Machine Learning and Data Mining project: Leaf identification

Matteo Marturini[1]

[1] problem statement, solution design, solution development, writing

Course of AA 2020-2021 - Master degree in Data Science and Scientific Computing

1 Problem statement

A dataset of 340 leaves from 30 different plant species is given; each leaf is described by a 15-dimensional vector, composed of 14 attributes (numerical), plus a categorical variable (the specie). The goal is to provide a classifier which, based on the leaf attributes, is able to predict the specie, hence a multi-class classification problem.

2 Performance indexes

Suppose to have a multi-class classification model that, given a new observation x, it returns the predicted class $c(x) \in C = \{1, 2, ..., K\}$.

Accuracy: To assess its overall ability to predict, the most straightforward index is the accuracy

$$Accuracy = \frac{number\,of\,correct\,predictions}{total\,number\,of\,predictions} \tag{1}$$

The domain of values belongs to the interval $[0, 1]$, where 1 is perfect accuracy and 0 is the worst result.

Macro-averages: To evaluate if the classifier is able to precisely predict each class in the dataset, the *macro-averages* (the average over all classes) for

precision, recall and F-1 score can be considered:

$$P_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_i}{TP_i + FP_i} = \frac{1}{|C|}\sum_{i=1}^{|C|}P_i \tag{2}$$

$$R_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_i}{TP_i + FN_i} = \frac{1}{|C|}\sum_{i=1}^{|C|}R_i \tag{3}$$

$$F1_{macro} = \frac{2P_{macro}R_{macro}}{P_{macro} + R_{macro}} \tag{4}$$

where P_i and R_i are respectively the precision and recall for the i-th class, considering as "positives" the observations belonging to the i-th class, and "negatives" the ones belonging to every other class (TP stands for true positives while FN stands for false negatives). F-1 is the harmonic mean of precision and recall, so the higher it is, the better is the classifier at predicting each individual class. The domain of values in all 3 cases is [0,1], 0 being "bad" and 1 being "good".

3 Proposed solution and assessment

The most suitable solution for this problem is the machine learning technique Random forest (details on algorithm in [3], chapter 15). It has been experimentally shown that Random forest is one of the "best" supervised classification algorithm [1]. However, since the performance of a classifier usually depends on the data, I compared the accuracy of Random forest with the accuracy of 2 other popular multi-class classification algorithms: the decision tree and the gradient boosting in the implementation *XGBoost* (detailed description of the algorithms can be found in [3], chapter 9 and 10 respectively).

One of the most common technique for assessing a classifier is k-fold cross validation (k-fold C-V). It consists on three steps: first, dividing the dataset into k equal slices; second, for each slice ($i \in \{1, , , k\}$), learn the classifier on all but the i-th slice (the training dataset), and compute the performance index on the left out slice (the testing dataset); last, to obtain a single index, the k performance values are averaged together. C-V is also used for hyper-parameter tuning: finding the parameter of the learning algorithm which results in the best performance, by applying cross-validation to the model with different values of the parameter. The hyper-parameter tuning C-V can be used inside the C-V for model assessment in a 2 nested cross-validation. The latter technique is used to assess the 3 models previously cited.

4 Experimental evaluation

4.1 Data

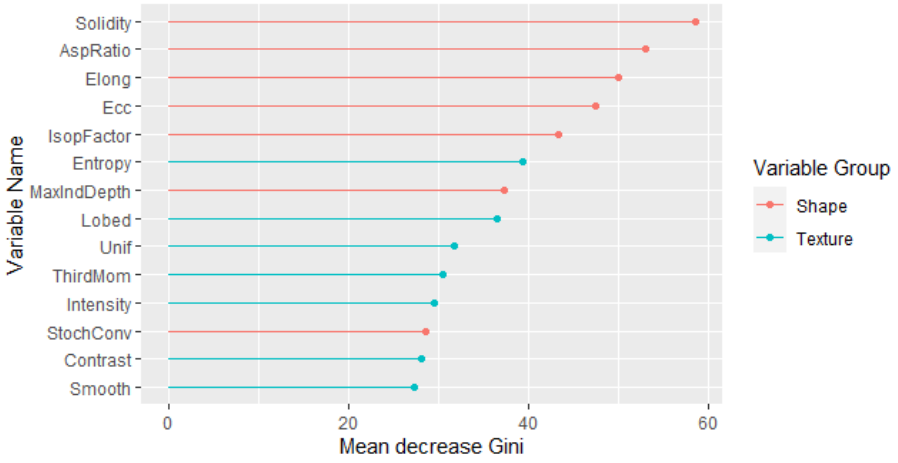
As stated before, the data is comprised of 340 leaves from 30 different species. Since the dataset is not large and the classes are 30, applying cross validation for the assessment of the model may end up in creating "testing" slices in which some species do not appear.

Therefore I decided to enlarge the dataset using the SMOTE algorithm. This method is usually employed in binary classification when the dataset is unbalanced, in order to increase the size of the minority class. SMOTE generates synthetic observations for the minority class based on the existing ones (more details on the algorithm [2]). To extend it to a multi-class setting, for each specie ($i \in Species = \{1, ..., 30\}$), I considered the i-th specie as "rare", and the rest as "common", and applied SMOTE so that each specie has 19 observations (570 leaves in total).

4.2 Procedure

Before assessing the classifier, I applied the Random forest algorithm to the dataset to point out which feature is more relevant (**Figure 1**).

Figure 1: Variable importance



To assess the Random forest model for this specific leaf identification problem, I used the 2 nested cross-validations technique previously described, with a value of $k = 5$ for both. The inner most is used to find the best value for the parameter m, and the values tested are the integers from 1 to 14 included; the best model is chosen to be the one with the highest accuracy. This model is then applied on left-out fold of the outer-most C-V for prediction, and a confusion

matrix is obtained; the latter is then used to compute the perfromance indexes described in Section 2. This process is repeated over the 5-fold of the outer C-V, and the 5 values of the performance measures obtained are then averaged together.

I then applied the same assessment technique to the classification tree model, tuning the complexity parameter *cp* between the values from 0.005 to 0.03 by a step of 0.001, and the gradient boosting model, over the default values for the parameters given by the *caret* package in R (`https://topepo.github.io/caret/`).

Finally, to check that the difference in accuracy measured by the 3 models is not due to randomness, I repeated the assessment procedure 10 times for each algorithm changing the seed correspondingly, and applied a paired student-t test to these samples (just in the case of Random forest and boosting).

4.3 Results and discussion

The performance measures of Random forest resulting from the 2 nested C-V technique applied to the leaf dataset are reported in the following table (seed vale of 264).

	Accuracy	Macro precision	Macro recall	Macro F1
Mean	0.90	0.92	0.91	0.90
Std dev	0.02	0.03	0.02	0.02

These results suggest that the random forest model not only has a good overall performance (high accuracy), but that is able to classify correctly each independent class (high F1).

In the following table are reported the 10 samples for accuracy obtained by applying the assessment technique changing the seed.

seed	12	162	264	345	42	57	69	71	888	941
RF	0.88	0.89	0.90	0.88	0.88	0.89	0.90	0.90	0.91	0.90
Tree	0.71	0.74	0.73	0.72	0.71	0.74	0.73	0.74	0.72	0.71
XGBoost	0.85	0.86	0.87	0.85	0.86	0.83	0.90	0.91	0.84	0.87

The accuracy of the tree is clearly lower than the one of Random forest, so no statistical test is required. The means and standard deviation of the samples of RF and boosting are respectively 0.89 ± 0.01 and 0.87 ± 0.03, so a statistical test is needed. By applying the student-t test with a significance value of $\alpha = 0.05$ to the samples of RF and boosting, a p-value of 0.003 is obtained, advocating to reject the null hypotheses according to which those random variables follow the same distribution. This confirms in fact that the distributions are distinct, and that the difference is not due to randomness. Therefore, Random forest can be considered the best classifier.

References

[1] Manuel Fernàndez-Delgado et al. Do we need hundreds of classifiers to solve real world classification problems. *J.Mach. Learn. Res.*, 15.1:3133–3181, 2014.

[2] Nitesh V. Chawla et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[3] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume Second edition. Springer, 2009.