Dog breeds prediction using hyperparameter tuning in multiple classifiers

Dimitri Leandro de Oliveira Silva
dimitri.leandro@aluno.ufabc.edu.br

Simon Leroy
simon.leroy@edu.em-lyon.com

Nnaka Emmanuel Chinonso
emmanuel.nnaka@uni-muenster.de

ABSTRACT

In this work, we evaluated 6 different Machine Learning algorithms using an Sequential Feature Selection and Grid Search with a 10 Fold Cross Validation to predict 20 different dog breeds. It was shown that XG Boost was able to reach an accuracy of over 88% having low variance.

Keywords: Machine Learning, Grid Search, XGBoost.

1. INTRODUCTION

In this project, we looked at a training dataset containing around 32.000 observations, each having 100 features generated by a Convolutional Neural Network corresponding to a dog breed among 20 different ones. The dataset was far from being balanced. Breeds like “teddy” reached 23% of the total observations, while 10 others didn't have 1%. Class imbalance tends to cause difficulties in training Machine Learning algorithms, as discussed in [1].

Our goal was to predict the breeds of dogs in a test dataset containing 800 samples and no given labels. To do so, we evaluated different models in order to find which one had the best performance. Therefore, we were able to train the best model with all the training samples and use it to predict the test set.

2. METHODS

The idea of comparing multiple Machine Learning algorithms applied to a single objective is not new and can be seen in different scenarios and approaches [2]. Furthermore, hyperparameter tuning can play a key role in improving predictions, as shown in [3]. For that reason, we used a hyperparameter Grid Search to improve the performance of a set of 6 machine learning algorithms.

After normalizing the training set, we measured the Feature Importances using the Gini Index within a Random Forest. Given this result, we sorted the features from the most important to the least in order to apply a Sequential Feature Selector (SFS) for each classifier. Then, given the best quantity of features for each one, we ran a Grid Search to adjust the hyperparameters. For all the procedures, we applied a 10 Fold Cross-Validation.

It was also observed that the features were not correlated with each other. The Spearman Coefficient was 0 in every measure taken, indicating that there was no need for removing correlated features that could produce overfitting. It is important to point out that the SFS was not applied in the tree-based algorithms, since they already select the best feature to split the data in each of its internal nodes.

3. RESULTS AND DISCUSSIONS

Table 1 shows the results obtained for the classifiers after running the SFS and Grid Search. The algorithm that obtained the best accuracy value was XG Boost, not only because it reached the highest mean, but also for its low standard deviation. This result indicates that this classifier, beyond having a low bias, also had a low variance, expressing it can be robust into furthermore predictions of not known data.

The results for the hyperparameters to the XGBoost were: 100 *parallel trees*: 100; 15 *boosting rounds*; *maximum depth* of 12; 75% of *samples per tree*; 75% of *features per tree*; 0.25 *learning rate*; and *gamma* equal to 1. We believe that the usage of Bagging (*parallel trees*) and limited *depth* of the trees helped prevent overfitting, as well as the hyperparameters *samples per tree* and *features per tree* lower than 100%.

Classifier	Accuracy	Nº of features
XGBoost	0.883 ± 0.004	100
Logistic Regression	0.873 ± 0.007	80
Support Vector Machine	0.869 ± 0.008	50
Random Forest	0.863 ± 0.012	100
Linear Discriminant Analysis	0.862 ± 0.009	90
K Nearest Neighbors	0.861 ± 0.008	60

Table 1: Mean and standard deviation accuracy results for all classifiers.

Furthermore, we can mention that the SFS benefited the Grid Search as it was able to significantly reduce the algorithms' training process. Besides, a lower number of features can also prevent overfitting, but we believe it could have supported it even more if the features were not uncorrelated.

4. CONCLUSIONS

Given the observed results, we conclude that, concerning the number of data entries and given features, XGBoost provided the best accuracy among all the compared models, having reached an accuracy of over 88% and low standard deviation.

5. REFERENCES

[1] Bae, S., *et al*. Effective data-balancing methods for class-imbalanced genotoxicity datasets using machine learning algorithms and molecular fingerprints. Computational Toxicology, 2021.

[2] Shreemali, J., *et al*. Comparing performance of multiple classifiers for regression and classification machine learning problems using structured datasets. Materials Today: Proceedings, 2021.

[3] Valarmathi, R., *et al*. Heart disease prediction using hyper parameter optimization (HPO) tuning. Biomedical Signal Processing and Control, 2021.