A New Learning Method for Piecewise Linear Regression

Giancarlo Ferrari-Trecate[1] and Marco Muselli[2]

[1] INRIA, Domaine de Voluceau
Rocquencourt - B.P.105, 78153 Le Chesnay Cedex, France
Giancarlo.Ferrari-Trecate@inria.fr
[2] Istituto per i Circuiti Elettronici - CNR
via De Marini, 6 - 16149 Genova, Italy
muselli@ice.ge.cnr.it

Abstract. A new connectionist model for the solution of piecewise linear regression problems is introduced; it is able to reconstruct both continuous and non continuous real valued mappings starting from a finite set of possibly noisy samples. The approximating function can assume a different linear behavior in each region of an unknown polyhedral partition of the input domain.
The proposed learning technique combines local estimation, clustering in weight space, multicategory classification and linear regression in order to achieve the desired result. Through this approach piecewise affine solutions for general nonlinear regression problems can also be found.

1 Introduction

The solution of any learning problem involves the reconstruction of an unknown function $f : X \to Y$ from a finite set S of samples of f (*training set*), possibly affected by noise. Different approaches are usually adopted when the range Y contains a reduced number of elements, typically without a specific ordering among them (*classification problems*) or when Y is an interval of the real axis with the usual topology (*regression problems*).

However, applications can be found, which lie on the borderline between classification and regression; these occur when the input space X can be subdivided into disjoint regions X_i characterized by different behaviors of the function f to be reconstructed. One of the simplest situations of such kind is piecewise linear regression (PLR): in this case X is a polyhedron in the n-dimensional space $\mathbb{R}^n$ and $\{X_i\}_{i=1}^s$ is a polyhedral partition of X, i.e. $X_i \cap X_j = \emptyset$ for every $i, j = 1, \ldots, s$ and $\bigcup_{i=1}^s X_i = X$. The target of a PLR problem is to reconstruct an unknown function $f : X \to \mathbb{R}$ having a linear behavior in each region X_i

$$f(\boldsymbol{x}) = w_{i0} + \sum_{j=1}^{n} w_{ij} x_j \quad \text{if } \boldsymbol{x} \in X_i \tag{1}$$

when only a training set S containing m samples $(\boldsymbol{x}_k, y_k)$, $k = 1, \ldots, m$, is available. The output y_k gives a noisy evaluation of $f(\boldsymbol{x}_k)$, being $\boldsymbol{x}_k \in X$; the region

X_i to which $\boldsymbol{x}_k$ belongs is not known in advance. The scalars $w_{i0}, w_{i1}, \ldots, w_{in}$, for $i = 1, \ldots, s$, characterize the function f and their estimate is a target of the PLR problem; for notational purposes they will be included in a vector $\boldsymbol{w}_i$.

Since regions X_i are polyhedral, they are defined by a set of l_i linear inequalities, which can be written in the following form:

$$A_i \begin{bmatrix} 1 \\ \boldsymbol{x} \end{bmatrix} \geq 0 \tag{2}$$

where A_i is a matrix with l_i rows and $n+1$ columns, whose estimate is still an output of the reconstruction process for every $i = 1, \ldots, s$.

The target of the learning problem is consequently twofold: to generate both the collection of regions X_i and the behavior of the unknown function f in each of them, by using the information contained in the training set. In these cases, classical learning algorithms for connectionist models cannot be directly employed, since they require some knowledge about the problem, which is not available a priori.

Several authors have treated this kind of problems [2–4, 7], providing algorithms for reaching the desired result. Unfortunately, most of them are difficult to extend beyond two dimensions [2], whereas others consider only local approximations [3, 4], thus missing the actual extension of regions X_i. In this contribution a new connectionist model for solving PLR problems is proposed, together with a proper learning algorithm that combines clustering, multicategory classification, and linear regression to select a reduced subset of relevant training patterns and to derive from them suitable values for the network weights.

2 The proposed learning algorithm

Following the general idea presented in [7], a connectionist model realizing a piecewise linear function f can be depicted as in Fig. 1. It is formed by three layers: the *hidden layer*, containing a linear neuron for each of the regions X_i, the *gate layer*, whose units delimit the extension of each X_i, and the *output layer*, that provides the desired output value for the pattern given as input to the network. The task of the gate layer is to verify inequalities (2) and to decide which of the terms z_i must be used as the output y of the whole network. Thus, the i-th unit in the gate layer has output equal to its input z_i if the corresponding constraint (2) is satisfied and equal to 0 in the opposite case. All the other units perform a weighted sum of their inputs; the weights of the output neuron, having no bias, are always set to 1.

As previously noted, the solution of a PLR problem requires a technique that combines classification and regression: the first has the aim of finding matrices A_i to be inserted in the gate layer of the neural network in Fig. 1, whereas the latter provides the weight vectors $\boldsymbol{w}_i$ for the input to hidden layer connections. The method we propose is summarized in Fig. 2; it is composed of four steps, each one devoted to a specific task.

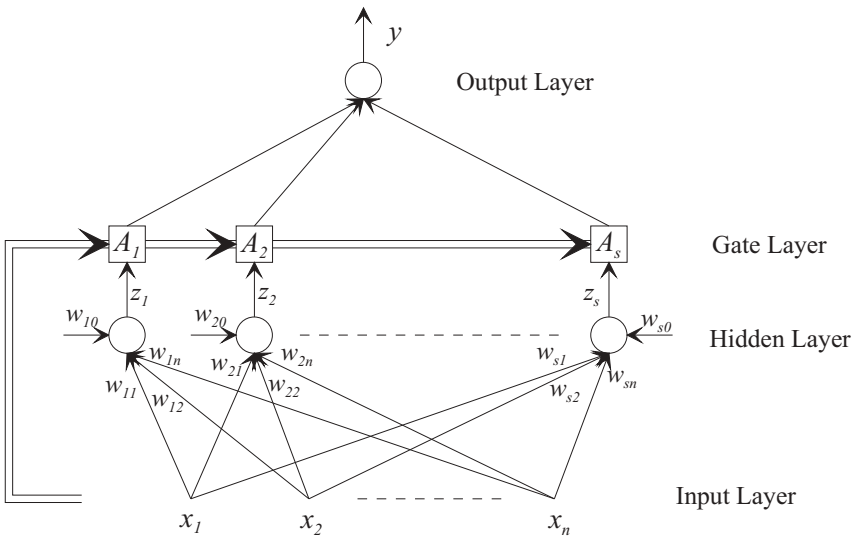

Fig. 1. Connectionist model realizing a piecewise linear function.

The first one (Step 1) aims at obtaining a first estimate of the weight vectors $\boldsymbol{w}_i$ by performing local linear regressions based on small subsets of the whole training set S. In fact, points $\boldsymbol{x}_k$ that are close to each other are likely to belong to the same region X_i. Then, for each sample $(\boldsymbol{x}_k, y_k)$, with $k = 1, \ldots, m$, we build a local dataset C_k containing $(\boldsymbol{x}_k, y_k)$ and the $c-1$ distinct pairs $(\boldsymbol{x}, y) \in S$ that score the lowest values of the distance $\|\boldsymbol{x}_k - \boldsymbol{x}\|$. It can be easily seen that most sets C_k, called *pure*, contain samples belonging to the same region X_i, while the remaining ones, named *mixed*, include input patterns deriving from different X_i. These lead to wrong estimates for $\boldsymbol{w}_i$ and consequently their number must be kept minimum.

Denote with $\boldsymbol{v}_k$ the weight vector produced through the local linear regression on the samples $(\boldsymbol{x}_k^1, y_k^1)$, $(\boldsymbol{x}_k^2, y_k^2)$, $\ldots$, $(\boldsymbol{x}_k^c, y_k^c)$ of C_k. If $\boldsymbol{\Phi}_k$ and $\boldsymbol{\psi}_k$ are defined as

$$\boldsymbol{\Phi}_k = \begin{bmatrix} \boldsymbol{x}_k^1 & \boldsymbol{x}_k^2 & \ldots & \boldsymbol{x}_k^c \\ 1 & 1 & \ldots & 1 \end{bmatrix}' , \qquad \boldsymbol{\psi}_k = [y_k^1 \; y_k^2 \; \ldots \; y_k^c]'$$

being $'$ the transpose operator, $\boldsymbol{v}_k$ is generated through the well known formula

$$\boldsymbol{v}_k = (\boldsymbol{\Phi}_k' \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_k' \boldsymbol{\psi}_k$$

A classical result in least squares theory allows also to obtain the empirical covariance matrix V_k of the vector $\boldsymbol{v}_k$ and the scatter matrix [5] Q_k

$$V_k = \frac{S_k}{c-n+1} (\boldsymbol{\Phi}_k' \boldsymbol{\Phi}_k)^{-1} , \qquad Q_k = \sum_{i=1}^{c} (\boldsymbol{x}_k^i - \boldsymbol{m}_k)(\boldsymbol{x}_k^i - \boldsymbol{m}_k)'$$

ALGORITHM FOR PIECEWISE LINEAR REGRESSION

1. *(Local regression)* For every $k = 1, \ldots, m$ do

1a. Build the local dataset C_k containing the sample $(\boldsymbol{x}_k, y_k)$ and the pairs $(\boldsymbol{x}, y) \in S$ associated with the $c-1$ nearest neighbors $\boldsymbol{x}$ to $\boldsymbol{x}_k$.

1b. Perform a linear regression to obtain the weight vector $\boldsymbol{v}_k$ of a linear unit fitting the samples in C_k.

2. *(Clustering)* Perform a proper clustering process in the space $\mathbb{R}^{n+1}$ to subdivide the set of weight vectors $\boldsymbol{v}_k$ into s groups U_i.

3. *(Classification)* Build a new training set S' containing the m pairs $(\boldsymbol{x}_k, i_k)$, being U_{i_k} the cluster including $\boldsymbol{v}_k$. Train a multicategory classification method to produce the matrices A_i for the regions $\hat{X}_i$.

4. *(Regression)* For every $i = 1, \ldots, s$ perform a linear regression on the samples $(\boldsymbol{x}, y) \in S$, with $\boldsymbol{x} \in \hat{X}_i$, to obtain the weight vector $\boldsymbol{w}_i$ for the i-th unit in the hidden layer.

Fig. 2. Proposed learning method for piecewise linear regression.

being $S_k = \boldsymbol{\psi}_k' \left(I - \Phi_k (\Phi_k' \Phi_k)^{-1} \Phi_k'\right) \boldsymbol{\psi}_k$ and $\boldsymbol{m}_k = \sum_{i=1}^{c} \boldsymbol{x}_k^i / c$

Then, consider the *feature vectors* $\boldsymbol{\xi}_k = [\boldsymbol{v}_k' \; \boldsymbol{m}_k']'$, for $k = 1, \ldots, m$; they can be approximately modeled as the realization of random vectors with covariance matrix

$$R_k = \begin{bmatrix} V_k & 0 \\ 0 & Q_k \end{bmatrix}'$$

If the generation of the samples in the training set is not affected by noise, most of the $\boldsymbol{v}_k$ coincide with the desired weight vectors $\boldsymbol{w}_i$. Only mixed sets C_k yield spurious vectors $\boldsymbol{v}_k$, which can be considered as outliers. Nevertheless, even in presence of noise, a clustering algorithm (Step 2) can be used to determine the sets U_i of feature vectors $\boldsymbol{\xi}_k$ associated with the same $\boldsymbol{w}_i$. If the number s of regions is fixed beforehand, a proper version of the K-means algorithm [6] can be adopted. It uses the following cost functional

$$J\left(\{U_i\}_{i=1}^{s}, \{\boldsymbol{\mu}_i\}_{i=1}^{s}\right) = \sum_{i=1}^{s} \sum_{\boldsymbol{\xi}_k \in U_i} \|\boldsymbol{\xi}_k - \boldsymbol{\mu}_i\|_{R_k^{-1}}^2$$

where $\boldsymbol{\mu}_i$ is the center of the cluster U_i. This choice allows to recover the influence of poor initialization and of outliers on the clustering process.

The sets U_i generated by the clustering process induce a classification on the input patterns $\boldsymbol{x}_k$ belonging to the training set S, due to the chain of bijections among the set of input patterns $\boldsymbol{x}_k$, the collection of local datasets C_k, and the class of feature vectors $\boldsymbol{\xi}_k$. Now, if $\boldsymbol{\xi}_k \in U_i$ for a given i, it is likely that the local dataset C_k is fitted by the linear neuron with weight vector $\boldsymbol{w}_i$ and consequently $\boldsymbol{x}_k$ is located into the region X_i. An estimate $\hat{X}_i$ for each of these regions can then be determined by solving a linear multicategory classification

problem (Step 3), whose training set S' is built by adding as output to each input pattern $\boldsymbol{x}_k$ the index i_k of the set U_{i_k} to which the corresponding feature vector $\boldsymbol{\xi}_k$ belongs.

To avoid the presence of multiply classified points or of unclassified patterns in the input space, multicategory techniques [1] deriving from the Support Vector Machine approach and based on linear and quadratic programming can be employed. In this way the s matrices A_i for the gate layer are generated. Finally, weight vectors $\boldsymbol{w}_i$ for the neural network in Fig. 1 can be directly obtained by solving s linear regression problems (Step 4) having as training sets the samples $(\boldsymbol{x}, y) \in S$ with $\boldsymbol{x} \in \hat{X}_i$, being $\hat{X}_1, \ldots \hat{X}_s$ the regions found by the classification process.

3 Simulation Results

The proposed algorithm for piecewise linear regression has been tested on a two-dimensional benchmark problem, in order to analyze the quality of the resulting connectionist model. The unknown function to be reconstructed is

$$f(x_1, x_2) = \begin{cases} 3 + 4x_1 + 2x_2 & \text{if } 0.5x_1 + 0.29x_2 \geq 0 \text{ and } x_2 \geq 0 \\ -5 - 6x_1 + 6x_2 & \text{if } 0.5x_1 + 0.29x_2 < 0 \text{ and } 0.5x_1 - 0.29x_2 < 0 \\ -2 + 4x_1 - 2x_2 & \text{if } 0.5x_1 - 0.29x_2 \geq 0 \text{ and } x_2 < 0 \end{cases} \quad (3)$$

with $X = [-1, 1] \times [-1, 1]$ and $s = 3$. A training set S containing $m = 300$ samples (x_1, x_2, y) has been generated, according to the model $y = f(x_1, x_2) + \varepsilon$, being ε a normal random variable with zero mean and standard deviation $\sigma = 0.1$. The behavior of $f(x_1, x_2)$ together with the elements of S are depicted in Fig. 3a. Shaded areas in the (x_1, x_2) plane show the polyhedral regions X_i.

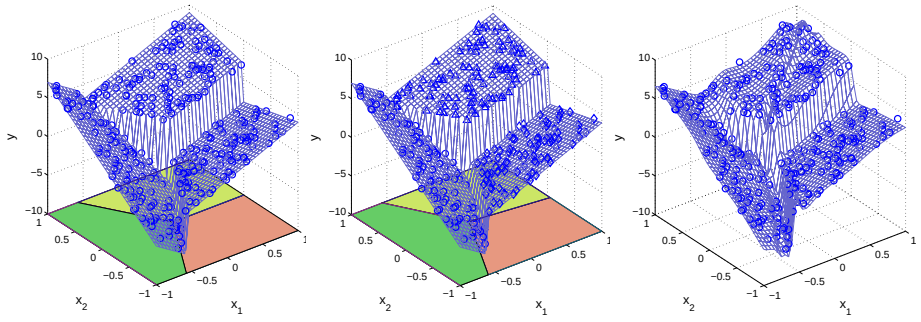

Fig. 3. Two dimensional benchmark problem: a) behavior of the unknown piecewise linear function and training set, b) simulation results obtained with the proposed algorithm and c) with a conventional two layer neural network.

The method described in Fig. 2 has been applied by choosing at Step 1 the value $c = 8$. The resulting connectionist model realizes the following function, graphically represented in Fig. 3b:

$$f(x_1, x_2) = \begin{cases} 3.02 + 3.91x_1 + 2.08x_2 & \text{if } 0.5x_1 + 0.28x_2 \geq 0 \text{ and } 0.12x_1 + x_2 \geq 0.07 \\ -5 - 5.99x_1 + 6.01x_2 & \text{if } 0.5x_1 + 0.28x_2 < 0 \text{ and } 0.5x_1 - 0.26x_2 < 0.04 \\ -2.09 + 4.04x_1 - 2.08x_2 & \text{if } 0.5x_1 - 0.26x_2 \geq 0.04 \text{ and } 0.12x_1 + x_2 < 0.07 \end{cases}$$

As one can note, this is a good approximation to the unknown function (3); the generalization Root Mean Square (RMS) error, estimated through a 10-fold cross validation, amounts to 0.296. Significant differences can only be detected at the boundaries between two adjacent regions X_i; they are mainly due to the effect of mixed sets C_k on the classification process.

As a comparison, a two layer neural network trained with a combination of Levenberg-Marquardt algorithm and Bayesian regularization yields the nonlinear approximating function shown in Fig. 3c. Discontinuities between adjacent regions are modeled at the expense of reducing the precision on flat areas. It is important to observe that neural networks are not able to reconstruct the partition $\{X_i\}_{i=1}^{s}$ of the input domain, thus missing relevant information for the problem at hand.

Seven hidden nodes are needed to produce this result; their number has been obtained by performing several trials with different values and by taking the best performance. A 10-fold cross validation scores an estimate of 0.876 for the generalization RMS error, significantly higher than that obtained through the PLR procedure.

References

1. E. J. Bredensteiner and K. P. Bennett, Multicategory classification by support vector machines. *Computational Optimizations and Applications*, **12** (1999) 53–79.
2. V. Cherkassky and H. Lari-Najafi, Constrained topological mapping for nonparametric regression analysis. *Neural Networks*, **4** (1991) 27–40.
3. C.-H. Choi and J. Y. Choi, Constructive neural networks with piecewise interpolation capabilities for function approximation. *IEEE Transactions on Neural Networks*, **5** (1994) 936–944.
4. J. Y. Choi and J. A. Farrell, Nonlinear adaptive control using networks of piecewise linear approximators. *IEEE Transactions on Neural Networks*, **11** (2000) 390–401.
5. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis.* (1973) New York: John Wiley and Sons.
6. G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, A Clustering Technique for the Identification of Piecewise Affine Systems. In *HSSC 2001*, vol **2304** of *Lecture Notes in Computer Science* (2001) Berlin: Springer-Verlag, 218–231.
7. K. Nakayama, A. Hirano, and A. Kanbe, A structure trainable neural network with embedded gating units and its learning algorithm. In *Proceedings of the International Joint Conference on Neural Networks* (2000) Como, Italy, III–253–258.