MLP Coursework 3: Kobe Bryant Shot Selection

G106
s1879797, s1873947, s1830374

Abstract

Within the problem constraint of this paper, different aspects of a classification problem are considered, and attempts have been made to optimize them individually by considering the overall improvement in accuracy. These different aspects cover the domain of data-prepossessing, classification and accuracy evaluation methods. For each of these domains, different well-known algorithms are employed on a data set comprising of 30967 feature spaces, with 25 distinct features in each row. The data set comprises all the information recorded after observing the performance of all the shots made by the well-known basketball player Kobe Bryant throughout his career. The end objective of the project is to make accurate predictions of a shot being scored given some feature information, by improving upon the performance of the baseline, where a simple Random Forest classifier is used for classification with k-fold cross-validation.

1. Introduction

The idea of this project is based on a Kaggle competition (Kaggle, 2016a) from 3 years ago. We found this classification problem quite interesting because the data provided is not standardized and some of the data attributes have a specific form which could narrow or expand the range of the classifiers that we can use, from a simple Logistic Regression (Tolles & Meurer, 2016) or Naive Bayes Classifier to Nearest Neighbour or Neural Network problem.

The data used in this project, as we said before, is not standardized and it will be interesting to see how we can bring the data to the same format, what features can we remove (some of features are redundant and others simply do not make any contribution to the classification process) and if we can reduce the dimensionality of the data (e.g. by using PCA (Jolliffe & Cadima, 2016) with SVM, or other procedures) in order to attain the highest accuracy. We use machine learning to uncover hidden data trends. Our goal is to build multiple models which involve different types of classifiers and the use of various procedures to manipulate the data, so that we can obtain the optimal accuracy in predicting which of Kobe Bryant's shots will go and which will not. It would also be interesting to see how results differ when we use the entire data for the classification process from when we use just the data prior to a shot, for the same respective shot prediction.

2. Research questions and objectives

In the recent years Neural Networks for prediction of sports decisions have gained higher accuracy over the human predictions. (Loeffelholz, 2009). Hence in terms of that, the main objective of the problem statement at hand is that given relevant information, how accurately can we predict if a ball fired by a basketball player from a certain location and with certain angle within a courtyard is going to secure points or not. In terms of this, many different classification algorithms have been employed to determine upon accurate probabilities of classifications. In this paper, however, we would be trying to improve upon the accuracy of the probability by segregating the problem into three sections, e.g. Data pre-processing, classification and validation. We would then try to improve the overall general performance by targeting the performance accuracy of each of these segments individually by applying different techniques. In terms of data-prepossessing, it is an important question to ask if all the provided features are essential or not. In similar manner, if the dimensions of the features are reduced using algorithms such as PCA, then would it bring a significant amount of change in terms of accuracy or maybe in terms of improving its computational costs.

Furthermore, it is also worth exploring different algorithms for classification and trying to determine if employing more complex algorithms apart from Bayes Classifier and Random Forest Classifier, would result in a performance increase.

Lastly, the methods employed to separate the data set for training, validation and testing are also given importance for this problem. For the sake of which it is considered worth exploring to find a sorting pattern within the data base depending on the provided date feature. The question that we would then like to answer would be that if we divide the data base into chunks for training, where each chunk is then used for predicting a new data value, which is then updated into the next chunk of data, then would such a method of training yield better results as compared to a more well known and commonly used method, such as k-fold cross validation.

3. Data set and task

For this project we are going to use a data set provided by Kaggle (Kaggle, 2016b) which contains the circumstances

of every field goal attempted by Kobe Bryant during his 20-years career.

The data contains 30697 shots attempted by Kobe Bryant from which 5000 shots were removed (as for these 5000 records of the shots the *shot_made_flag* field was left empty). The idea behind this Kaggle competition was to write a classifier which could predict the outcomes for these 5000 shots that were removed. Since we do not have the actual outcomes for these, we cannot provide any measure of accuracy hence dropping the rows with the missing features was deemed unavoidable. We are then left with 25697 shots that we will split into training, validation and testing sets (70%, 15%, 15%), which means that we have around 3855 shots that we will use as test set (shots that we can use because we have the actual outcomes and we can compute accuracy of each model tested). In order to make possible to replicate this project, we chose these examples by selecting the next index of each one of the 5000 shots without an outcome (also checking that these selected shots have an outcome) and from all of these shots we have selected the last 3855 to be used for our testing examples. We will not touch this testing data set until the last step of our project when we will have 2-3 models which provide very good results. After these procedures we obtained a test file of 3855 examples (approximately 15% of the data, after we removed the 5000 shots without an outcome) and 21842 examples which we will use for training and validation.

Each example (every Kobe's shot) is represented using 25 attributes as we can see in table 1. As we received the data it was impossible to use it without pre-processing because some attributes were redundant and the data was too raw (some attributes were represented as numbers while other as strings), hence we needed some standardization. We dropped some attributes to facilitate the baseline system such as: team_id, team_name (Kobe Bryant played his entire career for only one team - Los Angeles Lakers), shot_id, game_event_id, game_id, game_id(these attributes are also unnecessary for the classification process), shot_zone_area, shot_zone_range, shot_zone_basic, matchup (some of these attributes could provide useful information for the classification process, but right now in order to provide a clearer baseline system we chose to drop them). We said earlier that we need some standardization so we converted the values for the following attributes from strings to numerical values using one-hot encoding: action_type, combined_shot_type, shot_type, opponent, period, season.

We provided a graphical display of Kobe's shots (the data) that we have to process and analyze. In figure 1 (up) are displayed all of Kobe's shots using X and Y locations (we can see how the shots are disposed in a basketball court), while in figure 1 (down) are displayed all of Kobe's shots based on Latitude and Longitude coordinates (we can observe a high similarity between the two figures since the Latitude and Longitude coordinates are linear transformation of Y and X locations).

Attribute	**Description**
action_type	How the player shot the ball: Jump Shot, Dunk Shot, Layup Shot... (57 possibilities)
combined_shot_type	The big categories of shots: Layup, Dunk, Hook... (6 possibilities)
game_event_id	Event Id
game_id	Game Id
lat	Latitude of the shot (between 33.253 and 34.088)
loc_x	The court latitude measured in tenths of a foot (between -250 and 248)
loc_y	The court latitude measured in tenths of a foot (between -44 and 791)
lon	Longitude of the shot (between -118.519 and -118.021)
minutes_remaining	Minutes left in the quarter (between 0 and 11)
period	Quarter of the game (with also extra time) (between 1 and 7)
playoffs	Game from playoff (0 and 1)
season	The season in which the shot was taken
seconds_remaining	Seconds left in the quarter
shot_distance	The distance between the net and the shot spot (between 0 and 79 feet)
shot_made_flag	Success of the shot (0 and 1)
shot_type	2PT Shot or 3PT Shot
shot_zone_area	The zone from which the player shot the ball (Center, Back Court, Left Side, Right Side, Left Side Center, Right Side Center)
shot_zone_basic	The 7 basic zones for shoting
shot_zone_range	Less Than 8 ft, 8-16 ft, 16-24 ft, 24+ ft, Back Court Shot
team_id	Team Id
team_name	Team Name
game_date	Date of the shot
matchup	The opponent and the location of the game (Home or Away)
opponent	All 33 teams that Kobe played against
shot_id	Shot Id

Table 1. The 25 attributes which define every shot

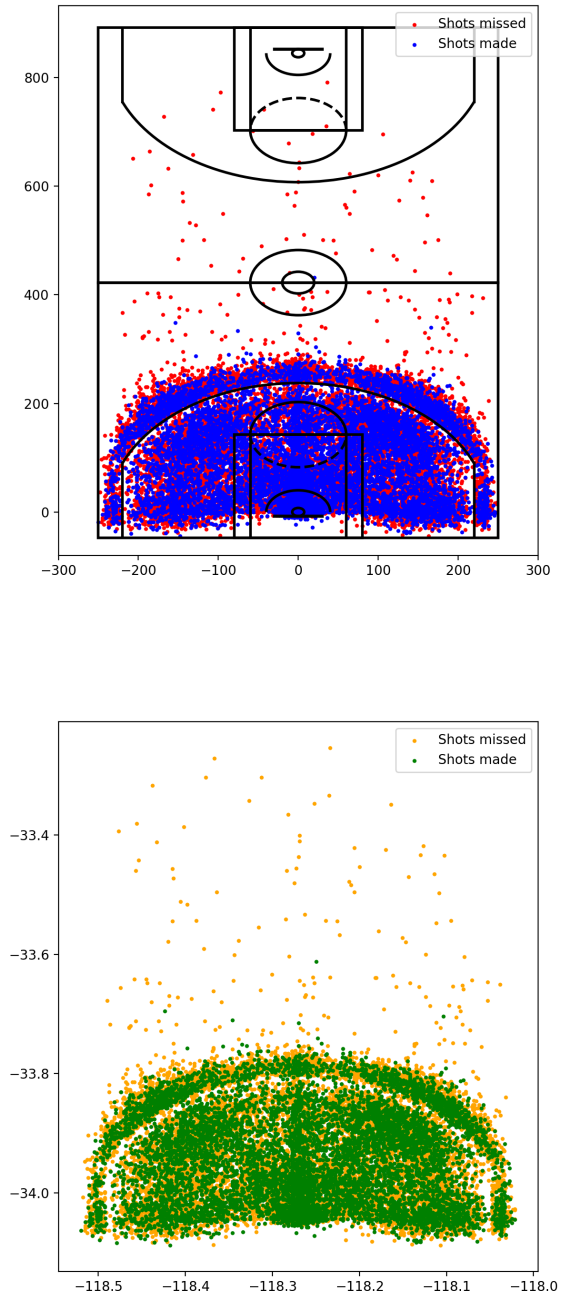

Figure 1. Kobe's shots based on X and Y locations (up) and based on Latitude and Longitude coordinates (down)

4. Methodology

Feature selection:
High prediction accuracy can be obtained by eliminating the irrelevant and redundant features, giving importance to the relevant features that actually contribute to the output. This feature selection is a process of finding a mapping/method that reduces the dimensionality of the data by selecting the relevant input features. Also, by collecting relevant features the computational cost can be reduced. We will be performing feature selection using 2 methods:

1. We will use the Neural Network Feature Selector proposed by Setiono and Liu (Setiono & Liu, 1997). This algorithm is a backward elimination method for feature selection. The method starts with the whole set of available feature variables and then for each attribute variable, the accuracy of the network is evaluated with all the weights associated with that variable set to zero. The variable that gives the lowest decrease in accuracy will be removed.

2. PCA (Jolliffe & Cadima, 2016) will also be investigated to select appropriate features as inputs to the neural network. PCA works by eliminating the smallest Eigen vectors and reducing the dimensionality of the data.

Classification:
In this paper, we will evaluate the accuracy of prediction using Recurrent Neural Networks as against the classic classification algorithms. For the stationary data set obtained after the feature selection process, we have evaluated the random forests (Liaw & Wiener, 2001) as the baseline experiment. We will be futher investigating traditional classification algorithms - Support Vector Machines (Burges, 1998) and Linear classifiers (Tolles & Meurer, 2016). The performance and accuracy of these traditional classification algorithms will be compared against the RNN (designed for the classification purpose) and the performance will be evaluated. The motivation behind using RNN for classification is from paper: "Combining recurrent neural networks with eigenvector methods for classification of ECG beats". This paper addresses the classification of data into 4 ECG classes. We will apply a similar strategy to classify the shots (hits/miss) in our data (Übeyli, 2009).

Accuracy evaluation:
K-fold cross validation (Kohavi, 1995) is used for accuracy estimation so that all the algorithms are evaluated under the same test conditions. The limited dataset is also made good use of for evaluation with the K-fold cross validation.

5. Experiments

In order to analyze the performance and accuracy of the algorithms that are to be used within this project in an attempt to answer the research questions and hypothesis, a baseline experiment was constructed. The premise of which comprised of minor data set modifications that are previously mentioned in the Data set and task section. To build a classification model for the baseline, a general 'Random Forest Classifier' algorithm was considered with k-fold cross validation. The classifier was selected due to its ability to handle missing values and predicting results with relatively high accuracy (Liaw & Wiener, 2001). Apart from this,

the algorithm is simple to implement, which secured its position as a perfect candidate for the baseline model. In order to determine the optimal number of estimators and the depth of the tree with the Random Forest Classifier, three different values (1, 10 and 100) were considered. The classifier was trained anew with varying values selected from the data space with 10-fold cross validation (Kohavi, 1995). Averaging the scores from each experiment with values from the data space, indicated the best results to be obtained from 100 number of estimators of the tree with a depth of 10. These scores for evaluating the accuracy of the results were calculated using the log-loss function (Shimodaira, 2000). The results attained from the baseline are indicated in the following table:

Data Type	Mean Absolute Error	Accuracy %
Train	0.31 degrees	69.34
Test	0.33 degrees	67.39

Table 2. Baseline Model Results

6. Interim conclusions

We standardized and split the data successfully and even if we lost 5000 examples (the shots that we had to remove because they did not have outcomes and we could not verify the accuracy of our predictions) we still had left 25697 examples which, in our opinion, are more than enough to build a fair and accurate classifier. The baseline model showed very good results (for a basic baseline system) getting an accuracy of 67.39% as stated in table 2.

As we stated in the beginning, we are going to look at the predictions from two perspectives: on one side we will predict the shots from the test set using all of the training data without any constraints (this is the perspective used in building the baseline model which brought more than decent results) and on the other side we will predict every shot from the test set using only data prior to the date of that shot tested. This second perspective could possibly show worse results because of the slimmer training data used for the first shots tested, but it is the realistic way of testing.

7. Plan

1. We will apply various algorithms for preprocessing the data - PCA (Jolliffe & Cadima, 2016) and the Neural Network Feature selector proposed by Setiono and Liu (Setiono & Liu, 1997) to remove irrelevant information and to make the computation is faster.

2. We will also explore different classifiers

   (a) Logistic Regression (Tolles & Meurer, 2016)

   (b) Support Vector Machines (Burges, 1998)

   (c) Recurrent Neural Networks (Übeyli, 2009)

3. We will use the combinations of feature selectors and classifiers proposed above in points 1 and 2. We will compare the performance and accuracy of each of these methods.

4. For the baseline experiment using Random Forests (Liaw & Wiener, 2001), we have used the entire dataset to make the predictions. Going forward, we would also experiment on the effects and consequences of the previous data in making the current shots instead of taking the entire dataset.

5. For the accuracy estimation, we will be using K-fold cross validation (Kohavi, 1995) which was also used in the baseline experiment.

References

Burges, Christopher J.C. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121âĂŞ167, Jun 1998. ISSN 1573-756X. doi: 10.1023/A:1009715923555. URL https://doi.org/10.1023/A:1009715923555.

Jolliffe, Ian T. and Cadima, Jorge. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2016. URL https://royalsocietypublishing.org/doi/full/10.1098/rsta.2015.0202.

Kaggle. Kobe Bryant Shot Selection Competition. 2016a. URL https://www.kaggle.com/c/kobe-bryant-shot-selection.

Kaggle. Kobe Bryant Shot Selection Data. 2016b. URL https://www.kaggle.com/c/kobe-bryant-shot-selection/data.

Kohavi, Ron. A study of cross-validation and bootstrap for accuracy estimation and model selection. Technical report, Computer Science Department, Stanford University, CA, 1995.

Liaw, Andy and Wiener, Matthew. Classification and regression by randomforest. *Forest*, 23, 11 2001.

Loeffelholz, B., Bednar E. Bauer K. Predicting NBA Games Using Neural Networks. Journal of Quantitative Analysis in Sport. 2009. URL doi:10.2202/1559-0410.1156.

Setiono, Rudy and Liu, Huan. Neural-network feature selector. *IEEE Transactions on Neural Networks*, 8(3): 654–662, May 1997. ISSN 1045-9227. doi: 10.1109/72.572104.

Shimodaira, Hidetoshi. Improving predictive inference under covariate shift by weighting the log-likelihood function. Technical report, The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-ku, Tokyo 106-8569, Japan, 2000.

Tolles, Juliana and Meurer, William J. Logistic Regression: Relating Patient Characteristics to OutcomesReview of Logistic RegressionReview of Logistic Regression. *JAMA*, 316(5):533–534, 08 2016. ISSN 0098-7484.

doi: 10.1001/jama.2016.7653. URL https://dx.doi.org/10.1001/jama.2016.7653.

Übeyli, Elif Derya. Combining recurrent neural networks with eigenvector methods for classification of ECG beats. *Digital Signal Processing*, 19(2):320–329, 2009. ISSN 1051-2004. doi: 10.1016/j.dsp.2008.09.002. URL http://www.sciencedirect.com/science/article/pii/S1051200408001516.