DOI: 10.20472/IAC.2019.047.030

ELDA XHUMARI
University of Tirana, Faculty of Natural Sciences, Department of Informatics, Albania

JULIAN FEJZAJ
University of Tirana, Faculty of Natural Sciences, Department of Informatics, Albania

USAGE OF ARTIFICIAL NEURAL NETWORKS IN DATA CLASSIFICATION

Abstract:

Data classification is broadly defined as the process of organizing data by respective categories so that it can be used and protected more efficiently. Data classification is performed for different purposes, one of the most common is for preserving data privacy. Data classification often includes a number of attributes, determining the type of data, confidentiality, and integrity. Neural networks help solve different problems. They are very good at data classification problems, they can classify any data with arbitrary precision.

Keywords:

Artificial Neural Networks, Data Classification, Naïve Bayes, Discriminant Analysis, Nearest Neighbor

JEL Classification: C45

1 Introduction

When solving classification problems, it is necessary to assign the available samples *for example, medical examination data) to certain classes. There are several ways to present data. The most common is the way in which a sample is represented by a vector. The components of this vector are the various characteristics of the sample that influence the decision about which class this sample belongs to. For example, for medical tasks, the data from the patient's medical record may be components of this vector. Thus, on the basis of some information about the example, it is necessary to determine which class it can be attributed to. The classifier thus relates the object to one of the classes in accordance with a certain partitioning of the N-dimensional space, which is called the input space, and the dimension of this space is the number of components of the vector.

First of all, you need to determine the level of complexity of the system. In real-life tasks, a situation often arises when the number of samples is limited, which makes it difficult to determine the complexity of the problem. It is possible to distinguish three main levels of difficulty:

1. When classes can be divided by straight lines (or hyperplanes, if the input space has a dimension greater than two) - the so-called linear separability.
2. When classes cannot be separated by lines (planes), but it is possible to separate them using a more complex division — non-linear separability.
3. When classes intersect and we can speak only about probabilistic separability.

After preprocessing, we must obtain a linearly separable problem, since after this the construction of the classifier is greatly simplified. Unfortunately, when solving real problems, we have a limited number of samples, on the basis of which the classifier is built. At the same time, we cannot carry out such preprocessing of data, at which linear separability of samples will be achieved.

Neural networks are the most effective way to classify, because they actually generate a large number of regression models (which are used in solving classification problems by statistical methods). Unfortunately, applying neural networks in practical problems arise a number of problems. First of all, it is not known in advance what complexity (size) a network may need for sufficiently accurate mapping. This complexity may be prohibitively high, which will require a complex network architecture. Sometimes the simplest single-layer neural networks are capable of solving only linearly separable tasks. This limitation is surmountable when using multilayer neural networks. In general, we can say that in a network with one hidden layer, the vector corresponding to the input sample is transformed by the hidden layer into some new space, which can have a different dimension, and then the hyperplanes corresponding to the neurons of the output layer divide it into classes. Thus, the network recognizes not only the attributes of the source data, but also the "attributes of the attributes" (metadata) formed by the hidden layer.

2 Data classification algorithm based on neural networks

To solve classification problems using neural networks, the process consists of the following steps:

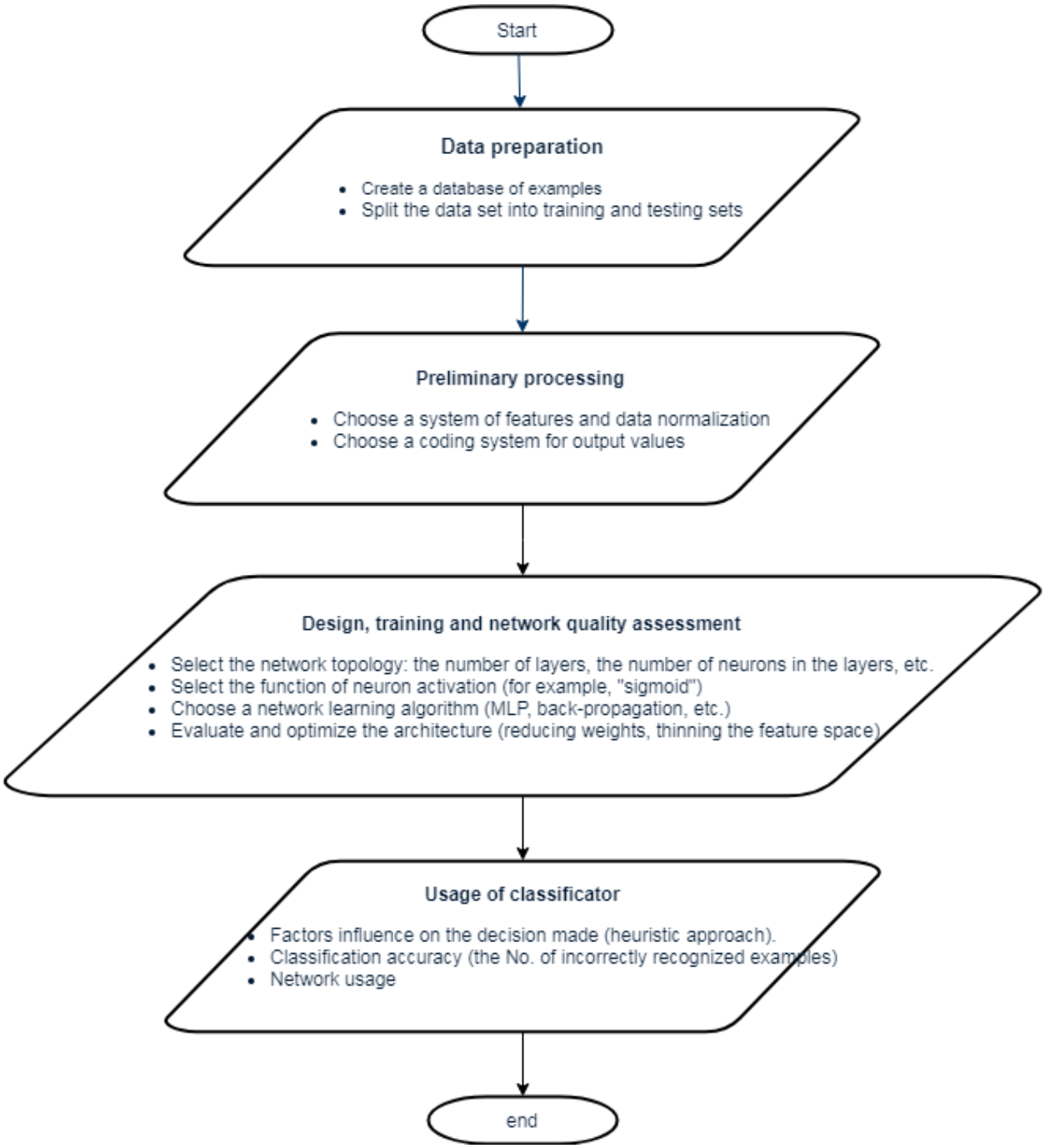


FIGURE 1. DATA CLASSIFICATION ALGORITHM USING NEURAL NETWORKS

To classify data, it is necessary to determine which parameters influence the decision on which class the sample belongs to. There may be two problems:

1. If the number of parameters is small, then a situation may arise in which the same set of input data corresponds to examples in different classes. Then it is impossible to train a neural network, and the system will not work correctly (it is impossible to find a minimum that corresponds to such a set of initial data). The source data must be consistent. To solve this problem, it is necessary to increase the dimension of the feature space (the number of components of the input vector corresponding to the sample).
2. If we increase the number of parameters, but with an increase in the dimension of the feature space, a situation may arise when the number of examples may become insufficient for network training, and instead of generalization, it will simply remember the examples from the training sample and will not be able to function correctly. Thus, in determining the signs, it is necessary to find a compromise with their number.

It is necessary to determine the method of presenting the input data for the neural network, i.e. determine the method of regulation. Normalization [5] is necessary because neural networks work with data represented by numbers in the range [0..1], and the source data can have an arbitrary range or even be non-numeric data. In this case, various methods are possible, ranging from simple linear transformation to the required range and ending with multidimensional analysis of parameters and nonlinear normalization depending on the influence of parameters on each other.

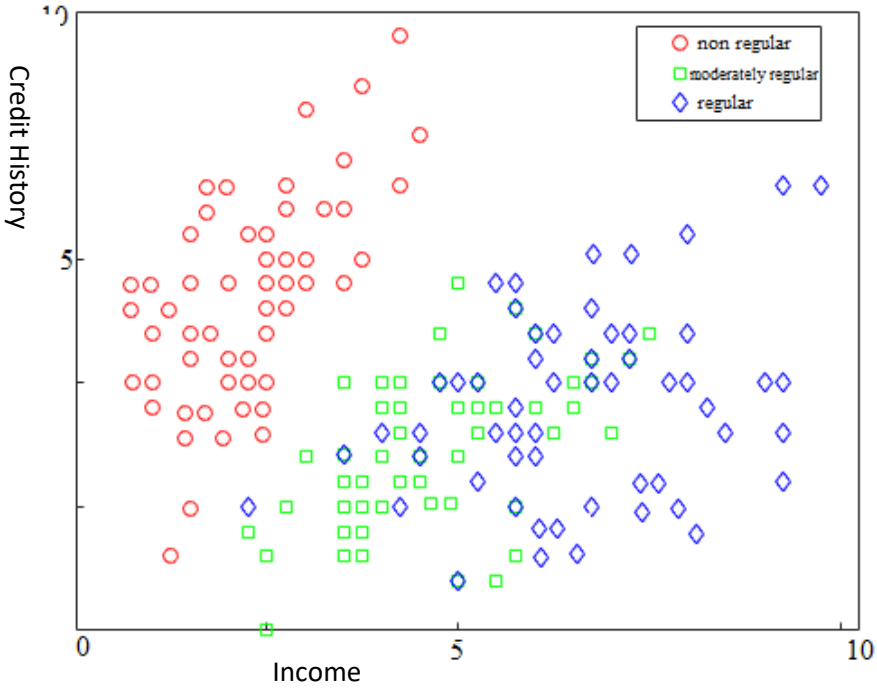
3 Experiments

For the purpose of this paper, experiments have been conducted regarding to the classification of a bank's clients, based on the data that the bank holds for them, to determine whether the client can be a regular loan payer or not.

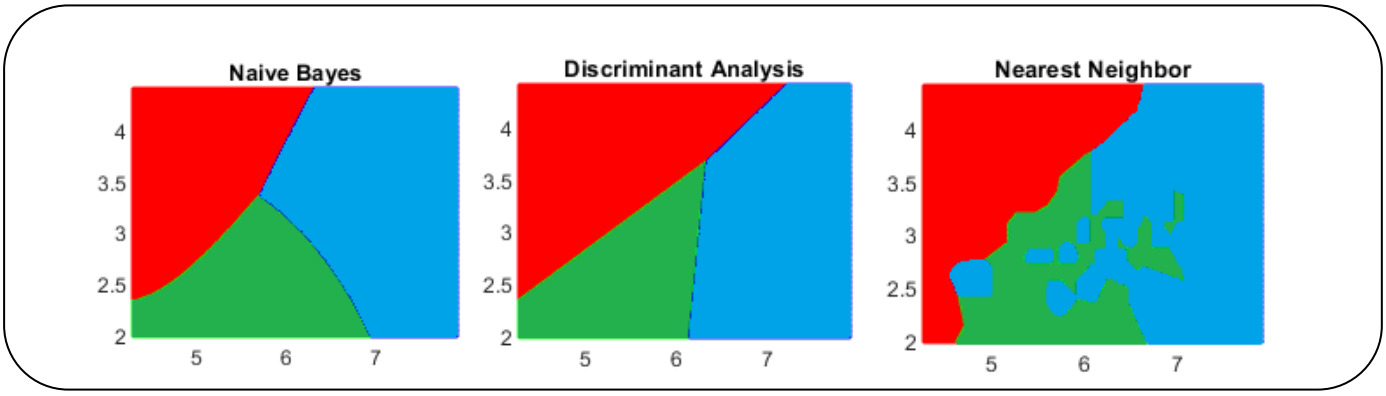
For client classification, an input vector is created, which contains personal data as well as data on the income or credit history of clients in the bank.

Each factor is normalized through the following formula: $\mathbf{x_i=x_i/x_{max}}\ x\ \mathbf{10,}$ so the values vary in the range of [0..10].The input database contains approximately 180 samples. For data classification we used 3 different classifiers: *Naive Bayes*, *Discriminant Analysis* and *Nearest Neighbor*. For modeling the use of neural networks for classification problem was used the application Deductor, in which the category of problems when using NN is determined automatically.

The data is visualized using a scatter plot as below:



The results of 3 algorithms are visualized as below:



4 Conclusions

Three algorithms were used during this work: Naïve Bayes, Discriminant Analysis, Nearest Neighbor. Data used as input for neural networks was taken randomly from a database of a bank. It turned out that each algorithm generated different rules.

5 References

Shekhar Pandey, Supriya Muthuraman, Supriya Muthuraman, Abhilash Shrivastava: Data Classification Using Machine Learning Approach

J. Weng, "Natural and Artificial Intelligence: Introduction to Computational Brain-Mind, BMI Press, ISBN 978-0985875725, 2012.

Ludovic Denoyer, Patrick Gallinari: Bayesian Network Model for Semi-Structured Document Classification

https://it.mathworks.com/help/stats/classificationlearner-app.html

http://masters.donntu.org/2012/fknt/gaydukov/library/neuro.pdf