Project Final Report

EECS 349 Machine Learning — 2016 Spring

Xinyu Zhang, Chenhui Zhou - June 7, 2016

INTRODUCTION

NBA is full of talented athletes, they are attempt to help their team to win the NBA final Championship also they are compensated with great number of money as well. NBA players' salaries is always an interesting topic. Our task is to predict the salary of a certain NBA player in the coming season based on his past performance in NBA. We will also be able to validate whether a player worth for his salary, or whether a player should get more.

DATA

The data we used were the individual statistics records by each player in NBA from 2011-12 season to 2015-16 season from http://www.basketball-reference.com/?lid=homepage_logo. What we included in the final table we used are shown as following image:

Salary	Age	GP	GS	MP	FG%	3P%	FT%	OR	DR	AST	STL	BLK	TO	PF	PTS	Position	Salary Cap

Age is the player's age at that season and G, GS, MP represent the number of games this player took and how many minutes he played at that season. Also the table includes the FG%: Field Goals Percentage, 3P%: Three Point Field Goal Percentage, FT%: Free Throw Percentage, ORB: Number of Offensive Rebound, DRB: Number of Defensive Rebound, AST: Number of Assists, STL: Number of Steals, BLK: Number of Blocks, TOV: Number of Turnovers, PF: Number of Personal Fouls and PTS: Points at that season.

There are 16 attributes. Attribute 'Position' is nominal, other attributes are all numerical. We use attribute 'Salary'(player's salary for next season) and attribute 'SalaryCap'(The salary cap for next season) for labeling, we divide 'Salary' by 'SalaryCap' and categorize the results ranging from 5% to 25%(due to NBA salary cap) into 5 classes. Every 5% count as one class. Right now there are 1200 examples used for training and 300 examples used for testing.

RESULTS & ANALYSIS

The algorithms we tried are: SVM, k Nearest Neighbors, AdaBoost kNN, Random Forest, Naive Bayes. We evaluated these algorithms by testing their 10 fold cross-validation training accuracy. The result is shown in Figure 1. And we also calculated their confusion matrixes, as shown in Figure 2.

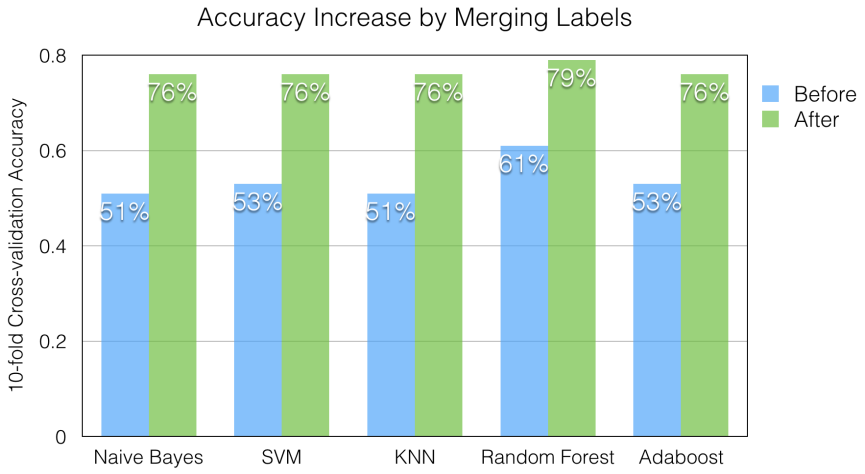

Figure 1

As the histogram shown above, the accuracy for each algorithm increases after merging label. Before merging, there are 8 classes for our data, 0%, 5%, 10%, 15%, 20%, 25%, 30%, 35%, and the accuracy for each algorithm is not so good. After analyzing, we found this is due to that players in 0% and 5% have very similar performance, and players above 20% also have very similar performance. So it's very hard to classify such classes. To address this problem, we merged all players under 10% to 5% and all players above 20% to 25%. According to the result shown in the histogram, this solution gives a much better result. And among all the algorithms, Random Forest Algorithm performs the best. So we finally use Random Forest model to make predictions.

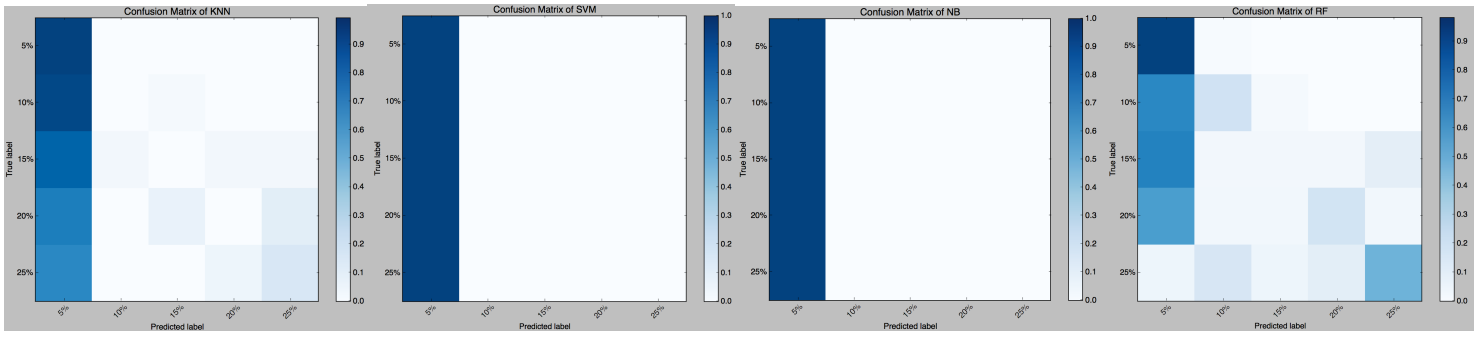

Figure 2

Figure 2 shows the confusion matrix for each algorithm, we can see that SVM and Naive Bayes classify every sample to 5%, while kNN and Random Forest do a pretty good job. According to both confusion matrix and cross-validation accuracy, Random Forest algorithm gives the best performance. So we choose to implement Random Forest model for our task, and we also further evaluated our Random Forest model. First, we tried to find out the most suitable value for the number of estimators in our Random Forest model. The result is shown in Figure 3.

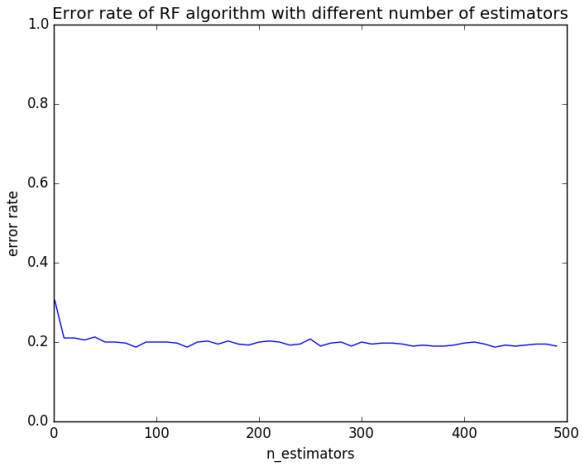

Figure 3

As Figure 3 shows, the error rate almost keeps stable after 100 estimators, so for our Random Forest model, we set the n_estimators parameter as 100. And we also evaluated the importance of features on our classification task. The result is shown in Figure 4. The red bars are the feature importances of the forest, along with their inter-trees variability. And the result fits our understanding towards our task, since Age, PTS and GS are the 3 most important features, and position wouldn't impact the salary when considered individually.

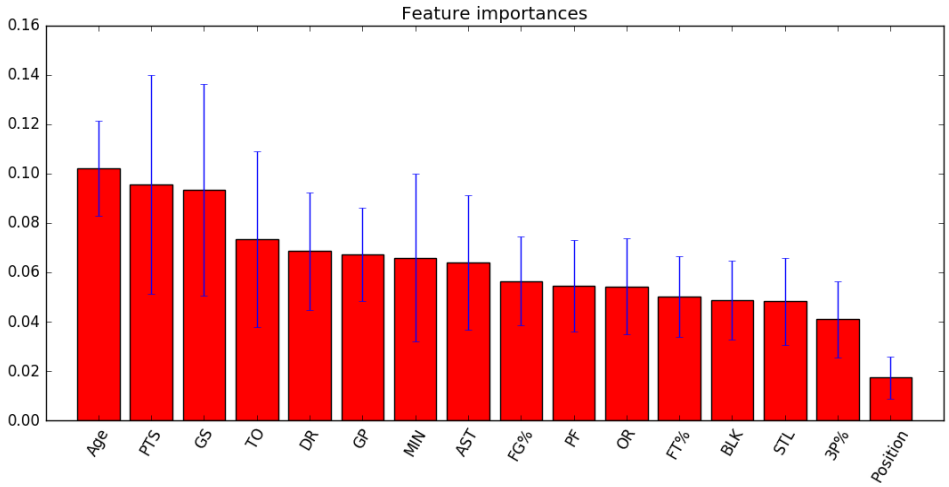

Figure 4

FUTURE WORK

In the future there are some work need us to do. Right now our data set for train or test is not sufficient enough, so the result of our model is not perfect. We could include more data from other seasons. Also, right now we categorize the results into 5 classes, the number of classes we select is determine by our model performance. If we could include more data to increase our accuracy, we may could categorize ore results. Moreover, there is a lot of noise in our data set. Specifically, some players' salary couldn't represent their performance and sometimes higher salary doesn't mean good state data. For examples Kobe Bryant has the highest salary in NBA but his state data is not as impressive as other famous players. In order to solve this problem, we may include more feature into our data like reputation or do more complex data preprocessing to eliminate this situation.

TEAM

Xinyu Zhang: Wrote the Python program.
Chenhui Zhou: Collected and processed the dataset.