Information technology of accounting the impact of data reduction methods on the results of classification of plant leaves

V Kharakhinov¹, S Sosinskaya¹, R Dorofeev¹, A Dorofeev¹ and R Bazhenov²

¹ Irkutsk National Research Technical University, 83 Lermontov str., Irkutsk, 664074, Russia
² Sholom-Aleichem Priamursky State University, 70a Shirokaya str., Birobidzhan, 679015, Russia

E-mail: r-i-bazhenov@yandex.ru

Abstract. The paper proposes an information technology for classifying a sample of leaves of various plant species described by a set of numerical characters based on the use of various types of neural networks and a technique for reducing network learning time and improving the quality of classification. Neural networks of the multilayer perceptron type and softmax layer are considered. The application of factor analysis and auto-encoder is proposed in order to reduce the dimension of the initial feature space, which allowed reducing the time for training networks. When using factor analysis, two popular criteria were used to determine the number of factors: the Kaiser criterion and the criterion for the fraction of reproducible dispersion. The quality of classifications and the time spent on the learning process of networks are displayed in tabular form. As a result of the study, conclusions were drawn about the effect of reducing signs on reducing the training time of different types of neural networks and on the quality of classification.

1. Introduction

Traditionally, the classification of plants is carried out directly by a specialist in this field - a taxonomist-biologist. However, this approach is gradually losing its relevance due to the development of the field of information technology, in particular, the field of data mining.

The use of data mining methods is relevant when using robotics in agriculture, since this will significantly increase production volumes. This is especially true in Russia, which is a major exporter of agricultural products.

Many well-known scientific papers are devoted to the classification of leaves of various plant species using data mining methods.

Cope et al. conducted a study in which one of the methods for extracting numerical characters from the original leaf images was developed, and the data obtained was reduced using the principal component method [1].

In [2], a similar technique was used to extract traits from leaf images, however, data reduction was not performed, but classification by the K-nearest neighbours’ method was carried out.

A classification based on the application of graph theory, namely, supervised locality projection analysis (SLPA), was demonstrated by Zhang et al. [3].
An article by Chaki et al. is devoted to the classification of leaves using two neural classifiers: a neuro-fuzzy controller and a multilayer perceptron; the traits were extracted using the Gabor filter [4]. Aakif and Khan classified using a multilayer perceptron [5].

Article [6] contains a detailed analysis of the results of classifications obtained by different authors when analysing leaves from various samples. The method of K-nearest neighbours’ was used as a classifier. A distinctive feature of this work is the construction of a histogram of flowers in leaf images.

From the point of view of classification, using the neural network approach, the work of Sekeroglu and Inan is interesting, in which the authors managed to avoid the stage of data pre-processing and used the original images as data supplied to the network input. The classification was carried out by a multilayer perceptron, including 2 hidden layers in its architecture [7].

One of the latest works in this area is a study of Chaki et al. [8], which is devoted to various algorithms for extracting features from analysed images. At Porto University, a system was developed to automatically identify and collect traits from images of various leaves, which identified, collected, and preserved traits from 340 plant specimens [9]. However, it has one significant drawback - the classification accuracy is not high enough.

The aim of the study is to develop information technology to account the influence of various methods of data reduction on the results of classification of plant leaves.

Based on the analysis of previous studies in this area, it can be concluded that the application of reduction methods, as well as the assessment of their influence on the quality of leaf classification, in particular on classification by softmax layer, performed by the authors, represents a scientific novelty.

2. Proposed methods

To improve the accuracy of leaf classification, an approach based on the use of neural networks was used, namely, a multilayer perceptron (with two and three hidden layers) and a softmax layer [10].

The process of training networks can take a long time, so the space of features of the analysed data has been reduced. There are many ways to reduce the dimension of the feature space, the most popular being: factor analysis [11-13] and auto-encoder [10, 14-15].

A similar combination of data reduction methods, selection criteria for the number of factors and the types of neural networks used was performed by the authors for the first time.

Before the classification, the initial sample was divided into training and testing samples. The training sample comprised 70% of the total number of observations in the original sample, the testing sample included all the remaining 30%. The observations of each sample are unique.

The implementation of the neural network approach to solving the classification problem with preliminary data reduction was performed in the MATLAB system using the Statistics and Machine Learning Toolbox, Deep Learning Toolbox extension packages.

3. Data reduction

When using factor analysis to perform data reduction, the correlation and covariance matrices are calculated for all 14 attributes. Based on (1), the maximum number of factors cannot exceed 9.

\[
d + m \leq (d - m)^2,
\]

where \(d\) – number of signs, \(m\) – number of factors.

The main difficulty of factor analysis is the selection and interpretation of the main factors. There are several commonly used criteria for determining the number of factors. Some of them are alternative in relation to others, and some of these criteria can be used together so that one complements the other.

In the study, the following two criteria are used to determine the number of factors.

1. Kaiser criterion or criterion of eigenvalues. Only factors with eigenvalues equal to or greater than 1 are selected. This means that if the factor does not highlight the variance equivalent to at least the variance of one variable, then it is omitted. The eigenvalues of the factors are calculated based on the calculated values of the correlation matrix. Table 1 shows the eigenvalues for each factor.
Table 1. Eigenvalues of the factors.

| Factor | Eigenvalue | Factor | Eigenvalue |
|--------|------------|--------|------------|
| 1      | 5.6829     | 8      | 0.1138     |
| 2      | 4.1948     | 9      | 0.0734     |
| 3      | 2.1021     | 10     | 0.0453     |
| 4      | 0.7355     | 11     | 0.0246     |
| 5      | 0.4377     | 12     | 0.0175     |
| 6      | 0.3888     | 13     | 0.0121     |
| 7      | 0.1712     | 14     | 0.0002     |

Based on the obtained values, the optimal number of factors according to the Kaiser criterion is 3.

2. The criterion for the proportion of reproducible dispersion. Factors are ranked by the proportion of reproducible variance, when the percentage of variance is insignificant, the selection should be stopped. Table 2 contains the per cent of reproducible dispersion for each factor.

Based on them, the optimal number of factors by the criterion of the fraction of reproducible dispersion is 5. In this case, the sum of per cent of variances for the selected factors is 99.9169%.

Table 2. Percentage of reproducible variance of factors.

| Factor | Dispersion per cent | Factor | Dispersion per cent |
|--------|---------------------|--------|---------------------|
| 1      | 82.1774             | 8      | 0.0131              |
| 2      | 13.3138             | 9      | 0.0099              |
| 3      | 3.6406              | 10     | 0.0008              |
| 4      | 0.4372              | 11     | 0.0005              |
| 5      | 0.3478              | 12     | 0.0000              |
| 6      | 0.0385              | 13     | 0.0000              |
| 7      | 0.0203              | 14     | 0.0000              |

Factor analysis was performed with varimax rotation.

The simplest architecture of the auto-encoder is a direct distribution network without feedback, most similar to the perceptron and containing an input layer, an intermediate hidden layer and an output layer [15].

The output layer of the auto-encoder should contain as many neurons as the input layer, and the hidden layer should contain the number of neurons less than the number of neurons in the input (output) layer.

When using an autoencoder for reduction, 3 and 5 neurons were selected in a hidden layer, i.e. two auto-encoders were implemented. These values were set for a comparative analysis of various methods of data reduction. Sigmoidal activation functions were also used. For training, the Moller algorithm was used. The maximum number of learning iterations was set, which cannot exceed 5000.

After training, each auto-encoder was used for coding - a training sample was submitted to its input. The result of the work of each auto-encoder is a matrix containing encoded values; in the first matrix there are 3 columns, in the second there are 5 of them.

4. Classification
Classification refers to a learning strategy with a teacher, which is also called supervised or guided training.

This classification problem is most often solved by applying a multilayer perceptron.

In a multilayer neural network, neurons are arranged in layers. The neurons of the first layer receive input signals, convert them and transmit them to the neurons of the second layer, then the second layer
is triggered, etc. to the last layer that produces the output signals. Such a network is called a multilayer perceptron.

The intermediate layers between the external input signal and the output layer are called hidden.

In the work for training the network, the Levenberg-Marquardt algorithm was used, which is designed to optimize the parameters of nonlinear regression models. It is assumed that the standard error of the model in the training set is used as the optimization criterion. The algorithm consists in the sequential approximation of the given initial values of the parameters to the desired local optimum.

Two-layer and three-layer perceptrons were used in the study.

To solve the classification problem, one can also use the softmax layer, the use of which can often be observed in the architecture of reconciliation neural networks.

Its architecture is trivial: it consists of a single layer using the softmax activation function. The main advantage of the soft maximum function is that it uses an exponent.

The soft maximum activation function allows us to interpret the output values of neurons as some probability with which the vector belongs to the class, and due to the use of the exponent, only one neuron has a value close to unity.

Classification results without data reduction are given in Table 3.

| Network type         | The number of training iterations | Training time, s | Error rate |
|----------------------|-----------------------------------|------------------|------------|
| Perceptron (2 layers)| 20                                | 13.2501          | 18.6275    |
|                      | 40                                | 26.6774          | 19.6078    |
|                      | 80                                | 52.8668          | 29.4118    |
| Perceptron (3 layers)| 20                                | 39.5102          | 16.6667    |
|                      | 40                                | 80.7107          | 18.6275    |
| Softmax layer        | 4432                              | 6.4060           | 18.6275    |

The results of classifications based on reduced data using factor analysis and on data encoded using auto-encoders are shown in Table 4.

| Network type         | Number of factors | The number of training iterations | Factor values | Coded values |
|----------------------|-------------------|-----------------------------------|---------------|--------------|
|                      |                   | Training time, s | Error rate | Training time, s | Error rate |
| Perceptron (2 layers)| 3                 | 20                 | 6.7075     | 36.2745      | 1.6943      | 43.1373    |
|                      |                   | 40                 | 12.0590    | 36.2745      | 17.1723     | 41.1765    |
|                      |                   | 80                 | 27.7374    | 40.1961      | 31.7730     | 44.1176    |
|                      |                   | 160                | 59.5023    | 40.1961      | 61.4729     | 41.1765    |
|                      |                   | 20                 | 7.7524     | 36.2745      | 7.6184      | 50.098     |
|                      |                   | 40                 | 15.6403    | 34.3137      | 15.7994     | 45.098     |
|                      |                   | 80                 | 30.9774    | 36.2745      | 39.3720     | 39.2157    |
|                      |                   | 160                | 70.0819    | 39.2157      | 68.5313     | 39.2157    |
|                      |                   | 20                 | 32.2916    | 39.2157      | 29.2297     | 44.1176    |
|                      |                   | 3                  | 60         | 56.8962      | 40.1961     | 57.2131    |
|                      |                   | 40                 | 83.5421    | 47.0588      | 85.0780     | 42.1569    |
|                      |                   | 160                | 50.0000    | 50.0000      | 107.7826    | 37.2549    |
| Perceptron (3 layers)| 5                 | 20                 | 32.034     | 29.3725      | 32.0910     | 43.1373    |
|                      |                   | 40                 | 60.4611    | 33.3333      | 63.0195     | 38.2353    |
|                      |                   | 60                 | 94.2608    | 50.0000      | 107.7826    | 37.2549    |
5. Conclusion
As a result of the study, the following conclusions were obtained:

- Neural networks trained in classification according to data reduced by factor analysis have a smaller percentage of incorrectly classified observations.
- Neural networks using reduced data obtained by factor analysis with the choice of the optimal number of factors by the criterion of the reproducible dispersion fraction made fewer errors in the classification of observations from the test sample.
- When using the softmax layer for classification, the best results were obtained, especially when using factor analysis with 5 factors.
- On average, the training time of each network was reduced by about 2-3 times.
- Auto-encoder is a more modern way of performing data reduction, however, in this sample, its use leads to a significantly higher percentage of incorrectly classified observations due to the previously described properties of the sample.
- Neural networks using reduced data obtained by factor analysis with the choice of the optimal number of factors by the criterion of the reproducible dispersion fraction made fewer errors in the classification of observations from the test sample.

References
[1] Cope J, Corney D, Clark J, Remagnino P and Wilkin P 2012 Plant species identification using digital morphometrics: A review Expert Systems with Applications 39 7562-73
[2] Novotny P and Suk T 2013 Leaf recognition of woody species in Central Europe Biosystem engineering 115 444-52
[3] Zhang S, Lei Y, Dong T and Zhang X 2013 Label propagation based supervised locality projection analysis for plant leaf classification Pattern Recognition Letters 46 1891-7
[4] Chaki J, Parekh R and Bhattacharya S 2015 Plant leaf recognition using texture and shape features with neural classifiers Pattern Recognition Letters 58 61-8
[5] Aakif A and Khan M F 2015 Automatic classification of plants based on their leaves Biosystem engineering 139 66-75
[6] Munisami T, Ransum M, Kishnah S and Pudaruth S 2015 Plant leaf recognition using shape features and colour histogram with k-nearest neighbour classifiers Procedia Computer Science 58 740-7
[7] Sekeroglu B, and Inan Y 2016 Leaves recognition system using a neural network Procedia Computer Science 102 578-82
[8] Chaki J, Parekh R and Bhattacharya S 2018 Plant leaf classification using multiple descriptors: A hierarchical approach Journal of King Saud University – Computer and Information Sciences
[9] Silva F B, Marcal R S, and Almeida da Silva M. 2013 Evaluation of features for leaf discrimination Springer Lecture Notes on Computer Science 7950 197-205
[10] Adem K, Kilicarslan S and Comert O 2019 Classification and diagnosis of cervical cancer with stacked autoencoder and softmax classification Expert Systems With Applications 115 557-64
[11] Tezzele M, Salmoiraghi F, Mola A and Rozza G 2017 Dimension reduction in heterogeneous parametric spaces with application to naval engineering shape design problems Advanced Modeling and Simulation in Engineering Sciences 5 25
[12] Marzouk M and Elkadi M 2016 Estimating water treatment plants costs using factor analysis and artificial neural networks Journal of Cleaner Production 112 4540-9
[13] Yang X, Huang K, Zhang R, Goulermas J and Hussain A 2018 A new two-layer mixture of factor
analyzers with joint factor loading model for the classification of small dataset problems
Neurocomputing 312 352-63
[14] Kannadasan K, Damodar R E and Venkatanaresbabu K 2018 Type 2 diabetes data classification
using stacked autoencoders in deep neural networks Clinical Epidemiology and Global Health
[15] Hinton G E and Salakhutdinov R R 2006 Reducing the dimensionality of data with neural
networks Science 313 504-7