Approximation of Multi-pattern to Single-Pattern Functions by Combining FeedForward Neural Networks and Support Vector Machines

Vijayanarasimha Hindupur Pakka

Department of Electrical Engineering, Indian Institute of Science, Bangalore 560 012, India
vijay@ee.iisc.ernet.in, hpvijaynarasimha@hotmail.com

Abstract. In many fields there are situations encountered, where a function has to be estimated to determine its output under new conditions. Some functions have one output corresponding to differing input patterns. Such types of functions are difficult to map using a function approximation technique such as that employed by the Multilayer Perceptron Network. Hence to reduce this functional mapping to Single Pattern-to-Single Pattern type of condition, and then effectively estimate the function, we employ classification techniques such as the Support Vector Machines. This paper describes in detail such a combined technique, which shows excellent results for practical applications.

1 Introduction

Function approximation (FA) or function estimation is typically the estimation of the output of an unknown function for a new input pattern, provided the function estimator is given sufficient training sets such that the unknown parameters defining the function are estimated through a learning strategy. FA is more commonly known as regression in statistical theory. This function is usually a model of a practical system. The training sets are obtained usually by simulation of the system in real time. If a training set is given by,

\[
\{(x_1,y_1), (x_2,y_2), (x_3,y_3), \ldots, (x_N,y_N)\}
\]

(1)

\(x = \) input pattern vector, \(y = \) target vector, \(N = \) number of patterns.

Then we need to estimate the functional relation between \(x\) and \(y\) i.e.,

\[
y = c_i f(x; t_i)
\]

(2)

\(c_i = \) constants in the function, \(t_i = \) parameters of the function, \(f:S \rightarrow \mathbb{R}\), where \(S = \left\{x \in \mathbb{R}^n \mid a_i \leq x_i \leq b_i, 1 \leq i \leq n \right\}\) is a closed bounded region.

FA by multilayer perceptron networks like the FeedForward Neural Networks (FFNNs) is proven to be very efficient [2], [3], considering various learning strategies like the simple Back Propagation or the robust Levenberg Marquardt and Conjugate Gradient approaches. Assume \(f(x_1, x_2, \ldots, x_{m_0})\) is the approximate function with \(m_0\) variables. Now if the true function is \(F(\cdot)\), then the FFNN equates this to
\[ \sum_{i=1}^{m} \alpha_i \phi \left( \sum_{j=1}^{m} w_{ij} x_{ij} + b_i \right). \]  

(3)

Now the objective is to find the parameter values of \( m_i \) and values of all \( w_{ij} \)'s, \( b_i \)'s and \( \alpha_i \)'s, such that \( |F(x) - f(x)| < \epsilon \), for all \( x_1, x_2, \ldots, x_{m_0} \).

## 2 Multi-pattern to Single-Pattern Functions

Let us look at the problem of FA as a mapping problem, where, by one-to-one mapping we mean that each input vector has a corresponding and unique target vector. These mappings are simple to model by FFNNs.

![Fig. 1a. Sine wave characteristics of a sample system. Forward sine represents cycle ‘a’ and backward sine represents cycle ‘b’](image)

But, this is not the case in many fields. For example, let us study the case of a sine function. Assume that a system has the characteristic shown in figure 1a, which has to be estimated. The data sets that are available for training of the FFNN are the data corresponding to the two cycles a and b. For convenience, figure 1a is redrawn as figure 1b. Inputs \( X_{a1} \) and \( X_{b1} \) have same output \( Y_i \). This value is stored by the FFNN in the form of a straight line. For both the inputs running through one cycle, we have a set of such straight lines with varying amplitudes (figure 2). Let us name this type of mapping as **Two-way mapping** or **Multi-Pattern to Single-Pattern mapping** in general, because, estimation of the actual function is the first FA problem, and estimation of the shapes of the lines in figure 2 is the second FA problem, i.e., two different input patterns \( X_{a1} \) and \( X_{b1} \) correspond to a single output pattern \( Y_i \).

Now suppose there exists an intermediate sine cycle (p) between cycles a and b. If p has similar shape and size as of a and b, then, we can estimate its \( Y \) throughout the cycle just by noting the \( Y \) values at corresponding intermediate point \( X_p \) in figure 2. This estimation of \( Y \) turns out to be equal to that at \( X_a \) or \( X_b \). Now instead of p being similar to a or b, suppose it to be of different size as shown in figure 3. Then, if \( Y \)'s are estimated at \( X=X_p \), the results would not match with that of the true function represented by p. This is due to the fact that the curves joining the two sets of vertical points in figure 2 are still straight lines, though in reality they are of the shape of curves with amplitudes (\( Y \)'s) at \( X_p \) different from that at \( X_a \) or \( X_b \).