Plant Species Classification Using Extreme Learning Machine

Larissa F. S. Britto[1], **Luciano D. S. Pacifico**[1]

[1]Departamento de Computação – DC
Universidade Federal Rural de Pernambuco – UFRPE
Recife, Pernambuco, Brasil

{larissa.feliciana, luciano.pacifico}@ufrpe.br

Abstract. *Plants play an important role in nature, but correct plant species identification is still a challenging task for non-specialized people. Many works have been proposed towards the development of automatic plant species recognition systems through Machine Learning methods, but most of them lack the proper experimental analysis. In this work, we evaluate the performance of a general-purpose Artificial Neural Network to perform plant classification task: the Extreme Learning Machine (ELM). We compare ELM with several classifiers from plant recognition literature by means of three real-world data sets obtained from different image processing and feature extraction processes. A statistical hypothesis test is employed to perform proper experimental evaluation.*

1. Introduction

Plants exist everywhere on the Earth, performing many important functions to their environments. Plants are the major source of oxygen (through photosynthesis process) and food on the planet, since no animal is able to supply the components necessary to its survival without plants. Plants also provide shelter, clothing material, coal, medicines, paper products, reduce noise levels and wind speed, reduce water runoff and soil erosion [Rahmani et al. 2015].

Although people see many kinds of plants in their daily lives, correct plant identification is a difficult problem. Gathering information about unknown plants manually, such as reading books or researching on Internet, is a time consuming and tedious task, and users not familiar with botanical knowledge can make mistakes. In an attempt to overcome such limitations, automatic plant identification has become an active area of research in both botany and computer community. As machine learning, image processing and hardware technologies advance, sophisticated systems have been proposed to deal with this task, but despite many efforts (like in [Agarwal et al. 2006, Kumar et al. 2012]), plant identification is still considered a challenging and unsolved problem.

Many works of automatic plant recognition literature focus on feature extraction techniques [Charytanowicz et al. 2010, Kumar et al. 2012, Mallah et al. 2013, Jin et al. 2015, Sahay and Chen 2016], leaving the choice of the classier as a secondary task, what may compromise the accuracy obtained by such systems. In general, these works lack the proper evaluation criteria, and yet, such works may not even provide any form of statistical analysis to validate the obtained results. Also, it is common to find works that do not furnish any kind of comparison among the proposed classifier and other classifiers from literature (they only present the obtained results for the models being proposed), like in [Kumar et al. 2012, Rankothge et al. 2013, Liu et al. 2016].

Artificial Neural Networks (ANNs) [Haykin 2001] are deeply connected bio-inspired computational models known to be universal approximators, presenting good performances even when dealing with hard problems. Some characteristics of ANNs are adaptability, tolerance to failures, capacity of learning by examples and the ability to organize or to generalize data. ANNs have been applied as classifiers in many fields, such as engineering, bioinformatics, medicine, time series forecasting, and so on. The main limitation concerning the application of ANNs in real-world systems is the fact that many traditional algorithms for ANNs training are gradient-based strategies, demanding high computational costs to train. In this context, Extreme Learning Machine (ELM) [Huang et al. 2006] was proposed as a fast algorithm for Single-Hidden Layer Feedforward Neural Networks (SLFNs) training. ELM increases the learning speed by means of randomly generating input weights and hidden biases, differently from gradient-based approaches, which commonly tune iteratively the network parameters. ELM has been successfully applied to a large variety of real-world problems in the past years [Rahma et al. 2017, Pacifico et al. 2018a, Shen et al. 2018, Sönmez et al. 2018, Song et al. 2019, Vijendran and Dubey 2019, Xu et al. 2019].

In this work, we evaluate the performance of Extreme Learning Machine neural network to deal with plant classification task. Our main objectives are:

1. To present a brief review on the state-of-the-art of plant classification;
2. To show the main characteristics of Extreme Learning Machines;
3. To execute experiments on plant data sets obtained from different feature extraction processes;
4. To compare the performance of ELM in relation to some standard classifiers in plant classification literature;
5. To evaluate the experimental results through the application of a statistical hypothesis test.

This work is organized as follows. Section 2 presents a brief review on plant identification systems. Next, ELM model is presented (Section 3). Section 4 describes the adopted data sets. The experimental results are discussed in Section 5. Finally, the main conclusions and trends for future works are given in Section 6.

2. A Brief Review on Automatic Plant Identification Systems

The main steps in an automatic plant recognition system are: plant image acquisition, image pre-processing, feature extraction and plant classification [Sahay and Chen 2016].

Plant image acquisition step is generally employed under controlled environmental situations (such as controlled background and illumination sources), to avoid image acquisition problems like object partial occlusion, folding, illuminant changes, and so on. For example, in [Mallah et al. 2013], the proposed data set is composed of color plant images captured when each leaf sample is placed on a white surface, providing high contrast between the target objects (the plant leaf samples) and the background. The same approach was adopted by [Sahay and Chen 2016] and [Kumar et al. 2012]. But in some applications [Cerutti et al. 2013, Sun et al. 2017], plant images are captured from natural environments.

After image acquisition, image pre-processing starts. The main objectives of image pre-processing is to standardize the scale and orientation of the image before fe-

ature extraction step. Plant image pre-processing is generally employed through gray scale or binary conversion, noise removal, contrast stretching and histogram modification [Sabu and Sreekumar 2017].

Features generally considered for automatic plant recognition are: color, leaf features, flower features, seed features and other organs features [Cope et al. 2012, Rahmani et al. 2015, Sabu and Sreekumar 2017]. Many plant classification systems focus on the study of plant leaf features, such as shape, margin, texture, venation, teeth and fractal dimensions [Cope et al. 2012, Jin et al. 2015, Sabu and Sreekumar 2017]. Plant taxonomy suggests that a species can be successfully inferred from the leaves, since the leaves are the more readily available, easily found and collected than other parts of the plant [Mallah et al. 2013]. Although most plant identification systems consider only plant organ images and their combination, some authors adopt external information (i.e., information that are not contained in the plant images) to help the recognition process. For example, in [Liu et al. 2016] proposes to automatically identify plant species using images from different plant organs, such as flowers and leaves, and also by taking into consideration the location information (represented by GPS coordinates) about the region where the plant specimens have been acquired.

The final step in automatic plant recognition systems is the classification phase [Britto and Pacifico 2018, Pacifico et al. 2018b]. Some methods commonly employed as classifiers for plant identification systems are: K-Nearest Neighbors classifier [Cover and Hart 1967], Decision Tree classifier [Mitchell et al. 1997], Naive Bayes classifier [Mitchell et al. 1997, De Stefano et al. 2012], Support Vector Machine [Haykin 2001] and Artificial Neural Networks [Haykin 2001]. ELM has also been adopted in the context of plant classification [Zhai and Du 2008], but its performance has only been compared to an Artificial Neural Network, with a limited set of experiments.

In [Mallah et al. 2013] and [Mallah and Orwell 2013], a K-Nearest Neighbors density estimation method is developed for plant leaf classification based on integration of shape, texture and margin features. A weighted K-Nearest Neighbors is also adopted as the classifier in [Sahay and Chen 2016] and compared to standard K-Nearest Neighbors algorithm. In [Rahmani et al. 2015] present an evaluation on the performance of Decision Tree classifier, Naive Bayes classifier and K-Nearest Neighbors algorithm for different plant leaves classification scenarios. The Leafsnap system [Kumar et al. 2012] is based on Support Vector Machine classifier. Support Vector Machine is also employed by [Prasad et al. 2011], where Relative Sub-image Coefficients (RSC) are extracted from plant leaves and used as features for the proposed plant leaf identification system. An ANN trained with Backpropagation algorithm is used in [Rankothge et al. 2013]. Other classifiers are also employed, as in [Jin et al. 2015], where a sparse representation based classifier and four leaf tooth characteristics are employed to perform automatic plant identification.

Some interesting reviews on plant recognition are presented in Cope et al. [Cope et al. 2012], Sethulekshmi and Sreekumar [Sethulekshmi and Sreekumar 2014] and Sabu and Sreekumar [Sabu and Sreekumar 2017].

3. Extreme Learning Machine

The Extreme Learning Machine [Huang et al. 2006] algorithm was presented as a fast algorithm for single-hidden layer feed-forward neural networks training. A SLFN has three layers of neurons, but the name *single* comes from the hidden layer (only layer of non-linear neurons in the model). Input layer provides data features and performs no computations, while an output layer is linear without a transformation function and without biases [Akusok et al. 2015].

The random generation of input weights and hidden biases and the straightforward analytical determination of output weight matrix makes ELM too much faster than traditional gradient-based learning methods like Backpropagation algorithm. Also, there are no parameters to tune during ELM training, once ELM provides a non-iterative linear solution for the output weights, which is possible because there is no dependence between the input and output weights like in the Backpropagation [Akusok et al. 2015].

Formally, suppose we are training SLFNs with N hidden neurons and activation function $f(x)$ to learn M distinct samples $(\mathbf{x}_i, \mathbf{t}_i)$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{ik}]^T \in \Re^k$ and $\mathbf{t}_i = [t_{i1}, t_{i2}, \ldots, t_{id}]^T \in \Re^d$. By doing so, the nonlinear system has been converted to a linear system:

$$\mathbf{H}\,\beta = \mathbf{T} \tag{1}$$

where $\mathbf{H}$ is the hidden-layer output matrix denoted by:

$$\mathbf{H} = \begin{bmatrix} f(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & f(\mathbf{w}_N \cdot \mathbf{x}_1 + b_N) \\ \vdots & \ddots & \vdots \\ f(\mathbf{w}_1 \cdot \mathbf{x}_M + b_1) & \cdots & f(\mathbf{w}_N \cdot \mathbf{x}_M + b_N) \end{bmatrix}$$

where $\mathbf{w}_j = [w_{j1}, w_{j2}, \ldots, w_{jk}]^T (j = 1, \ldots, N)$ is the weight vector connecting j-th hidden neuron and input neurons, and b_j denotes the bias of j-th hidden neuron; $\mathbf{w}_j \cdot \mathbf{x}_i (i = 1, \ldots, M)$ denotes the inner product of $\mathbf{w}_j$ and $\mathbf{x}_i$; $\beta = [\beta_1, \beta_2, \ldots, \beta_N]^T$ is the matrix of output weights and $\beta_j = [\beta_{j1}, \beta_{j2}, \ldots, \beta_{jd}]^T (j = 1, \ldots, N)$ denotes the weight vector connecting the j-th hidden neuron and output neurons; $\mathbf{T} = [\widehat{\mathbf{t}}_1, \widehat{\mathbf{t}}_2, \ldots, \widehat{\mathbf{t}}_M]^T$ is the matrix of targets (desired output). In the case where the SLFN perfectly approximates the data, the errors between the estimated outputs $\widehat{\mathbf{t}}_i$ and the actual outputs $\mathbf{t}_i$ are zero and the relation is:

$$\mathbf{t}_i = \sum_{j=1}^{N} \beta_j f(\mathbf{w}_j \cdot \mathbf{x}_i + b_j) \tag{2}$$

Thus, the determination of the output weights (linking the hidden layer to the output layer) is determined by the least-square solution to the linear system represented by Equation 3. The minimum norm least-square solution to the linear system is:

$$\widehat{\beta} = \mathbf{H}^{\dagger}\mathbf{T} \tag{3}$$

where $\mathbf{H}^{\dagger}$ is the Moore-Penrose generalized inverse of matrix $\mathbf{H}$. The minimum norm least-square solution is unique and has the smallest norm among all the least-squares solutions. As analyzed by Huang et al. [Huang et al. 2006], ELM using such Moore-Penrose inverse [Serre 2002] method tends to obtain good generalization performance with dramatically increased learning speed.

A pseudocode for the ELM algorithm is presented in Algorithm 1.

Algorithm 1 Extreme Learning Machine

Randomly initialize all input weights and hidden biases.
Calculate the hidden-layer output matrix **H**.
Estimate $\mathbf{H}^{\dagger}$ as the Moore-Penrose generalized inverse obtained from **H**.
Calculate the output weights matrix $\widehat{\beta}$

4. Data sets

In this section, the adopted data sets are presented. All data sets are real-world problems obtained from UCI Machine Learning Repository [Asuncion and Newman 2007].

Fisher's Iris Plant [Fisher 1936] is composed of 150 samples equally distributed in three classes, where each class refers to a type of iris plant. Each instance is described by a set of four features: sepal length (in cm), sepal width (in cm), petal length (in cm) and petal width (in cm). The sample values for each feature have been manually collected at the same day and using the same instruments by Edgar Anderson [Anderson 1935].

The Wheat Seed Kernels data set [Charytanowicz et al. 2010] comprises 210 randomly selected samples equally distributed in three varieties of wheat (Kama, Rosa and Canadian). The features have been extracted using a soft X-ray technique to detect the visualization of the internal kernel structure for each sample seed.

The 100 Plant Leaves data set [Mallah et al. 2013] comprises one-hundred species of leaves (problem classes). The data set contains 1600 instances, and for each species, there are sixteen distinct specimens, photographed as a color image on a white background. For each sample, three distinct features have been extracted: a Centroid Contour Curve shape signature (Sha), an interior texture feature histogram (Tex), and a fine-scale margin feature histogram (Mar). Each feature is represented by a 64-dimensional vector.

5. Experimental Results

In this section, the experimental results are presented. We compare the ELM method to five different classifiers from plant classification literature and some of their variants: Decision Tree classifier (DT), Naive Bayes classifier (NB), K-Nearest Neighbors (KNN, with $k = 3, 4$ and 5), Support Vector Machine with RBF (SVM$_{rbf}$) and $Linear$ (SVM$_{lin}$) kernel functions and a Multi-Layer Perceptron trained with Backpropagation algorithm (MLP).

All algorithms have been implemented in Python programming language, and all tests have been executed in a computer with an i5-5250U CPU and 8 GB RAM. DTC, NBC, KNN, SVM and MLP have been implemented using scikit-learn library [Pedregosa et al. 2011, Buitinck et al. 2013] and ELM has been implemented using HP-ELM library [Akusok et al. 2015]. No GPU acceleration has been employed for ELM, so we could perform a fair comparison among all adopted algorithms in relation to average execution times. All algorithms have been executed using scikit-learn default configurations, except for SVM, where two different kernel functions have been applied. For ELM, the best number of hidden nodes for each data set has been selected by trial-and-error method.

Our experiments have been conducted using a ten-folds cross-validation framework. Each data set has been randomly split into ten parts to form the training and

Tabela 1. Benchmark Plant Data Sets Description. Attr.: number of features; Classes: number of classes; Total: total number of data patterns.

Data set	Attr.	Classes	Total
Iris	4	3	150
Seeds	7	3	210
Margin (M)	64	100	1600
Shape (S)	64	100	1600
Texture (T)	64	100	1600
Margin and Shape (MS)	128	100	1600
Margin and Texture (MT)	128	100	1600
Shape and Texture (ST)	128	100	1600
Margin, Shape and Texture (MST)	192	100	1600

testing sets. Nine folds are used each time to compose the training set, and the remaining fold is used as the testing set. To generate a large variety of tests, the ten-folds cross-validation process has been executed ten times, and, for each execution, ten new distributions have been obtained for each data set, in such a way that we could have one hundred different tests for each data set (the ten-folds cross-validation method has been executed ten times, each time starting with a new random distribution of the data patterns into the folds). The adopted resampling process has been performed to avoid results obtained by chance.

For the experiments, three benchmark plant data sets obtained from UCI Machine Learning Repository are employed: Iris, Seeds and 100 Plant Leaves (see Section 4). 100 Plant Leaves is decomposed in seven data sets, so we could evaluate the influence of each plant leaf feature (margin, shape and texture) and their combinations on the behavior of the adopted classifiers: Margin, Shape, Texture, Margin and Shape, Margin and Texture, Shape and Texture, and Margin, Shape and Texture [Rahmani et al. 2015]. The selected data sets are summarized in Table 1.

The evaluation includes an empirical analysis concerning the average accuracy for the testing set and the execution time for each data set. The evaluation also includes a rank system employed through the application of Friedman test [Friedman 1937] to the overall average testing accuracies. The Friedman test is a non-parametric hypothesis test that ranks all algorithms for each data set separately. If the null-hypothesis (all ranks are not significantly different) is rejected, Nemenyi test [Nemenyi 1962] is adopted as the *post-hoc* test. According to Nemenyi test, the performance of two algorithms are considered significantly different if the corresponding average ranks differ by at least the critical difference

$$CD = q_a \sqrt{\frac{n_{alg}(n_{alg}+1)}{6n_{data}}} \tag{4}$$

where n_{data} represents the number of data sets, n_{alg} represents the number of compared algorithms and q_a are critical values based on a Studentized range statistic divided by $\sqrt{2}$ [Demšar 2006]. Given that the overall average test accuracy is a maximization measure, the best ranked algorithms for the Friedman/Nemenyi test will obtain high ranks. Once our experiments are executed with $n_{data} = 9$ and $n_{alg} = 9$, we have $CD = 4.0043$.

Tabela 2. Experimental results for all plant classification data sets. Mean: average accuracy for the test set; Std: standard deviation for the test set; Time: average execution time in seconds.

Data set	Metric	Algorithm								
		DT	NB	KNN_3	KNN_4	KNN_5	SVM_{lin}	SVM_{rbf}	MLP	ELM
Iris	Mean	0.949	0.955	0.954	0.961	0.959	0.843	0.895	0.963	**0.974**
	Std.	0.048	0.048	0.051	0.049	0.050	0.086	0.076	0.044	**0.039**
	Time	0.001	0.001	0.001	0.001	0.001	0.005	0.007	0.602	0.005
Seeds	Mean	0.914	0.902	0.929	**0.951**	0.930	0.912	0.915	0.924	**0.961**
	Std.	0.064	0.067	0.060	**0.044**	0.061	0.066	0.067	0.059	**0.039**
	Time	0.001	0.001	0.001	0.001	0.001	0.007	0.006	0.236	0.003
M	Mean	0.460	0.748	0.746	0.746	0.756	0.718	0.711	**0.827**	0.806
	Std.	0.039	0.035	0.035	0.035	0.033	0.035	0.033	**0.026**	0.031
	Time	0.081	0.019	0.197	0.211	0.196	0.831	0.949	3.955	0.565
S	Mean	0.440	0.532	**0.591**	**0.588**	0.579	0.409	0.477	0.516	0.576
	Std.	0.041	0.034	**0.038**	**0.038**	0.035	0.036	0.038	0.035	0.037
	Time	0.268	0.018	0.2054	0.214	0.2010	2.729	1.121	3.913	0.443
T	Mean	0.516	0.668	0.766	0.764	0.761	0.736	0.724	0.822	**0.847**
	Std.	0.040	0.033	0.033	0.034	0.033	0.034	0.036	0.033	**0.032**
	Time	0.125	0.019	0.203	0.212	0.199	0.809	0.929	3.866	0.755
MS	Mean	0.679	0.851	0.937	0.931	0.934	0.916	0.880	**0.954**	**0.951**
	Std.	0.043	0.026	0.019	0.020	0.018	0.021	0.027	**0.015**	**0.016**
	Time	0.395	0.025	0.619	0.614	0.617	1.388	1.554	4.645	1.982
MT	Mean	0.607	0.741	0.967	0.964	0.961	**0.975**	0.938	**0.977**	**0.976**
	Std.	0.035	0.031	0.015	0.014	0.015	**0.010**	0.018	**0.012**	**0.011**
	Time	0.216	0.025	0.605	0.609	0.610	1.1703	1.463	4.576	1.099
ST	Mean	0.666	0.801	0.922	0.918	0.916	0.905	0.874	**0.941**	**0.945**
	Std.	0.036	0.031	0.019	0.020	0.021	0.024	0.025	**0.018**	**0.015**
	Time	0.452	0.025	0.606	0.608	0.607	1.379	1.551	4.6011	0.757
MST	Mean	0.720	0.756	0.987	0.987	0.985	**0.993**	0.973	**0.992**	**0.993**
	Std.	0.038	0.032	0.009	0.009	0.010	**0.007**	0.013	**0.008**	**0.007**
	Time	0.570	0.032	0.906	0.906	0.906	1.6983	2.100	5.377	2.010

The experimental results are shown in Table 2.

From Table 2, in an empirical analysis, we can see that ELM is able to obtain performances at least as good as the best classifiers from literature for most of the evaluated data sets. ELM achieved the best performances according to the empirical analysis for seven out of nine data sets. Both ANNs (ELM and MLP) showed the higher degrees of robustness among the employed algorithms. ELM, MLP, KNN_3 and KNN_4 presented the best stability among all algorithms.

For the 100 Plant Leaves data set, we can observe that some features taken individually present a reduced discriminatory power, but when the extracted features are combined, the discriminatory power is enhanced. The performance of all adopted classifiers have been considerably compromised when only the shape feature vector is used. That is completely understandable, since the shape of a leaf may vary considerably according to some problems, like deformations caused by disease, insects, human or mechanical damage [Jamil et al. 2015], compromising its discriminatory power. Another point to be considered is the fact that some plant species may have very similar leaf shapes to each other, like some plant species belonging in the same family. For all three evaluated leaf features, texture has presented the best individual discriminatory power according to the

Tabela 3. Overall Evaluation: Average Ranks for the Friedman/Nemenyi Test.

Algorithm	Average Rank
DT	152.0972
NB	285.2706
KNN_3	543.5028
KNN_4	537.9994
KNN_5	524.6883
SVM_{lin}	395.3672
SVM_{rbf}	310.4961
MLP	615.5828
ELM	**689.4956**

experimental results.

All classifiers presented better average accuracies when the features are combined two-by-two. The best combination of two features for most of the evaluated algorithms have been obtained when margin and texture features have been combined (except for DT algorithm, where the best average accuracy have been obtained by the combination of margin and shape features). For example, ELM was able to achieve an average accuracy of 80.64% when considering leaf margin feature and 84.67% and considering leaf texture feature, but when leaf margin and leaf texture have been combined, the algorithm was able to achieve an average accuracy of 97.59%.

For most of the evaluated algorithms, the best results for 100 Plant Leaves data set have been achieved when all three features have been combined (for example, ELM obtained an average test accuracy of 99.31% in this situation). The only exception is NB, where the best average accuracy have been obtained when only leaf margin and leaf texture have been combined. The obtained results for 100 Plant Leaves problem showed that it is quite important to find the best set of features when dealing with automatic plant recognition, since some features present better discriminatory power than others. But, as pointed out by the average execution times, the computational cost for the classifiers may increase considerably when dealing with problems with higher dimensionalities (*the curse of dimensionality* problem [Bellman 1957]).

Table 3 presents the average ranks obtained by Friedman/Nemenyi hypothesis test. The Friedman/Nemenyi test shows that ELM obtained the best average performances according to an overall evaluation, in comparison to all other selected algorithms. The second and the third best ranks have been obtained by MLP and KNN_3, respectively. The worst overall performance have been achieved by Decision Tree classifier, followed by Naive Bayes classifier and Support Vector Machine. Figure 1 presents the obtained results for the Friedman/Nemenyi test, from the worst method (on the left side) to the best method (on the right side).

From the experiments, we can see that ANNs are good options to deal with plant classification problem. Although MLP has presented high computational costs in relation to all other classifiers, what may limit its application in real-world systems, ELM showed to be fast, presenting average execution times compatible with some other classifiers from the state-of-the-art (like Support Vector Machines), and yet presenting the capability to achieve high performances, what makes ELM algorithm a good option as a tool for the

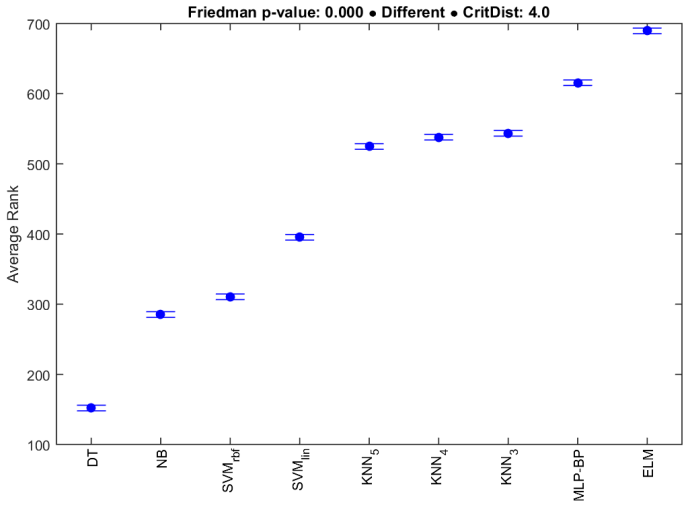

Figura 1. Overall Evaluation: Average ranks for the Friedman/Nemenyi test.

deployment of real-world plant recognition applications.

6. Conclusion

In this work, we evaluate the performance of Extreme Learning Machine algorithm in the context of plant classification. Five well-established supervised learning methods from plant classification literature are compared to ELM: K-Nearest Neighbor, Decision Tree classifier, Naive Bayes classifier, Support Vector Machine and a Multi-Layer Perceptron trained with Backpropagation algorithm. Different configurations for K-Nearest Neighbors algorithm (different k values) and for Support Vector Machines (two different kernel functions - $Linear$ and RBF functions) are employed.

For comparison purposes, three real-world plant data sets obtained from UCI Machine Learning repository are employed: Iris, Wheat Seeds and 100 Plant Leaves. 100 Plant Seeds data set have been divided in seven data sets, so we could test each individual plant leaf feature and all possible combinations of the three features (leaf margin, leaf shape and leaf texture).

The evaluation criterion is based on an empirical analysis complemented by a hypothesis test of type Friedman/Nemenyi test in relation to the average test accuracy obtained by each classifiers for each of the nine adopted data sets.

The experimental results pointed out that Extreme Learning Machine is able to achieve better performances then all comparison approaches according to Friedman/Nemenyi hypothesis test, and it is at least as good as the best methods for most of the tested data sets (seven out of nine). The experimental results also showed that ELM is able to achieve good generalization performances independently from the feature extraction process employed by the data sets, avoiding data set bias problem.

As future works, we intend to extend our study by the application of image processing techniques to extract other plant features (such as leaf venation and leaf tooth features) automatically, evaluating the influence of each individual new feature and their combination on the behavior of the selected classifiers. With a higher set of plant features, we intend to implement an automatic method for feature selection, aiming to reduce

the dimensionality and the computational costs for the final plant recognition system by eliminating redundant features and features with low discriminatory power. We also intend to develop a general purpose application for automatic plant identification as tool for botanists and researchers on the field of plant taxonomy.

Referências

Agarwal, G., Belhumeur, P., Feiner, S., Jacobs, D., Kress, W. J., Ramamoorthi, R., Bourg, N. A., Dixit, N., Ling, H., Mahajan, D., et al. (2006). First steps toward an electronic field guide for plants. *Taxon*, 55(3):597–610.

Akusok, A., Björk, K.-M., Miche, Y., and Lendasse, A. (2015). High-performance extreme learning machines: a complete toolbox for big data applications. *IEEE Access*, 3:1011–1025.

Anderson, E. (1935). The irises of the gaspe peninsula. *Bulletin of American Iris Society*, 59:2–5.

Asuncion, A. and Newman, D. (2007). Uci machine learning repository.

Bellman, R. E. (1957). *Dynamic programming*. Princeton University Press.

Britto, L. F. and Pacifico, L. D. (2018). Plant classification using weighted k-nn variants. In *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*, pages 58–69. SBC.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Cerutti, G., Tougne, L., Mille, J., Vacavant, A., and Coquin, D. (2013). Understanding leaves in natural images–a model-based approach for tree species identification. *Computer Vision and Image Understanding*, 117(10):1482–1501.

Charytanowicz, M., Niewczas, J., Kulczycki, P., Kowalski, P. A., Łukasik, S., and Żak, S. (2010). Complete gradient clustering algorithm for features analysis of x-ray images. In *Information technologies in biomedicine*, pages 15–24. Springer.

Cope, J. S., Corney, D., Clark, J. Y., Remagnino, P., and Wilkin, P. (2012). Plant species identification using digital morphometrics: A review. *Expert Systems with Applications*, 39(8):7562–7573.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27.

De Stefano, C., Fontanella, F., and Di Freca, A. S. (2012). A novel naive bayes voting strategy for combining classifiers. In *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, pages 467–472. IEEE.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of human genetics*, 7(2):179–188.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association*, 32(200):675–701.

Haykin, S. S. (2001). *Neural networks: a comprehensive foundation*. Tsinghua University Press.

Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, 70(1):489–501.

Jamil, N., Hussin, N. A. C., Nordin, S., and Awang, K. (2015). Automatic plant identification: Is shape the key feature? *Procedia Computer Science*, 76:436–442.

Jin, T., Hou, X., Li, P., and Zhou, F. (2015). A novel method of automatic plant species identification using sparse representation of leaf tooth features. *PloS one*, 10(10):e0139482.

Kumar, N., Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C., and Soares, J. V. (2012). Leafsnap: A computer vision system for automatic plant species identification. In *Computer Vision–ECCV 2012*, pages 502–516. Springer.

Liu, J.-C., Chiang, C.-Y., and Chen, S. (2016). Image-based plant recognition by fusion of multimodal information. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2016 10th International Conference on*, pages 5–11. IEEE.

Mallah, C., Cope, J., and Orwell, J. (2013). Plant leaf classification using probabilistic integration of shape, texture and margin features. *Signal Processing, Pattern Recognition and Applications*, 5(1).

Mallah, C. D. and Orwell, J. (2013). Probabilistic classification from a k-nearest-neighbour classifier. *Computational Research*, 1(1):1–9.

Mitchell, T. M. et al. (1997). Machine learning. wcb.

Nemenyi, P. (1962). Distribution-free multiple comparisons. *Biometrics*, 18(2):263.

Pacifico, L. D., Ludermir, T. B., and Oliveira, J. F. (2018a). Evolutionary elms with alternative treatments for the population out-bounded individuals. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*, pages 151–156. IEEE.

Pacifico, L. D., Macario, V., and Oliveira, J. F. (2018b). Plant classification using artificial neural networks. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Prasad, S., Kudiri, K. M., and Tripathi, R. (2011). Relative sub-image based features for leaf recognition using support vector machine. In *Proceedings of the 2011 International Conference on Communication, Computing & Security*, pages 343–346. ACM.

Rahma, O., Wikaya, S., Prawito, and Badri, C. (2017). Electroencephalogram analysis with extreme learning machine as a supporting tool for classifying acute ischemic

stroke severity. *2017 IEEE International Seminar on Sensors, Instrumentation, Measurement and Metrology (ISSIMM)*, 1:180–186.

Rahmani, M. E., Amine, A., and Hamou, M. R. (2015). Plant leaves classification. *ALLDATA 2015*, 82.

Rankothge, W., Dissanayake, D., Gunathilaka, U., Gunarathna, S., Mudalige, C., and Thilakumara, R. (2013). Plant recognition system based on neural networks. In *Advances in Technology and Engineering (ICATE), 2013 International Conference on*, pages 1–4. IEEE.

Sabu, A. and Sreekumar, K. (2017). Literature review of image features and classifiers used in leaf based plant recognition through image analysis approach. In *Inventive Communication and Computational Technologies (ICICCT), 2017 International Conference on*, pages 145–149. IEEE.

Sahay, A. and Chen, M. (2016). Leaf analysis for plant recognition. In *Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on*, pages 914–917. IEEE.

Serre, D. (2002). *Matrices: Theory and Applications.* Springer.

Sethulekshmi, A. and Sreekumar, K. (2014). Ayurvedic leaf recognition for plant classification. *Journal of Computer Science and Information Technologies*, 5(6):8061–8066.

Shen, C., Zhang, S.-F., Zhai, J.-H., Luo, D.-S., and Chen, J.-F. (2018). Imbalanced data classification based on extreme learning machine autoencoder. In *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*, volume 2, pages 399–404. IEEE.

Song, Y., He, B., Zhao, Y., Li, G., Sha, Q., Shen, Y., Yan, T., Nian, R., and Lendasse, A. (2019). Segmentation of sidescan sonar imagery using markov random fields and extreme learning machine. *IEEE Journal of Oceanic Engineering*, 44(2):502–513.

Sönmez, Y., Tuncer, T., Gökal, H., and Avcı, E. (2018). Phishing web sites features classification based on extreme learning machine. In *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, pages 1–5. IEEE.

Sun, Y., Liu, Y., Wang, G., and Zhang, H. (2017). Deep learning for plant identification in natural environment. *Computational intelligence and neuroscience*, 2017.

Vijendran, S. and Dubey, R. (2019). Deep online sequential extreme learning machines and its application in pneumonia detection. In *2019 8th International Conference on Industrial Technology and Management (ICITM)*, pages 311–316. IEEE.

Xu, X., Deng, J., Coutinho, E., Wu, C., Zhao, L., and Schuller, B. W. (2019). Connecting subspace learning and extreme learning machine in speech emotion recognition. *IEEE Transactions on Multimedia*, 21(3):795–808.

Zhai, C.-M. and Du, J.-X. (2008). Applying extreme learning machine to plant species identification. In *2008 International Conference on Information and Automation*, pages 879–884. IEEE.