REPORT ON LISA EXAM PROJECT

Alessandro Doretto (873088) - Enrico Bagattin (875241)

Università
Ca'Foscari
Venezia

H-FARM®

Phase 1, Introduction to the project:

Our project is based on realizing a prediction algorithm to “beat bookmakers” and gaining an advantage in making choices while betting on matches. We choose tennis as a sport where betting in for some reasons:

1. Variables in tennis are easier analyzable than other sports since we have found the elements of a good betting-strategy and this ones are more distinguishable and “evident” than other sports (for example the surface, the winning streak) ;
2. Tennis is a zero-sum-game and this is a prerogative for applying powerful features that we have decided to use such as Elo rating for example;
3. The datasets on the internet that we have found are well-made.

In this project we will first of all import the dataset and clean the data, then we will proceed by adding new features to improve the accuracy of the predictions. After a fine tune of the best classificator, we will conclude with a study on how to maximize the return on investment for a betting strategy.

Phase 2, Dataset and data cleaning

Analyzing the datasets available on the Internet, we have searched for datasets with the largest number of significant variables to improve the accuracy of our prediction model.

The first step has been to load our datasets (Tennis-Data) and check if there are some possible problems that we could face in the future steps like missing data.

Our columns at the end of those processes were:

- **Location**: Where the match took place
- **Tournament, Series, Round:** Kind of tournament or series where the match is part.
- **Court, Surface:** A variety of surfaces can be used to create a tennis court, each with its own characteristics which affect the playing style of the game.
- **Winner, Loser, Rank, Points**: The rank based on points gained of both winners and losers at the beginning of the match
- **Comment:** If the match has been concluded or not.
- **B365, PS, Avg**: Odds of bookmakers (Bet365, Pinnacle)

The first step in data cleaning has been to remove the Winner and loser reference by changing all column references such as winner or loser points.

The second step has been to fill the absent values in our dataset. There were missing values for rank and points column, we solved it by simply putting as defaults 0 and the maximum rank.

There were also missing odds for some bookmakers and, for this reason, we chose to consider only the two bookmakers with less null values (Bet365, Pinnacle) and the average odds between bookmakers. To fill the missing values we made a function called **“findOddsForRow”**. This function allows us to search for similar matches (based on the rank of players) and then use those odds to fill the missing ones. Doing this we could avoid deleting it and have at the same time reliable odds inside the dataset.

The next step was to perform the *one hot encoding,* in this step we decided to perform it also on the player's references other than for the categorical variables. By splitting all the players into columns we let the classifiers have the possibility to analyze also the historical matches for the two players (rivalry).

Phase 3, The features:

We decided to add five main features to our project based on the relevance that they could have in the tennis sport. We have found some variables that could, at least for us, add a concrete value for our analysis.

The first feature that we added is **Elo rating**. It's a method for calculating the relative skill levels of players in a zero-sum game. In our "data cleaning" process we decide to perform the Elo rating method in order to establish a rank of players and a relative rate. We made that in a progressive way because, after calculating Elo, we store it for the next match to be able to assign the rating of the players at the beginning of the match.

The second feature that we calculated is the **number of matches played** during the previous year, that will be important for the analysis because it allows having an indicator of the strength of the player, and also if it's a new player or a well known one.
Similarly to the previous, we also added the **percentage of matches won** during the previous year to add an indicator of the performance of the player.

Another feature that we choose to add has been **injuries**. They were calculated by taking the matches in which the player retired or walkover in the past three months in order to have an indicator of the conditions of the player and establish relevance to it.

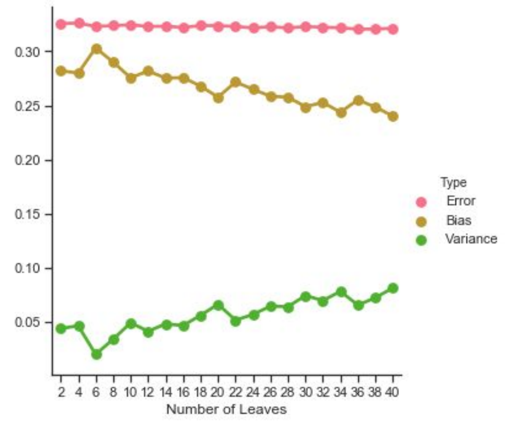
Finally, we added the **winning streak** because we think that the series of wins of a player could be a factor that matters in the outcome of a specific match. This is because we assumed that a player's series of wins could influence in a psychologically positive way the outcome of a match.

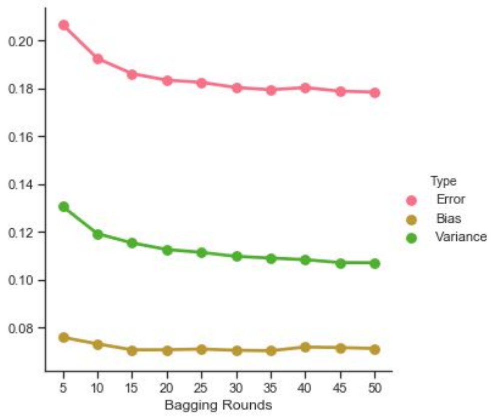
Before splitting data in training, validation and test we made another step: in order to balance both match outcomes for our prediction models we **duplicated each row**. We have done it by switching all the player features for each duplicated row and adding a Winner column for the match result.

Phase 4, The prediction models:

We started by calculating how strong were the bookmaker's algorithms by looking at the odds. This delimited the baseline, with an accuracy of 68%.
After testing **K-Nearest Neighbor** and **Naive Bayes** Classifiers, we have decided to not take them in consideration because they demonstrated poor efficiency and a minimum margin of improvement. On the other hand, the **Decision Tree** classifier had good results (comparable to the baseline efficiency).

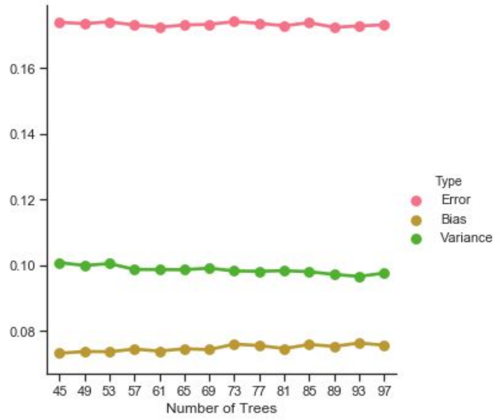
After a bias and variance evaluation we saw that the variance was increasing with more leaves, so we tried to apply the bagging to obtain new improvements.

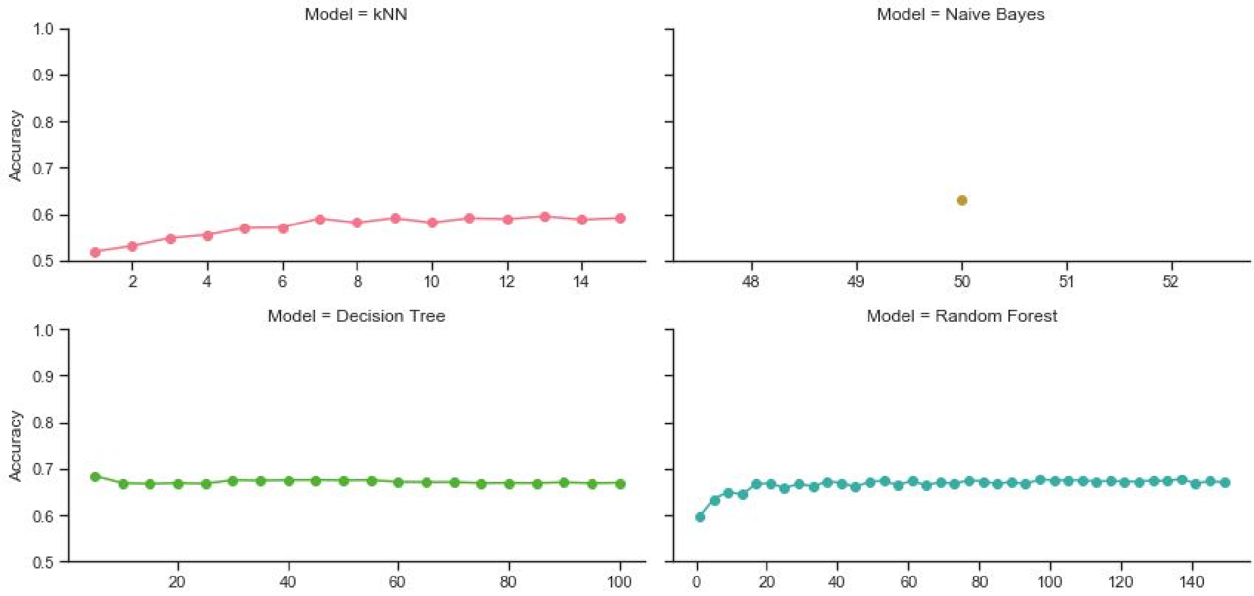




After the satisfying results with the decision tree and bagging we decided to move on **Random Forest** to have a more powerful model. With RF we have found that it had a very stable curve of bias and variance (when the number of trees was increasing).

We reached a fine tuned version of RF using two sklearn tools: Randomized Search and Grid Research.





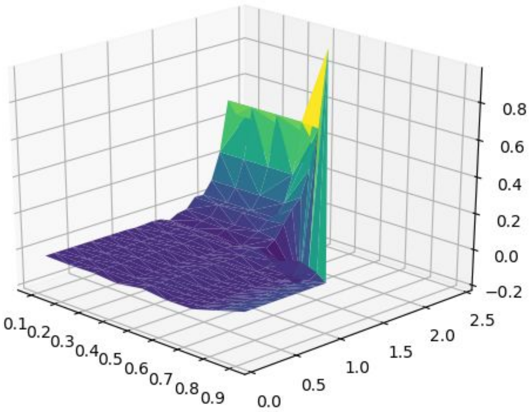
We have also tried to increase the accuracy by removing some features with the Recursive Feature Elimination. With this analysis we have found that our features were not too powerful, and our accuracy was based a lot on bookmakers odds. Secondly there were ranks of players, points, matches won, Elo rating and matches played.

Phase 5, The betting strategy:

After having maximized the prediction accuracy, we studied a way to have a great ability to find good matches to bet on and simulate the results of the strategy.

Taking into account our studies we can say that finding a perfect balance between the accuracy of prediction and the number of matches where we choose to bet on, is also a fundamental task to maximize the profit.

We have started the strategy formulation by putting a “baseline”: bet always on the smaller odd player. The result was surprising, the final ROI was -1.87%.
Then we used our classificator to build our strategy, taking into account some parameters: the probability assigned for the predictions (by the classifier) and a minimum odd level. We put a minimum odd level because very low odds, like 1.1 could lower our ROI, because the gain is very low compared to the loss. So we preferred fewer matches with a good probability and a good return for each match if we predict right.



To find the correct balance between these two parameters we took the validation portion of the dataset and we tried to maximize the ROI with each combination. We found that a good minimum odds level is between 2.1 and 2.3, and a minimum probability of 0.5, that’s the little yellow triangle in the image.

Finally, we tried the best parameters with the test portion: 19.23% of ROI obtained with 22 bets.
We beat bookmakers!

Phase 6, Problems and solutions found:

A problem that we have found during the accuracy testing with the decision tree was that the accuracy of our model was of 100% in almost all the tests and since it was a quite unrealistic output we decided to go much deeper.

The first idea was that something went wrong because the sample taken in account was part of the train. For this reason we have tried to detect if the “winner” column was not “dropped” but we have found that it has been done a few lines above.
After that we restricted the search and we have found that the problem must be on “injuries” or “winning streak”. At the end we discovered that the problem was “winning streak” since we assigned it taking into account the result of the match. This meant that the classifier knew in advance who the loser was because he had the streak reset, and the accuracy result was “perfect”.

Another problem that we have faced has been the clusteing analysis. We have performed it to see if we could determine where our classifier works the best and if it is able to isolate cases with wrong prediction to build a low risk betting strategy. Our purpose was also to understand if there were features that bring him in error and in this way reducing our accuracy. To our amazement, we have discovered that clustering seems to be useless. Even if we tune parameters the shape of the data between right and wrong predictions were very similar, so we couldn't find some independent cluster to work on for improving our strategy.

One big problem faced during our project development, has been the slowness in the computation of some processes. This has been caused by the amount of data processed and the heaviness of some tasks, for example, the ones regarding the randomized search that took a long time to be executed.
Given the relevance it has become necessary to use a more powerful device with 8 virtual threads cpu allowing us to go on with our work in a faster way.

Phase 7, Conclusions:

After performing our multi-step analysis we found out that our model returns pretty good confidence and a ROI of 19,23%, that seems to be a pretty good score considering the one of bookmakers. Odds of matches where we bet are listed at least 2.2 (this means that the amount wagered is paid in case of winning 2.2 times) we discover that a high odds is necessary for the sustainability of our betting strategy in order to smooth out any leaks.

We think to have reached the maximum result obtainable with these data. In our opinion a more consistent amount of features could be the key to beat bookmakers, but also having more data could give more stable results when building a strategy.
Taking into account a deeper analysis of the historical matches between two players, and also their physical and mental status, could also be a good point for improvement. We are anyway proud of our work, to have reached a good result with not too powerful features.