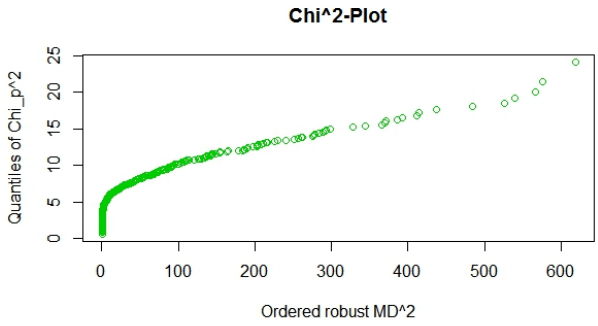


Figure 5: Plot of ordered squared Mahalanobis distance of the data against quantiles of chisquare distribution

Chisq.plot in the figure 5 plots the ordered squared robust mahalanobis distances of the data against the quantiles of the Chi-squared distribution. If the data is normally distributed, the two values should approximately correspond to each other, if a data point is deviating from this behaviour it can be identified as an outlier.

```
Remove outliers with left-click, stop with right-click on plotting device
 [1] "Vince Carter"    "Antoine Walker"    "Allen Iverson"    "Dwyane Wade"
 [5] "Chris Webber"    "Amare Stoudemire"  "LeBron James"     "Kevin Garnett"
 [9] "Dirk Nowitzki"   "Craig Claxton"     "Ben Wallace"      "Marcus Camby"
[13] "Cuttino Mobley"  "Shawn Marion"      "Kobe Bryant"      "Gilbert Arenas"
[17] "Steve Francis"   "Shaquille O'Neal"  "Baron Davis"      "Corey Maggette"
[21] "Tracy McGrady"   "Stephon Marbury"   "Tim Duncan"       "Ray Allen"
[25] "Steve Nash"      "Paul Pierce"       "Quentin Richardson" "Elton Brand"
[29] "Brevin Knight"   "Carmelo Anthony"
```

Figure 6: Outliers detected using chisq plot method

We considered 30 outliers detected using this method as shown in figure 6 and compared it against the MVP shortlist from NBA official website and found that 14 of the 16 players were detected successfully. Highlighted players were found in the MVP shortlist of NBA website

3.3.4 Outlier detection based on k-means clustering

K-means clustering aims to partition the given data points into k clusters in which each data point belongs to the cluster with the nearest mean. The effectiveness of the k-means algorithm lies in identifying the number of clusters required to partition the given dataset.

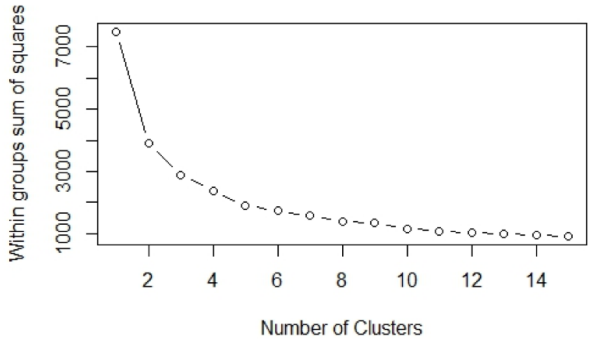


Figure 7: Plot of within group sum of squared distances vs number of clusters

In order to find the optimum number of clusters, the sum of the squared distances of data points form the centroids of their corresponding clusters were plotted. The elbow at number of clusters was found as 3 as observed from the figure 7.

Once we identify the number of clusters, we used k-means to cluster the given dataset and then calculate the distance of each data point from its cluster centroid. We have collected 30 data points which are away from their respective centroids.

Figure 8: Output of outlier detection using k-means clustering

We considered 30 outliers detected using K - Means as shown in figure 8 and compared it against the MVP shortlist from NBA official website and found that 13 of the players were detected successfully.

3.3.5 Outlier detection based on dbscan clustering

DBSCAN stands for Density-based spatial clustering of applications with noise, It is a density-based clustering algorithm: given a set of points in some space, it groups together points that are closely packed together (points with many nearby neighbors), marking as outliers points that lie alone in low-density regions (whose nearest neighbors are too far away). It categorizes each point as a core, border or noise point.

To detect outliers, we categorize the points within the distance of epsilon of 0.22 and minimum points of 3 as core points and the points which have fewer than the minPoints in the radius epsilon are considered as border points and the rest of the points are categorized as outliers.

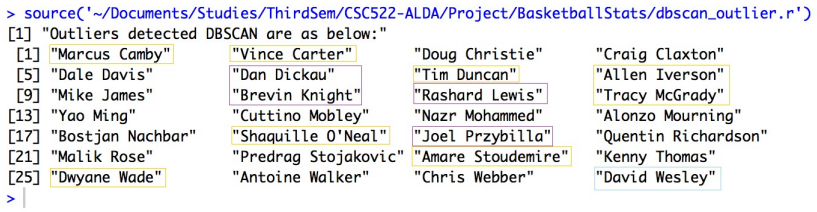


Figure 9: Output of outliers detected using DBSCAN

We considered 30 outliers detected using DBSCAN and compared it against the MVP shortlist from NBA official website and found that 8 of the players were detected successfully.

4 Conclusion

As a research project, we have worked on detecting outstanding players per season in NBA and predicting the outcome of NBA games by predicting team performance per season. Our project follows the paradigm of data mining with framing executable and valuable research problems, pre-processing data from the bottom ground, developing appropriate models, and training with evaluating models. For predicting game outcome part, we have approached it with regression and classification supervised techniques as below:

- From our regression analysis with the NBA dataset, multi-variate linear regression with step AIC used for feature selection provide a model with only 4% of RMSE to predict the win ratio of a team the season.
- Random forest gave us the best model when compared to other models with an accuracy of 61% and other statistical measurements per classes. It is notable that because of categorizing

the win ration of teams we are losing the accuracy of the dataset and hence the accuracy is low. Moreover, by looking only at overall accuracy is not enough,

For detecting NBA outstanding player per season, we have approached it with unsupervised outlier detection techniques as below:

- The designed models were able to detect the outstanding players to an extent of finding 14 of the 16 MVP shortlisted players in a pool of only 30 outliers. Multi-variate outlier detection based on chi-square distribution and k-means clustering gave superior results compared to DBSCAN for this dataset.
- It is important to keep in mind that in case of unsupervised learning as our outstanding player detection, ground truth is not available. Hence, there is no way to be empirically certain that the designed algorithm produce the intended results. We can see that even though DBSCAN and k-means clustering are effective methods to determine outliers, k-means clustering performed better than DBSCAN method for this dataset.

5 Acknowledgement

We want to thank Dr. Min Chi, Associate Professor at North Carolina State University, and the TAs including Yuan Zhang, Linting Xue, and Yihuan Dong for their support and guidance throughout this specific project.

6 Project Record

Our project is recorded at this Github repository link: `https://github.ncsu.edu/hqtu/BasketballStats`

References

[1] Basketball Reference. http://www.basketball-reference.com/

[2] Giarta, E. & Asavareongchai, N. (2015) *Predicting Win Percentage and Winning Features of NBA Teams*.

[3] Loeffelholz, B. & Bednar, E. & Bauer, K. W. (2009) *Journal of Quantitative Analysis in Sports*.

[4] Melo, P. O. S. V. D. & Almeida, V. A. F. & Loureiro, A. A. F. & Faloutsos, C. (2012) *Forecasting in the NBA and Other Team Sports: Network Effects in Action*.

[5] Sajesh, T. A. & Srinivasan, M. R. (2013) *An Overview of Multiple Outliers in Multidimensional Data*.

[6] Wheeler, K. (2012) *Predicting NBA Player Performance*.

[7] Yang, S. (2015) *Predicting Regular Season Results of NBA Teams Based on Regression Analysis of Common Basketball Statistics*.

[8] Newman, A. & Liu, C. L. & Green, M. & Gentles, A. & Xu, Y. & Hoang, C. & Diehn,M. & Ash A Alizadeh (2014) *Robust enumeration of cell subsets from tissue expression profiles*.

[9] Lin, J. & Short, L. &Sundaresan, V. (2014) *Predicting National Basketball Association Winners*.

[10] James, G. & Witten, D. & Hastie, T. & Tibshirani, R. (2014) *An Introduction to Statistical Learning with Applications in R*