# Machine Learning and Evolutionary Computation Project:

Predicting the outcome of NBA games

*Biagio Licari*

*Academic year 21/22*

## 1 Problem statement

Basketball is a popular sport around the world due to the amount of uncertainty in each game. Upsets happen all the time and every team has a high chance to win any game.

The aim of this research is to find out whether it is possible to predict the result of a whole season match on the basis of the statistical performances of the teams during their last matches.

The desired output consists in predicting which team will win and the chances as a confidence estimate.

In contrast to other projects which mostly make play-off predictions on the basis of the play-off stronger data, this project will try to get good results based on the whole NBA season in line with other significant papers.

A complete overview of the project including the source code is available and can be viewed on Github.

## 2 Data & Pre-processing

In this section, the data sets and the removal of observations and variables are described. The description of the data sets includes the number of observations, number of variables.

### *2.1 Data description*

Datas are organized into two datasets which are available on Kaggle and contain statistics on matches and individual players per match starting from season 2004 to season 2021.

The challenge was to choose good data designs in order to obtain significant final data and to enrich the existent data.

| DATA SETS | N. OBSERVATIONS | N. VARIABLES | OUTCOME VARIABLE |
|---|---|---|---|
| Games | 25.024 | 21 | WINNER<br>0: away, 1: home |
| Games_details | 626.111 | 29 | - |

### *2.2 Pre-Processing*

At this stage it was decided to rule out the pre-season friendly matches and the observations dating back to the season 2009 and earlier, given that the important rules change starting from the season 2010 may have affected the result. Instead, some new and significant statistics from the two datasets were added. Particularly, a strongly used sport statistics was added for each team: this allows to obtain a team's performance index after every match during the season. This index, which is called ELO Rating, has been implemented in Nate Silver's version.

Further derived data were added to the individual players data:

-Player Efficacy: measure of the player effectiveness on the match

-PIE: measure of the player efficiency on the match

-FG Missed: failed field goal

-FG Made: succeeded field goal

-PTS_FGA: points for shot
-EFG%: % effective field goal

All data are finally combined by using the average individual performance of the players for each match in order to obtain a dataset easily combinable with the previous one.

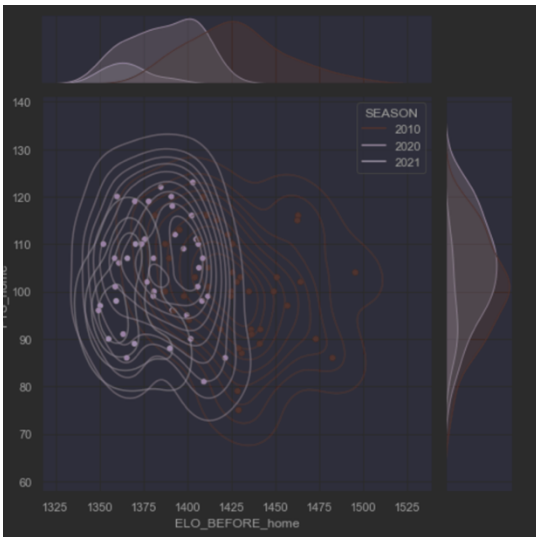

*Figura 1: Plotted ELO vs Average Points for Given Seasons of DETROIT*

2.3 *Final Data*

The final dataset was obtained by combining the two pre-processed datasets. The selected design has given life to a dataset where each match has been assigned the average performance data of *N* games of the two considered teams.

The obtained dataset is composed as follows:

| | N. OBSERVATIONS | N. VARIABLES |
|---|---|---|
| *Nba Recent Perfomance Data* | 14.145 | 68 |

# 3 Proposed solution

The proposed solution requires the use and the comparison of two statistical learning techniques, Logistic Regression and SVM, which are considered as very effective for this task.

Considering the different ranges of dataset values, it was important to apply a robust scaler to the outliers in order to scale data according to their IQR range.

Before executing the model training and considering that the dataset turned out to be moderately unbalanced, it has been decided to use an OverSampling/UnderSampling advanced technique, which is called SMOTE TOMEK-LINK, in order to obtain a more balanced dataset.

*3.1 SVM*

For the SVM Classifier a sklearn library was used.

The SVM is trying to find an hyperplane dividing the data into two categories. It solves this problem:

$$\begin{aligned} \min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i & \\ \text{subject to } y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i, & \\ \zeta_i \geq 0, i = 1, ..., n & \end{aligned}$$

The decision was made using this function:

$$\operatorname{sgn}(\sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + \rho)$$

Where the used function kernel K falls back down to its linear ($\langle x, x^{'} \rangle$) and polynomial ($(\gamma \langle x, x^{'} \rangle + r)^2$) form.

### 3.2 Logistic Regression

Also for the Logistic Regression Classifier a sklearn library was used.
It is a predictive analysis algorithm based on the concept of probability. This model uses a complex cost function, which can be defined as the 'logistic function' instead of a linear function.

For both models, the hyper parameter was executed via Randomized Search on a multitude of increasingly stringent parameters until optimal parameters are achieved.
The Randomized Search was in its turn executed via a Stratified Cross Validation including 3 repetitions and K fold equal to 10.

## 4 Evaluation procedure

The work done can be evaluated through several evaluation procedures. In this case, the cross-validation approach was adopted: particularly, the stratified K fold technique was used to get well-balanced and strong data for every fold and evaluate the performances of the compared models.

To achieve this, 2 evaluation indexes were used:

-*Accuracy*: measure of the right predictions of the model on every observation
-*AUC-ROC*: a performance measurement for the classification problems at various thresholds.
ROC is a probability curve and AUC represents the degree or measure of separability.
It tells how much the model is able to distinguish between classes.

Furthermore, these indexes allowed to compare the obtained results with the results of previous projects using the same assessment technique. The final obtained results turned out to be in line with the previous project ones.

## 5 Result and discussion

The final results turned out to be in line with the previous papers on the NBA matches predictions. Specifically, it is possible to notice how the evaluation indexes of the model using SVM and especially the linear kernel, succeeds in obtaining better results compared to the results obtained by LR or polynomial SVM. The average score obtained via cross-validation is shown in the table that follows:

| ***MODEL*** | ***ACCURACY*** | ***AUC*** |
|---|---|---|
| ***Logistic Regression*** | 0.67<br>std 0.0143 | 0.73<br>Std 0.014 |
| ***Linear Svm*** | 0.673<br>std 0.0132 | 0.732<br>std 0.014 |

| | | |
|---|---|---|
| ***Svm Rbf Kernel*** | 0.672<br>std 0.014 | 0.731<br>std 0.014 |

Once obtained, the model was tested with the data which haven't been used during the training and cross-validation stage. An example is shown on Github where the probability of the event is indicated together with the match predictions .

Any additions to the dataset, any use of features reduction or more advanced ML techniques could lead to better results even though the result obtained in the current project stays in line with the past publications on the whole NBA season predictions.

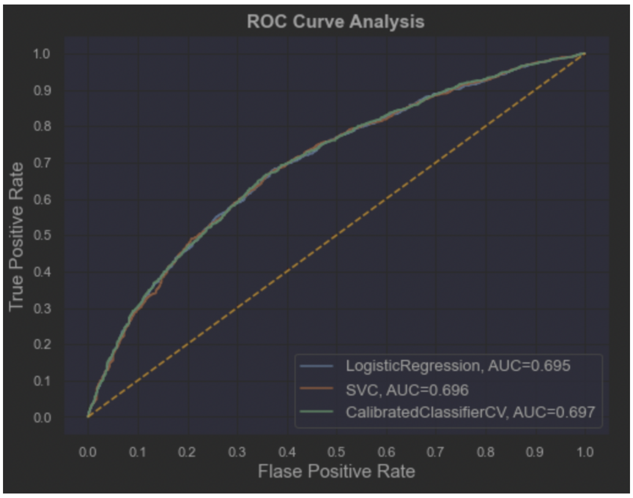

Figure 2: ROC_AUC Curve comparison btw model

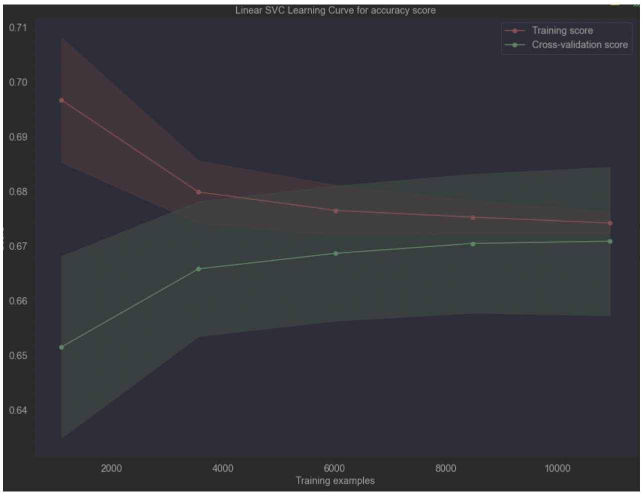

Figure 3: Learning Curve Linear SVM

## References

Josh Weiner (2020). *Predicting The Outcome Of Nba Games With ML*

Eric Scott (2016). *Predicting Outcomes Of Nba Basketball Games*

Nba Analytics 101

Paola Zuccolotto, Marica Manisera (2019). *Basketball Data Science*

Torres, R (2013). *Prediction of NBA games based on Machine Learning Methods*.

Nate Silver. *How Our Nfl Predictions Work*

Matthew Houde (2020). Predicting the outcome of NBA games

Biagio Licari. NBA Match Prediction