Vicari IML final project

Matteo Vicari

2023-01-13

Problem statement

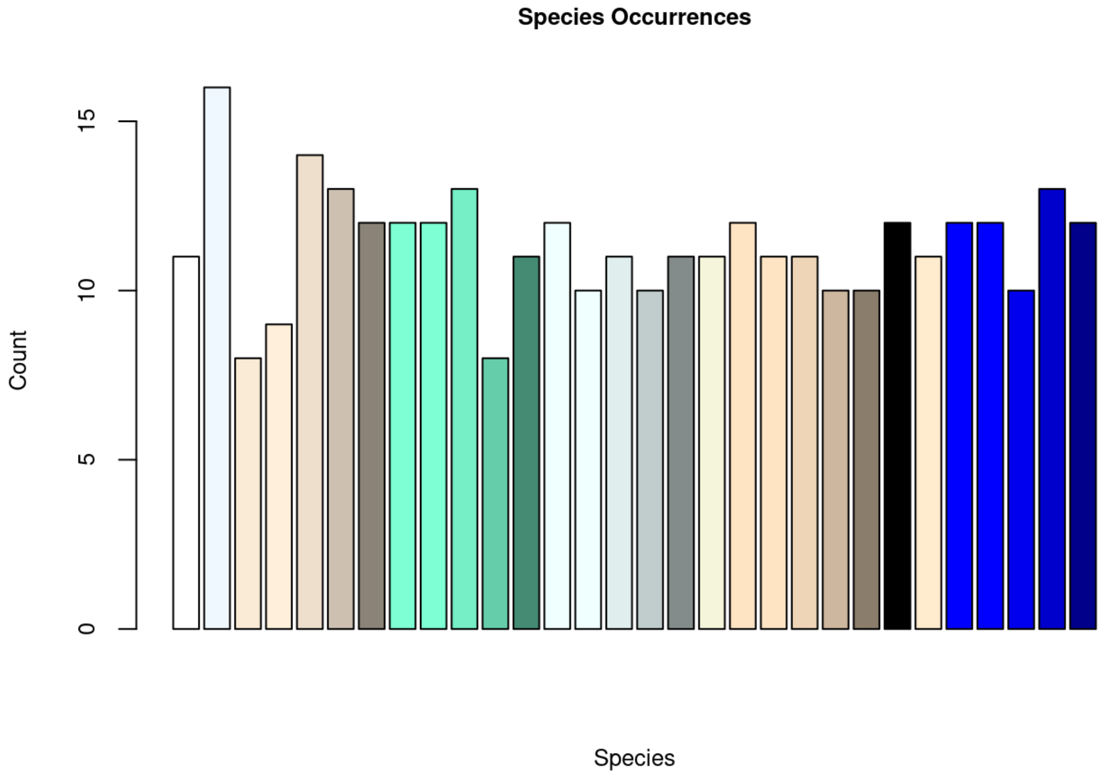
Starting from a dataset of leaf attributes and measurements from different species, a method is proposed for recognizing leaf species using one or more machine learning techniques. Different methods will be tested and the one considered best for predicting the species of a leaf based on the collected data will be proposed.

Analysis of the dataset and data description

The dataset is composed by **340 observations** and **30 different plant species** (40 Species are available only if we consider also images but that is out of scope of the course) and each observation has in the first column the class label, the number of leaf specimens and the other columns contain **shape attributes (columns from 3 to 9)** and texture attributes (columns from 10 to 16).From the input data for the problem, there are papers that already address the issue using an analysis methodology with images and techniques that go beyond the course content [1] [2]. Here is an example of an observation of the dataset restricted to shape attributes.

Species	Eccentricity	Aspect Ratio	Elongation	Solidity	Stochastic Convexity	Isoperimetric Factor	Maximal Indentation Depth	Lobedness	Average Intensity
Quercus suber	0.72694	1.4742	0.32396	0.98535	1	0.83592	0.0046566	0.0039465	0.04779

In the barplot below we can see that the dataset is unbalanced. While building the method for classifying the species, several tests have been made with both the original dataset and the dataset balanced in different way. So different models have been generated using both downsampling and upsampling and mantaining the original dataset. Moreover the dataset it's ordered by species type. Dataset shuffling will be needed to avoid errors while learning models and to avoid the risk of obtaining a biased model.



We are using R as a programming language to analyze the data and build the model and the code was made reproducible using a specific seed function. Data are consistent and there are no missing values. The second column of the dataset is not needed since it refers to the specimen number. Since the proposed dataset also has a response column, it makes sense to use a supervised learning method. We can proceed to build a model.

Knn

An attempt to create a model was made using the k-nearest neighbors learning method. In this case, the dataset was shuffled, scaled, and various learning function tests were performed, with percentages of division between the learning and test datasets of **0.6, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90**, and for each of them, a learning function was created **with k values** (number of nearest neighbors to consider when making predictions) **from 1 to 50** (it should be noted that the computational cost of the operation is so low that extreme values of the k parameter can be tested). Seems like the best weighted accuracy was obtained with a division of the **dataset of 0.85 for the learning and with values of k 4,5** but in any case the **weighted accuracy remains low around 0.65**.

Svm

We then attempted to create a model using the SVM machine learning technique. After scaling and shuffling the data ^{3} dedicated to the model's learning, we obtained a function that allows us to predict data with a **weighted accuracy variable mostly between 0.55 and 0.75**. In this case as well, since the computational effort is not significant, we performed several tests to find the model. Using different types of learning (**linear, polynomial, radial**) and with a **tolerance c parameter that vary in different tests from 1 to 100** and in case of the radial learning values of **gamma in a range from 0 to 10 with a step of 0.2**. Seems like the best result was obtained with the radial model and with a split value of the dataset of 0.85 between learning data and test data. Unfortunately, in any case, given equal conditions (tolerance of the margin and percentage of the dataset dedicated to learning/testing and gamma values), applying the test dataset to evaluate the weighted accuracy, results vary a lot suggesting that it depends on which observations are sampled for the learning dataset.

Random forest

An other attempt has been done to build a model for learning data using the Random Forest machine learning technique. Since the cost of repeating the learning function several times is irrelevant (just a few seconds of computation), we build a lot of different learning functions using as starting data the original dataset, downsampled dataset and upsampled dataset (so for example we set a limit of 5 obs. per species for the learning dataset and 3 obs. for testing the model). And this has been done for every possible value of the learning dataset to a **maximum of 14 observation per species** and for every possible value of the remaining observations used for testing dataset. Upsampling has been limited to 2 observations created for species that only have 8 observations so that the total number of observations for these species became 10. The number of trees used for the random forest has been tested with different values but seems like **500 trees** is quite enough since higher values does not improve accuracy. Also omitted variables while learning are not constant and different values has been tested. Unfortunately also this time OOB values results in mostly values of **error between 0.35 and 0.4** and **weighted accuracy returns, at most, around 0.70**. Only few test barely exceed 0.7.

Results and comments

Given the poor results achieved with previous attempts, a search has been done with a focus on the species that mostly decrease the model accuracy. Using the random forest again, but excluding n different species at a time, a model was constructed and the average OOB error was evaluated. **To have an average error lower than 0.2, it was necessary to remove at least 10 species**. To have an average error lower than 0.3, it was necessary to remove at least 3 species. As a result, an acceptable model could be one constituted by a prediction function that is the result of a random forest learning function that excludes at least three species (from the experimental data, it seems that the species Hydrangea sp., Magnolia soulangeana, Acca sellowiana, Ilex perado ssp. azorica increase the error a lot). By removing 3 or more species, we could obtain a functional predictive model. Further research should be requested for those species, possibly by obtaining more data and different attributes and building a separate model. Furthermore, the probabilities could be provided as a result of the predictive functions.

References

1. "Evaluation of Features for Leaf Discrimination", Pedro F. B. Silva, Andre R.S. Marcal, Rubim M. Almeida da Silva (2013), Springer Lecture Notes in Computer Science, Vol. 7950, 197-204.

2. “Development of a System for Automatic Plant Species Recognition”, Pedro Filipe Silva, Dissertação de Mestrado (Master’s Thesis), Faculdade de Ciências da Universidade do Porto. Available for download or online reading at http://hdl.handle.net/10216/ (http://hdl.handle.net/10216/) 67734

3. C.-W. Hsu, C.-C. Chang, C.-J. Lin. A practical guide to support vector classification . Technical report, Department of Computer Science, National Taiwan University. July, 2003.