Analysis over the box score and standing statistics from the

NBA between 2017 and 2018

1. **The task**

This project consists of comparing two machine learning methods on a specific dataset considering into account the following aspects:

* dataset description
* description of the implemented machine learning methods
* figures and/or tables with results
* comments on the results
* conclusion

In this report, I have decided to study the performances of the **KNN** and **SVM** classifiers in deciding the **game winner** based on game statistics.

1. **Dataset description**

The analyzed dataset is based on box score and standing statistics from the NBA. More precisely, the dataset contains the game statistics of the teams that have played matches throughout 2017 and 2018. Calculations such as number of possessions, floor impact counter, strength of schedule, and simple rating system are performed. For more details, I attached to this report a PDF that contains the metadata of the dataset.

**Feature engineering**

Even though the dataset contains numerous features, not all of them provide useful information in classification process. For example, the column “teamASST%” holds the shooting percentage of the team and is calculated as the number of assists (“teamAST”) divided by the number of fields goal shots(“teamFGM”). Since I use both “teamAST” and “teamFGM”, it makes perfectly sense to drop “teamASST%” which calculates only a percentage. Same reasoning was applied to other columns which calculate sums or percentages (figure 1).



Figure 1: Dropped columns since they calculated sums/percentages

During dataset analysis I also observed that columns “seasTyp”, ”teamPTS8” and “opptPTS8” contain only 1 unique value, so I dropped them as well because they do not bring information. I also dropped “teamPlay%”, “teamFIC” and “teamTS%” due to a correlation over 0.9 with “teamFGM”, “teamPTS” and “teamFG%”, thus they do not add information.

Also, from the data analysis it can be observed that officials’ names, team names or game dates are, naturally, not that important for analyzing a game result. Thus, I decided to drop every column that is correlated less than 0.35 with the game result.

The final features selected for the classifying process can be observed in figure 2, and they do have a strong correlation to the game result as it appears in the correlation matrix from figure 3.

The dataset was balanced regarding the number of wins/loses, so no further processing was required.



Figure 2: Columns that are correlated to the game result

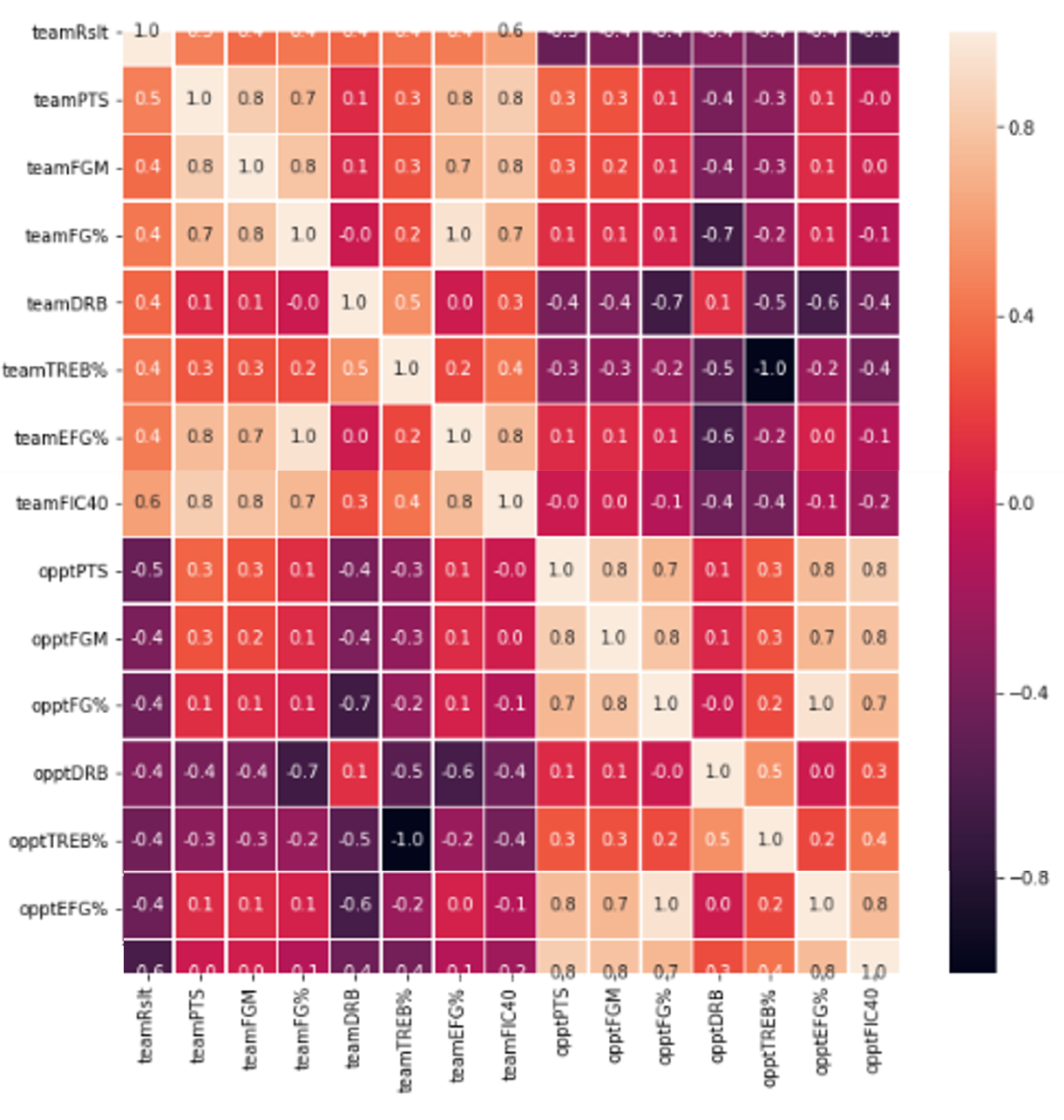


Figure 3: Correlation matrix of the final features

1. Description of the implemented machine learning methods

**K-nearest neighbors (KNN)**

KNN is a supervised learning method used for both classification and regression problems. In the case of classification, when inferring a new point, the algorithm returns the majority label of the first K nearest neighbors. The idea behind KNN is that it is unlikely to make a clusterization for the training data which does not remain valid for testing data.

The method uses a given distance function to find the neighbors, usually Euclidean or Manhattan. KNN have a high computational time, since it needs to calculate the distance to every point in the space needs to be considered at inferring. To remove this inconvenient, the space data is maintained under a tree structure for more efficient searches.

K is the most important parameter of the KNN. Setting a value too small for k could cause problems, especially when a point is equally departed from multiple points. However, setting a value too big for k could escalate in just one or two huge clusters.

**Support vector machines (SVM)**

SVM is a supervised learning method used for classification, regression and outlier detection. The problem that SVM is trying to solve is finding the maximum margin hyperplane. The points that give the maximum margin hyperplane are called support vectors.

SVM can only perform classification if the data is linearly separable which most usually is not the case. To bypass this issue, the SVM allows using the “kernel trick” which is mapping the input into a higher-dimensional feature space which might be linearly separable. Also, soft margin SVM can be used which allows the model to perform some mistakes at some cost.

Depending on the difference between the number of features versus the number of samples one can prefer to use SVM in primal form or in dual form. However, if the number of features is much greater than the number of samples, regularization is crucial to avoid over-fitting.

The most important hyperparameters are C which is the value of penalty (strictly positive) in the case of soft margin and the kernel function which tells in which feature space is the problem solved.

1. **Figures and table with results**

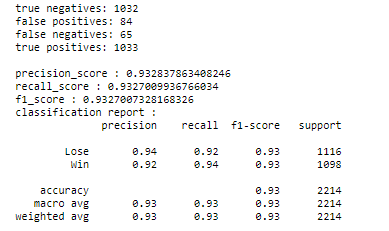
**KNN**

Figure 4: KNN score

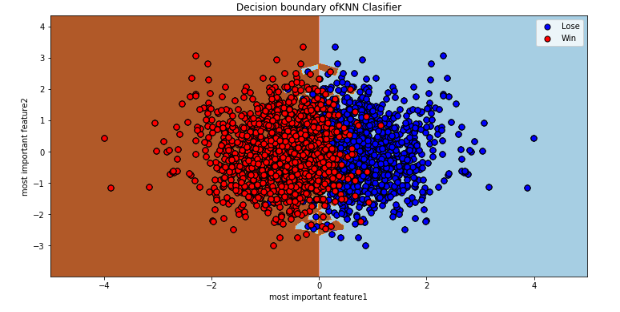


Figure 5: Decision boundary of KNN classifier

Best hyperparameters:

* **K** = 15;
* **Distance used** – Euclidian

**SVM**

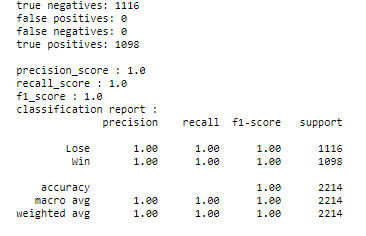
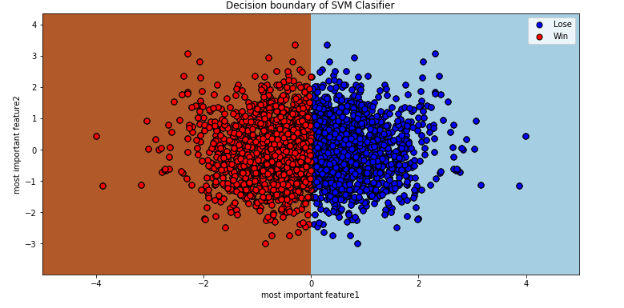


Figure 6: SVM score

Figure 7: Decision boundary of SVM Classifier

Best hyperparameters:

* C = 3
* Kernel – ‘linear’

1. Comments on the results

The high accuracies obtained from both KNN and SVM confirm a good selection of features for the classification process.

One probable explanation for the 0.93 score of the KNN is the huge number of samples which are situated near or even on the decision boundary of the KNN (figure 5) which makes harder or even impossible the selection of the best K. Other possible cause is the distance function used. I have experimented only with Euclidian and Manhattan distances. Considering the high dimensional space of our dataset, further experiments with multiple distances could improve the KNN’s accuracy. Analyzing from a sport perspective, the fact that the model does not consider the data to be linearly separable it’s not necessary a mistake. There are few cases when the team with the better stats loses.

The SVM gives a perfect score, which leads to the conclusion that the dataset is linearly separable. This proposition makes perfectly sense, especially when we take into consideration that a couple of the features that led to the classification process are based on ‘teamPTS’ which stands for total points scored by team, which in reality is the determinant of the game winner. Experiments were conducted without involving these features, and the accuracy of SVM drops to 94% while the KNN drops to only 91%.

1. Conclusion

Considering the analysis did within this report, one can conclude that both algorithms have solid performances in deciding the game winner of a basketball game when seeing the game statistics. Due to the better score and, also to the lower computational time needed, the SVM is obviously the better method in this particular case.