Chapter 11
Educational Software Based on Matlab GUIs for Neural Networks Courses

Pablo Díaz-Moreno
University of Valencia, Spain

José M. Martínez-Martínez
University of Valencia, Spain

Juan José Carrasco
University of Valencia, Spain

Pablo Escandell-Montero
University of Valencia, Spain

Emilio Soria-Olivas
University of Valencia, Spain

Juan Gómez-Sanchis
University of Valencia, Spain

ABSTRACT

Neural Networks (NN) are one of the most used machine learning techniques in different areas of knowledge. This has led to the emergence of a large number of courses of Neural Networks around the world and in areas where the users of this technique do not have a lot of programming skills. Current software that implements these elements, such as Matlab®, has a number of important limitations in teaching field. In some cases, the implementation of a MLP requires a thorough knowledge of the software and of the instructions that train and validate these systems. In other cases, the architecture of the model is fixed and they do not allow an automatic sweep of the parameters that determine the architecture of the network. This chapter presents a teaching tool for the its use in courses about neural models that solves some of the above-mentioned limitations. This tool is based on Matlab® software.

INTRODUCTION

Artificial Neural Networks (ANN) are the most widely used models for the solution of a number of different problems, including classification, regression, time series forecasting and complex system modeling (Marsland, 2014). These models have been applied extensively during the two last decades in practically all knowledge areas (Rabunal, 2005). The multilayer perceptron is likely the most widespread neural model (Mohri, 2012), being many applications developed for this model. This fact has led to the existence

DOI: 10.4018/978-1-4666-8823-0.ch011

of a large number of courses of neural networks in all areas of knowledge, and the vast majority of them contain several laboratory sessions where it is necessary to apply a MLP implemented in a particular software package. There are plenty of commercial software (e.g. SPSS, MATLAB, Mathematica, etc.) and free distribution software (e.g. R, WEKA and SNNS) that implement these models. The implementation of a MLP in a given software package can be done by programming either in command line or in a script. This procedure requires a thorough knowledge of the software and of the instructions for train and validates these systems. This causes that non-expert users in this software cannot use it.

On the other hand, there exist graphical user interfaces that allow training and validating these models (such as SPSS and MATLAB). In these interfaces, the architecture of the model is fixed and the training process is carried out for obtaining the parameters of the neural model. The limitation of such interfaces is due to the fact that they do not allow an automatic sweep of the parameters that determine the architecture of the network (which is a standard procedure to obtain the optimal architecture). Nor, these interfaces enable to carry out different initializations of the learning algorithm, which is necessary due to the fact that most of these algorithms are based on gradient descent methods and require different initializations to avoid falling into a local minimum of the error function (Haykin, 2008). Thus, although an easy to use interface is available, it is not very useful because it is necessary to perform a large number of tests manually to achieve a useful neural model.

This chapter deals to solve this problem by developing a MATLAB GUI to be used in a course of neural networks, in which a number of different initializations can be carried out as well as a variation of the architecture in a simple and automatic way. In addition, it also allows analyzing the neural models obtained in a simple and fast way.

BACKGROUND

In the literature it is possible to find all kinds of educational software developed for different engineering courses related to data analysis and machine learning. In (Carrasco Fernández, 2012), the authors present an educational software developed using the MATLAB GUIDE tool. This software allows engineering students gain knowledge about data sets via the exploratory data analysis (EDA). This application includes models like clustering algorithms and self-organizing maps. However, does not include multilayer perceptrons. In (Deperlioglu, 2011) an educational tool to work with different kinds of neural network models is presented. The developed tool includes MLP, LVQ and SOM models. The design of the models was done visually and interactively but, in contrast to the proposed application in this chapter, confusion matrix or sensitivity-specificity histograms are not shown in the results. Moreover, in (Marković, 2014) a software system developed to support the teaching of Intelligent Systems is presented. The tool includes decision trees (ID3), clustering (kmeans), Naive Bayes, and perceptron models. In works (Ugur, 2010; Hwang, 2003; García Roselló, 2003; Zatarain, 2011) similar applications are presented.

MULTILAYER PERCEPTRON

Multilayer Perceptron (MLP) is probably the most widely used neural network due to its interesting characteristics: universal function approximator and non-linear classifier. The MLP has shown excellent results in many different applications. In the above subsection, the elementary units which form the MLP, the so-called neurons, are presented. Next subsections are devoted to explain how it works, the