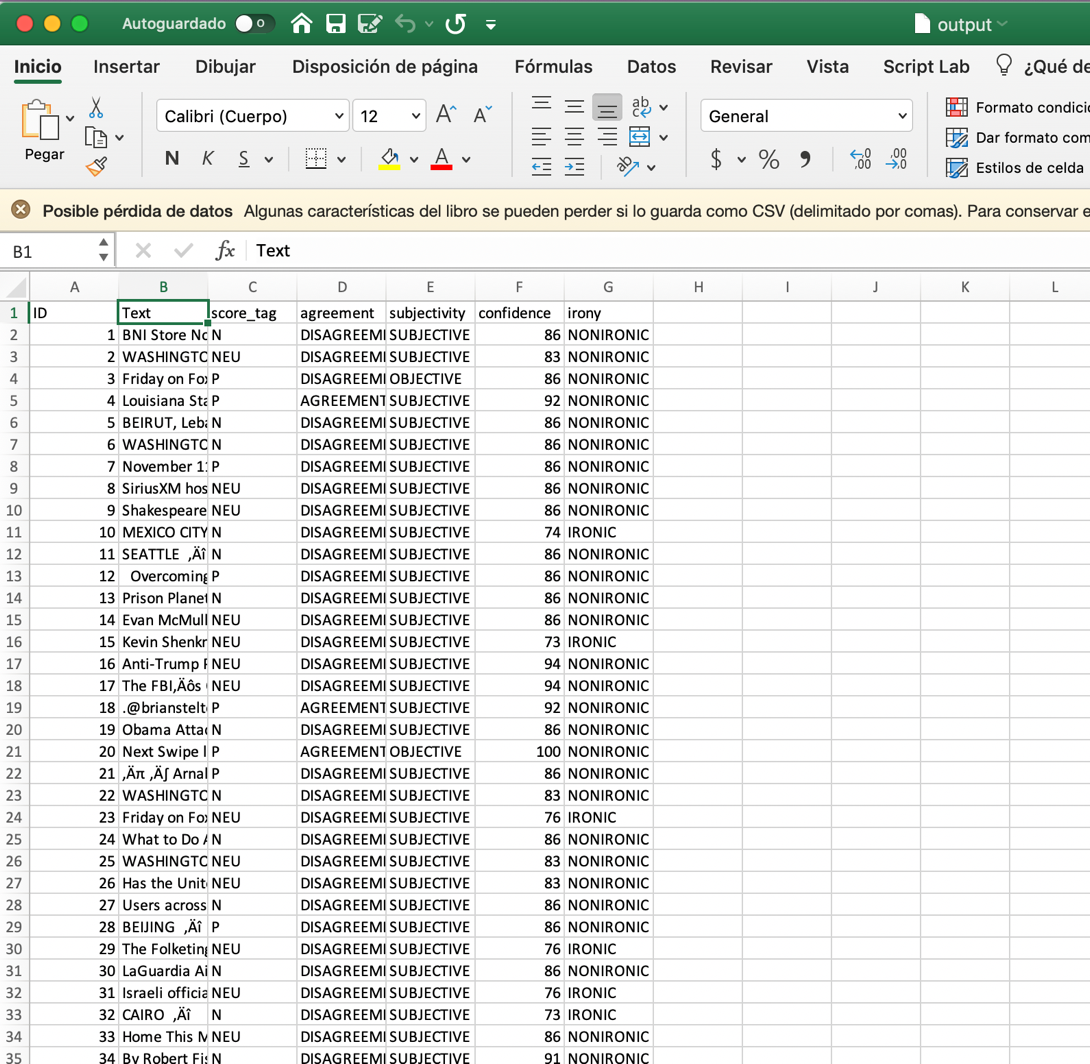
# H4 – Testing two pattern-based classifiers

**Introduction**

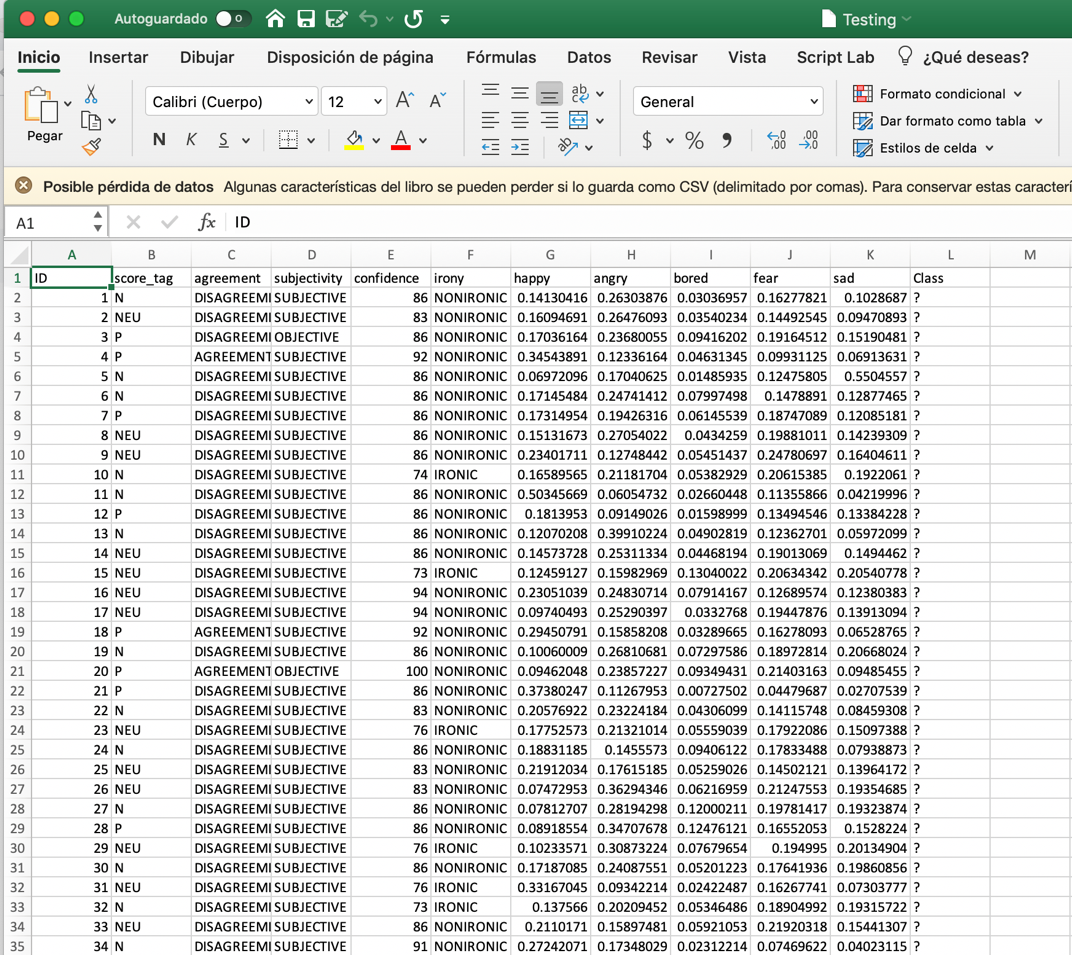
Pattern based classifiers are tools that help us to determine the class of a certain object depending on the patterns that can be identified. Examples of pattern-based classifiers are J48, RandomForest, REPTree and FURIA.

**Development**

Using the python script from HW1, sentiments were extracted from the Testing.csv, emotions were not extracted because they were extracted in the previous homework. The columns from output were pasted into the Testing.csv of the previous homework. ID and Text columns were removed. Also, a new Column named “Class” was added and filled with the symbol “?”.

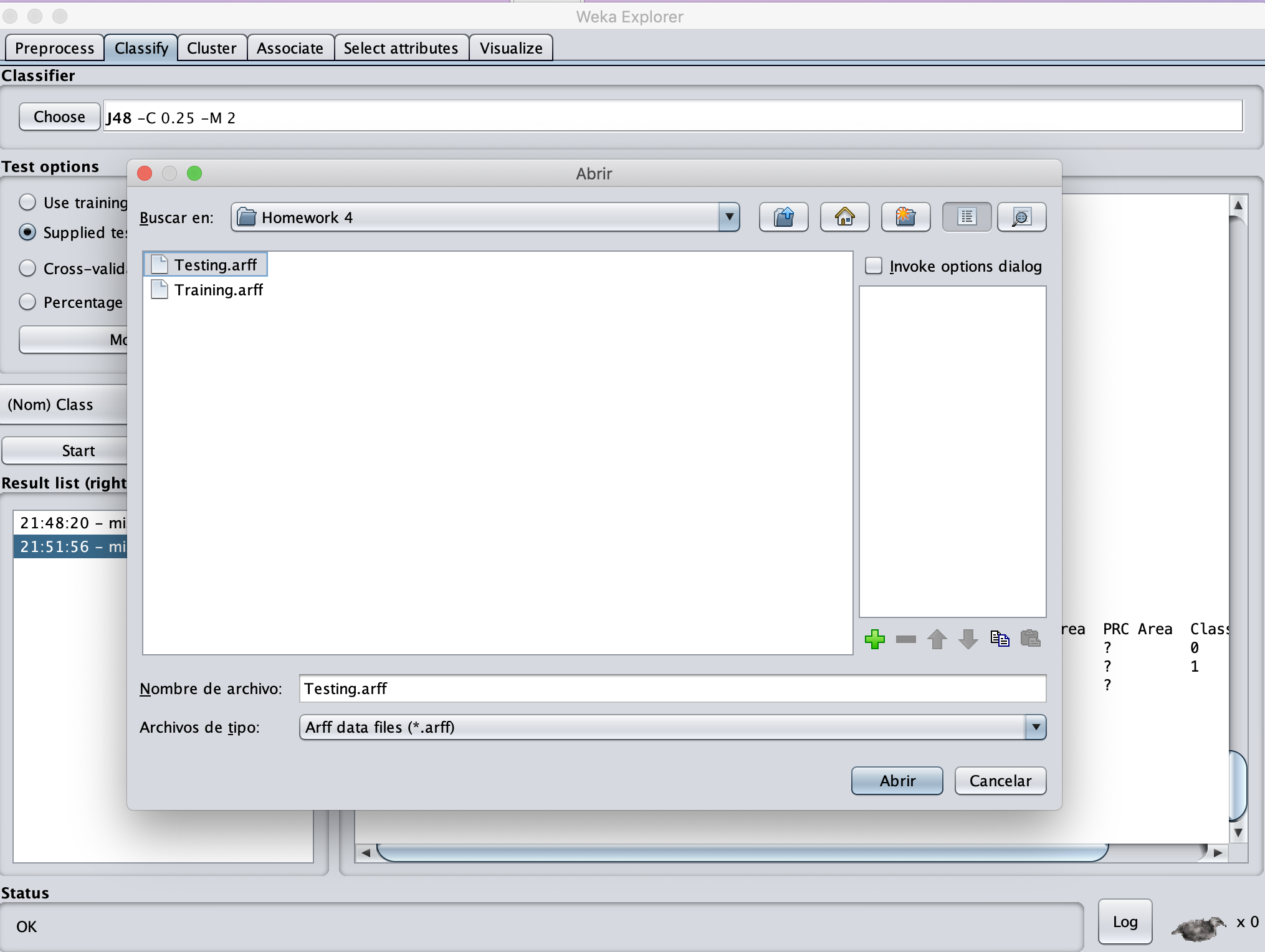


*Figure 1. output.csv features are shown.*

**

*Figure 2. Modified Testing.csv showing the new sentiment and previous emotion’s features*

Then in Weka, both Training.csv and Testing.csv were changed to arff format, it is important to clarify that in both files, The class type was changed to nominal.



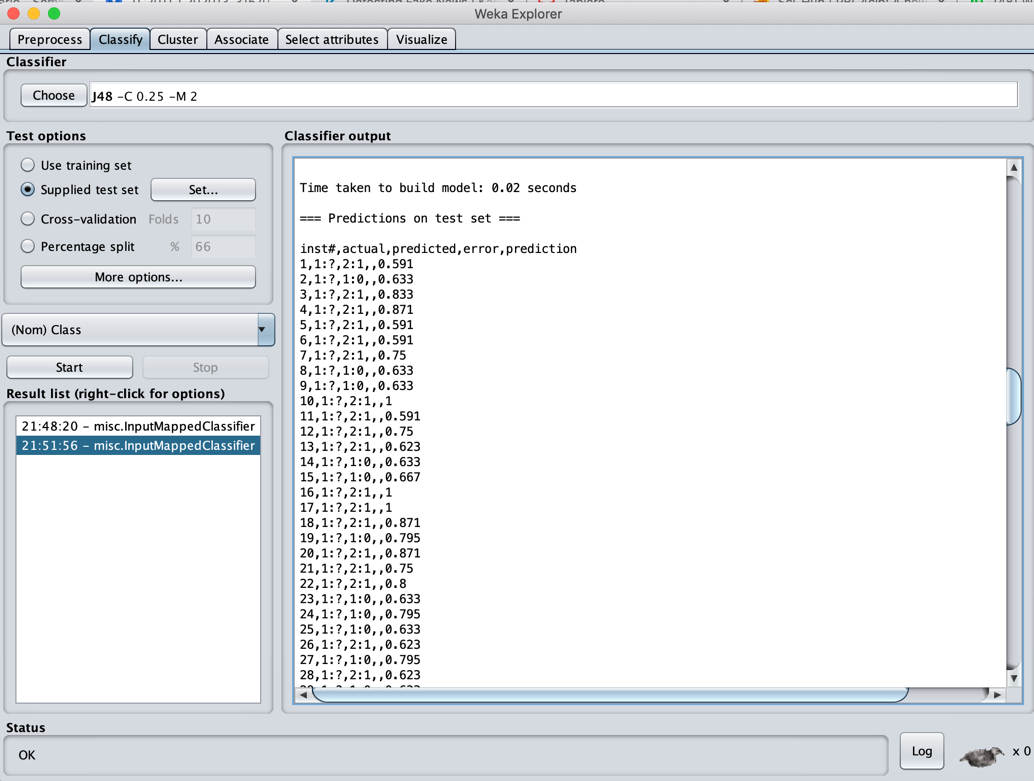
*Figure 3. Testing.arff is chosen as the test set.*

After all the above steps are made, we can pick the classifier that we want, for example J48, Random Forest, FURIA or REPTree.

**Results**

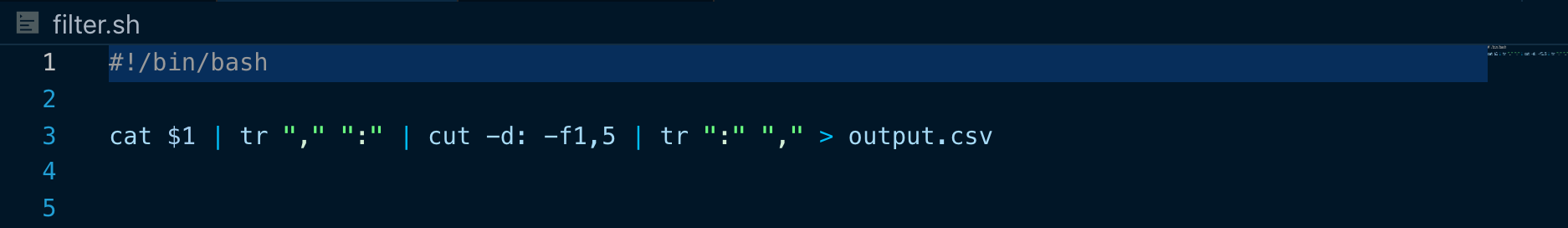
*J48*

In this case, J48 was chosen as the classifier. The output is generated in CSV format.

**

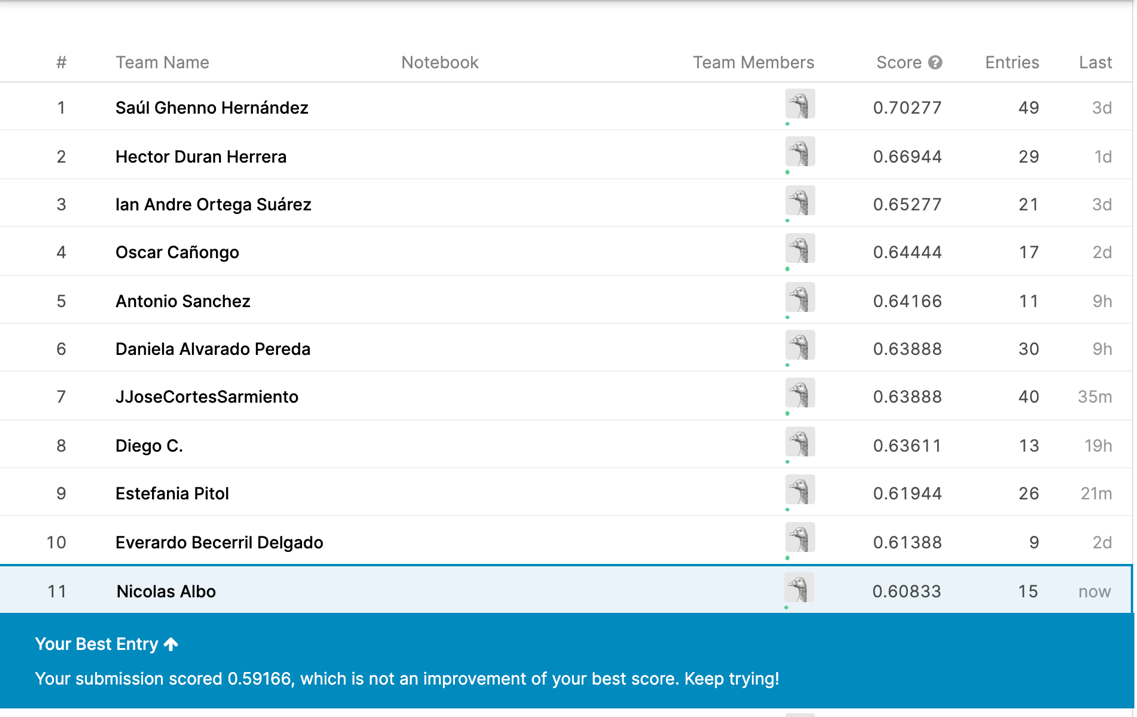
*Figure 4. J48 output in CSV format.*

The ouput from Weka was copied into a new file named J48.csv and thanks to a Bash script that I developed; the file was given the format required for uploading to Kaggle. In fact, this Bash script is also used for formatting the following output CSVs that Weka generate for the other classifiers.



*Figure 5. Simple Bash script that deletes the unnecessary columns of the original J48.csv*

Then the new J48.csv is uploaded to Kaggle for its evaluation.

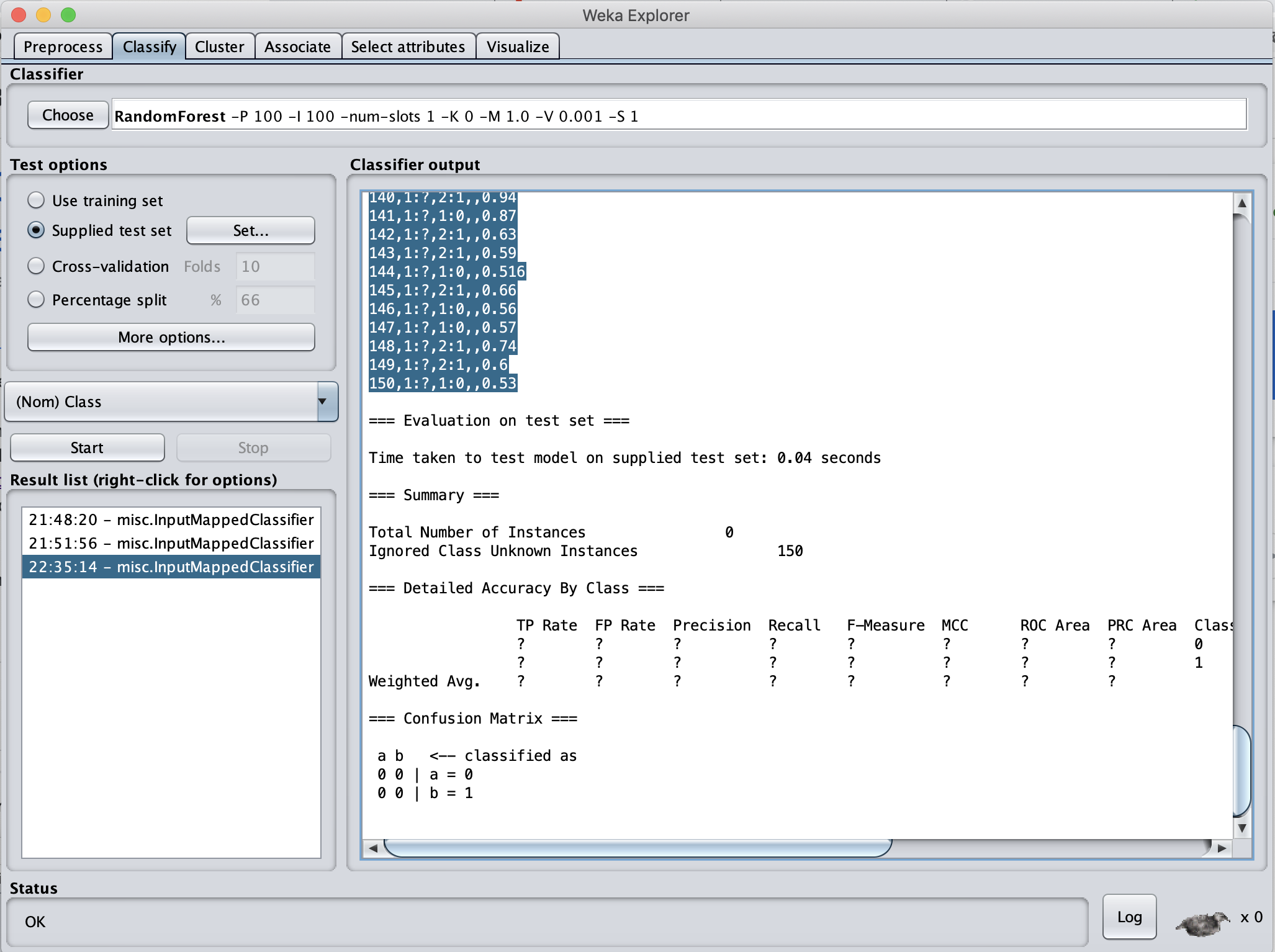
**

*Figure 6. J48 result in Kaggle.*

As it can be observed, the score obtained with this classifier couldn’t beat the score of the last entry, that was a k-NN classifier from H3.

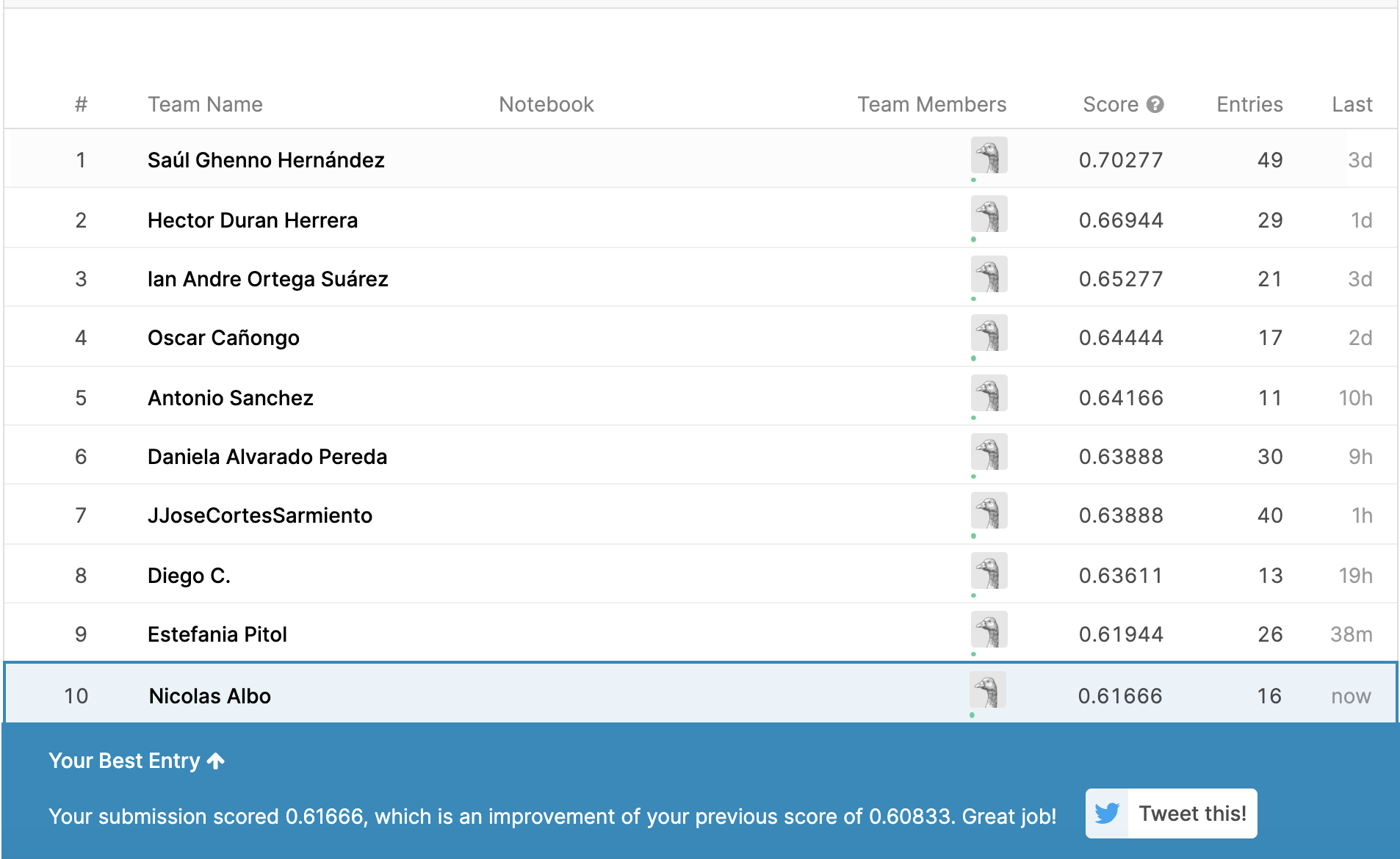
*RandomForest*

In this case, RandomForest classifier was chosen. The output is generated in CSV format.

**

*Figure 7. RandomForest output in CSV format.*

Then RandomForest.csv is uploaded to Kaggle for its evaluation.

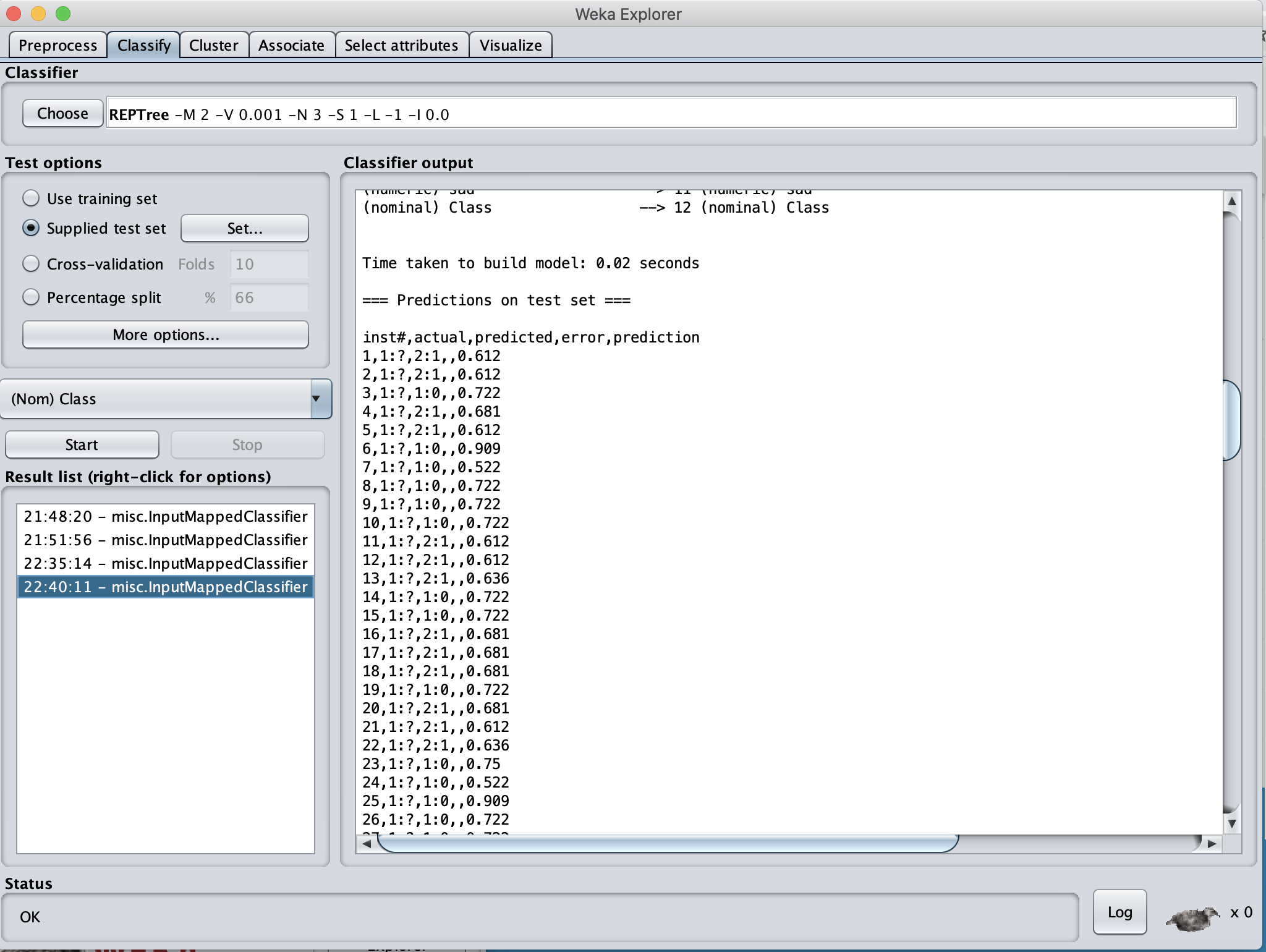
**

*Figure 8. RandomForest result in Kaggle.*

As it can be appreciated, the RandomForest classifier performed better than any other classifier that I have used in the past, therefore resulting in my best entry of all time.

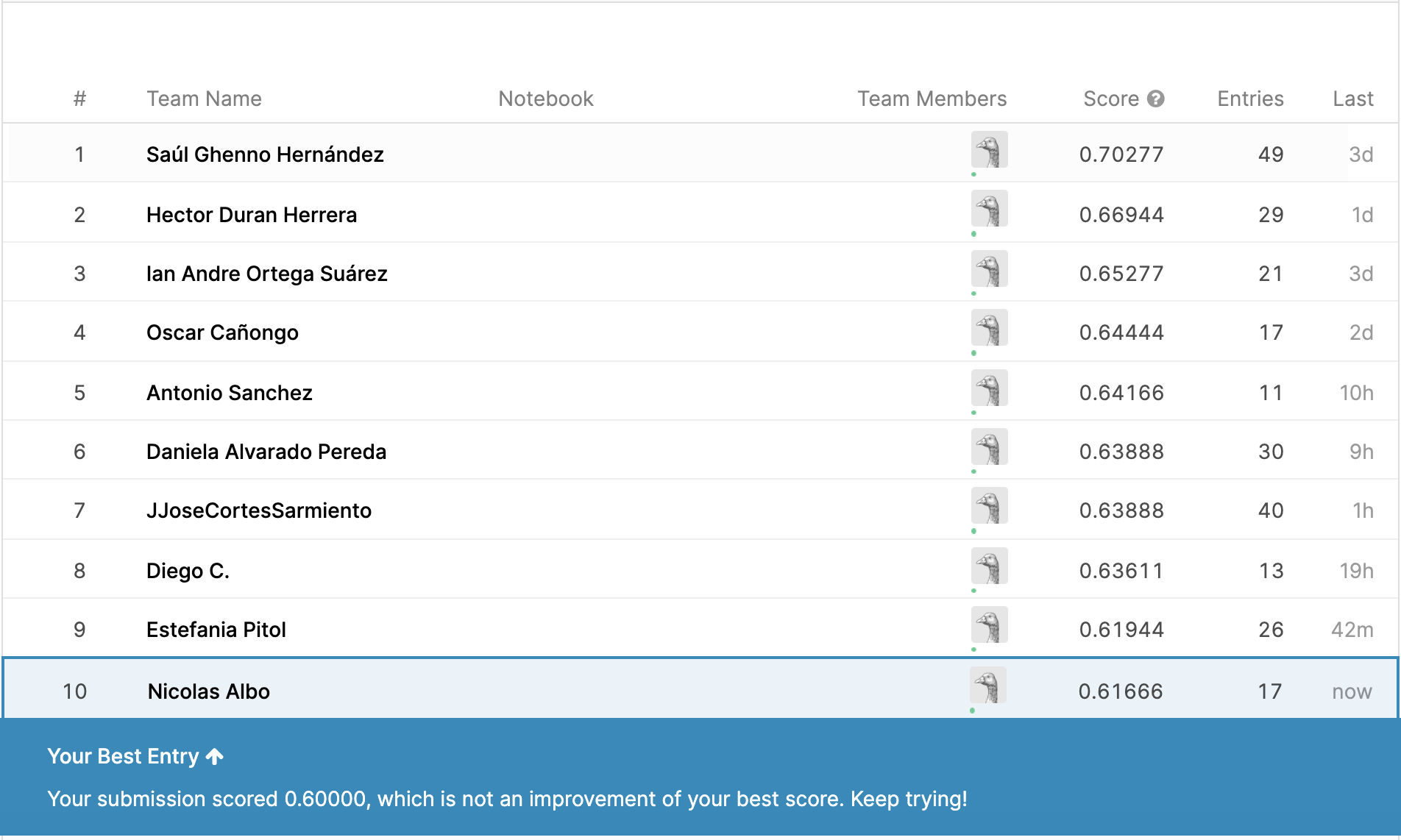
*REPTree*

In this case, REPTree classifier was chosen. The output is generated in CSV format.

**

*Figure 9. REPTree output in CSV format.*

Then REPTree.csv is uploaded to Kaggle for its evaluation.

**

As it can be appreciated, REPTree classifier did not had a results a good as the one generated by the RandomForest classifier, so my position in the Leaderboard remains equal.

**Conclusion**

Pattern-based classifiers are more useful than k-NN classifiers in certain situations. In this case, the RandomForest classifier performed better than the k-NN classifier that gave me my previous highest score in Kaggle, but, the other two Pattern-based classifiers that I used in this homework, could not beat the score produced by my k-NN classifier.