DEPARTMENT OF INFORMATICS, CA' FOSCARI UNIVERSITY

Web-Intelligence Project report

Flavius Adrian Holerga
January 15, 2020

1 IDEA AND PROJECT PLAN

Goal: building a program that uses studied techniques to predict the winner of the Australian Open 2020 competition.
Approach: implementing two different algorithms and tuning them in order to get a more accurate score. Observe the results and repeat.

For maximum efficiency, I have split my workload in the following main categories. I spend the first section observing and making assumptions on the data-set. In the second section I modify the table to aid examining the assumptions made previously. Then in following sections 3 and 4, I discuss the results obtained on two algorithms: Logistic Regression and Random Forest respectively. In the 5th section I discuss the possibility of removing certain attributes from the table based on a Recursive Elimination algorithm. And finally, in the last section I discuss the final results and the effects of picking the best data-sets from the bunch has an impact over the prediction.

Afterwards, in the second part I briefly discuss implementation issues that I have encountered along the way and possible present future improvements to increase the accuracy of the model.

2 PART 1: DISCUSSING THE IDEA

2.1 DATA-SET ANALYSIS

For the dataset I have used the player rankings from 2014, up until 2019["*http://tennis-data.co.uk/alldata.php*"].

The headers of the table include information such as the names of the players from a given round of a given tournament. Furthermore, for each match we know the winner, the rankings and points of each player and the terrain conditions (e.g. indoors/outdoors, soft/hard floor). Likewise, we also have data regarding the played sets and the progression in the current tournament.

As a disclaimer, I am by no means a tennis expert, therefore the first problem I encountered was making sense of data I did not understand. To solve this main issue, I learned how the game works and initially I dropped the information regarding the number of sets played and won by each player to make it easier for me to digest the relationship between general data (such as the Rank and Points of each player) and the conclusion of a match (if the game was won or not).

After making a couple of assumptions I have decided to drop the following attributes:

- **ATP:** I didn't completely understand the meaning and found it to be irrelevant to the outcome of a match
- **Best of:** The majority of entries were already categorized as 'best of 3'
- **SetsW/SetsL:** As previously mentioned, I discarded from the start the information regarding the sets
- **Betting info:** Initially I discarded the betting information, but soon I learned that it has a huge impact on the prediction accuracy since the data is obtained through many more factors than those available in our dataset. Therefore I only dropped betting information attributes that can not be found in the 2019 record.

2.2 Preparing the dataset

Since I have a collection of entries, I have quickly noticed that no entry has information regarding the outcome of the other games involving the same players(e.g. have the same players played against each other before?, how well have they played last year?). The initial assumption that I made here is that the performance of a player can vary a lot from year to year, especially when it comes to sports where the environment is always in a constant change(e.g. if a player was injured in 2016, it is unlikely he will fully recover for the cup in 2019). Therefore I have considered adding the following columns into my table:

- **P1dW/P2dW:** Number of total wins up until the game (here I consider all the games ignoring the fact that a player's performance may deteriorate over time, for the sole purpose of comparing it with a date dependent attribute.
- **P1dR/P2dR:** The ratio between wins and loses up until a date

2.3 Logistic Regression

After I have finished preparing the data I was ready to run my first test on Logistic Regression. The reason I have picked this method is due to the fact that in the beginning of the project my lack of understanding of the subject has lead me to seek information on the internet and I found this method as straightforward and a naive and simple implementation for my case initially. Compared to Random Forest, it is clearly behind in terms of accuracy in this specific case, as the score will soon demonstrate.

After getting over some implementation issues I will discuss in the second part, I obtained the following scores:

Score	Context
0.654	'Ranks"Pts','Court','Surface','Round','P1dW','P2dW' and betting with **lbfgs**
0.653	for same with **newton-cg**
0.638	for one extra column depending on P1dW and one for P2dW
0.573	without betting
0.499	using just the ratio and more betting tests

As it can be observed from the table above, the scores dropped significantly when the betting information was eliminated. Thus I consider my assumption made initially regarding the betting information as valid for this case. Furthermore, I have observed the impact of the total wins of a player on the score. Likewise, when this column was taken away, the score dropped to just 0.499.

Therefore, I have kept most of the Columns mentioned initially and dropped the ratio attribute for this test. Furthermore, I have increased the magnitude of the number of total wins by squaring it. I did the same for the ratio. This change resulted in a slight increase in score.

2.4 Random Forest

For Random Forest I have taken the same approach as I did for Logistic Regression such that from the same data-set, on the first test, the result obtained was 0.789. Since I was satisfied enough with the result I moved on initially, but came back after some testing and managed to increase the value up to 0.798 after increasing the magnitude of the 'Round' attribute.

2.5 Recursive Elimination

To further improve the accuracy of the model I have decided to use the Recursive Elimination method to select the best k features to make a prediction on.

I have tested for a couple of different k values and the results seem to be self-explanatory. The results obtained are as follows:

1. By far, the most impactful feature is always given by the **Points** or the **Rank**, but never by both. After some digging I have realized that Rank is computed using the points of a player amongst other criteria which is the reason they are direct proportional, such that using both in order to make a prediction is simply to put it, wasteful.

2. The second most important criteria was always **P1dW** and **P2dW**, which as previously defined state the number of matched won in the past up until the current game. This translated directly to a players experience. So far, using this model it seems that players who have played well in the past are likely to play very well in the near future as well.

3. The criteria obtained from this point on varies a lot on a selection of different **betting** sources. What I found interesting is that on all conducted tests, there were always mismatched betting sources such that when a betting feature was selected from 'Max' for Player1, 'PS' was selected for the other.

In the end, I have chosen 8 features to make the prediction on. These are depicted in the following image:

```
['WPts', 'LPts', 'P1dW', 'P2dW', 'P1dR', 'PSL', 'MaxW', 'MaxL']
```

2.6 Testing on the 2019 dataset

When I first started working on the project, the 2020 dataset was not yet available, so initially I considered the 2019 year only for testing. Therefore the training was done as follows: The dataset was composed of all the data from 2014 up until 2018 (the data before 2014 had a couple of differences regarding betting attributes and it complicated the implementation a little bit; furthermore, picking too many years for training could result in player data being too spread out since in that timeframe careers were created and destroyed, which can not lead to accurate data). Then I have used a function to split the training data in a train set (75%) and a test set(25%).

At the end, the 2019 dataset was considered as a production test, which had the benefit of being able to compare the performance of the semifinalists that the algorithm has picked with the real players that have made it that far.

Score	Classification Method	Set
0.76128	Random Forest	2019 full-set
0.82677	Random Forest	2019 AO only
0.69291	Logistic Regression	2019 AO only

3 Part 2: Implementation issues

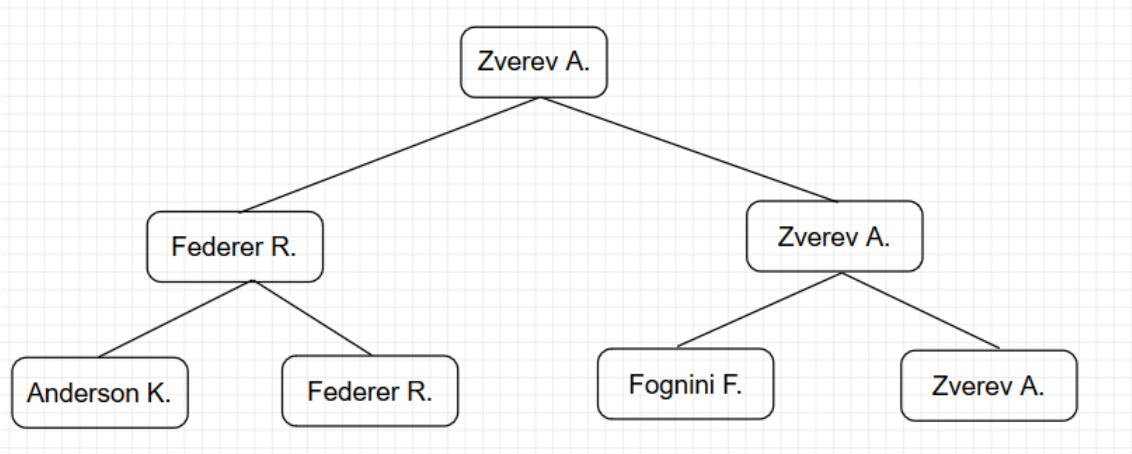
The main issues I have encountered along the way are in regards to bad planification in the beginning. Since my workflow was not well organized, I had to go back repeatedly to adjust certain bits of code. Couple errors I have made along the way:

- **Destroying my data-set during execution**: In the beginning, since I didn't know that the project required a decent amount of figuring out what works and what doesn't, I had built my model by cropping from the main data-set the "unnecessary" components. What I have realized very soon is that I wanted to check my hypothesis and verify that the components were truly wasteful. Because of this, I had to rebuild the structure of my project in order to avoid long running times every time I want to observe a case.
- **Mixing datasets:** At one point I decided to mix the women and men dataset and observe the outcome. Initially the result was not obvious so I did not consider duplicating my dataset as a problem. Later on, this caused issues with both execution time and prediction score on the men testing set.

4 Report Conclusion

4.1 Final results

In the end, we can compare the predictions for 2019 with what actually happened that year.



This is the prediction for 2019. All players listed above made it as far as 4th or 3rd round, but overall the prediction is incorrect and requires further tweaking.

4.2 Final thought

In conclusion, the project allowed me to experiment and observe first hand the different complications data classification presents. In the future I plan to continue studying the model and improve and observe different relations between the desired outcome and a set of chosen factors.