DESIGN OF CONTROLLER FOR MIXED DATA TYPE - A COMPOSITE FUZZY-NEURAL APPROACH

P. DAS¹*, D. BOSE²

¹SQC & OR Division, Indian Statistical Institute, Kolkata 700108, India
²HP Global Analytics, Bangalore 560 093, India
*Corresponding Author: dasprasun@rediffmail.com

Abstract

Input feature vectors in feed-forward neural network are generally processed by linear separation through plane or hyperplane. But discrete features are difficult to feed into these nonlinear computational processing elements. Even if the processing is carried out using some encoding mechanisms of quantization, retaining information of the original categorical variables appears to be difficult at times. This paper proposes a methodology to design an online controller through fuzzification of the discrete input features first and, then through supervised learning based on adaptive neuro-fuzzy inference system (ANFIS). Continuous feature spaces are processed simultaneously through multi-layered perceptrons (MLP) and both the systems are connected through a linear filter for prediction of response variables using log-likelihood cost function. This proposed architecture has been tested on standard data set for its efficacy. The system is proven to be more efficient as compared to the existing methods in terms of error metrics. The estimation is independent of the number of discrete input features and in case if the data set has several categorical features, this method will still be more effective than the available systems.

Keywords: Categorical variable, Feed forward neural network, Fuzzy inference system, Adaptive neuro fuzzy inference system, Least square estimation.

1. Introduction

Predictive analytics often deals with categorical data set and neural network being one of the most useful machine learning techniques faces a major bottleneck of not having a proper mechanism to handle categorical data in its architecture.

The issue gets complicated especially for nominal variables where natural ordering doesn’t hold true. For example, in a socio-economic scenario in US,
female workers are found to earn less than their male counterparts or black (or non-white) workers are found to earn less than whites [1]. Hence, such variables (sex or race stated) usually represent presence or absence of some quality or attribute [2], and are usually handled using dummy variables.

Neural network systems are simplified mathematical models of brain-like systems functioning as parallel distributed computing networks [3-7]. Essentially these processing units develop a boundary region (also known as decision boundary) to classify the observations based on a hyper-plane and predict the outcomes. In presence of discrete feature space, the inputs are converted to numerics, using different encoding systems. Successful modeling, therefore, needs a proper encoding system to retain the information on the discrete feature space. Bishop introduced the concept of conditional probability of the output variables in Mixture Density Network (MDN) which overcomes the Gaussian assumption in regression analysis [8]. Some popular efforts have been made in recent past to handle the categorical data using clustering methodologies where one of the best known methods for nominal attributes is CACTUS (CAtegorical ClusTerinG Using Summaries) [9]. Gibson et al. proposed an iterative approach for clustering the categorical data in dynamical systems using weight propagation method [10]. Hierarchical clustering (HCA) and fuzzy-c-means (FCM) are the most commonly used techniques for handling the categorical data [11]. The concept of fuzzy k-means technique has been extended based on a Tabu search based algorithm to the categorical domain [12] which was found out to be more efficient for handling the categorical data than regular fuzzy-k-means algorithm. In another work, the categorical variables, which are discontinuous in nature, are segregated from the continuous inputs by 1 out-of-n coding systems in a feed forward neural network [13]. However, this method does not include the interaction terms of the mixture variables.

The main focus of this work lies in designing neural systems in presence of categorical features with the help of fuzzy systems. The most influential work regarding Takagi-Sugeno-Kang (TSK) fuzzy model related to this domain has been done by Mannle [14]. Gabrys and Bargiela [15] proposed an approach to handle the categorical data using Simpson’s fuzzy min-max supervised neural network for clustering and classification. A similar work based on fuzzy neural network highlights the concept of imputation for categorical missing data [16]. Our basic
aim in this work is to develop a comprehensive method which can handle the mixed input variables and successfully perform the prediction modeling.

2. Plan of Work

In our proposed method, categorical variables are fuzzified depending upon the rules and the continuous variables are handled as regular neural network inputs. The resultant outcome of two parallel processors will give rise two continuous outputs and finally the net outcome will be obtained through using a logical linear filter (transfer function) based on the two responses. The main difference of the proposed system from the conventional fuzzy-neural network is that the fuzzy output is not taken as the input for the neural system in any layer. Rather, they are treated separately for the ease of computation. An additional module of cost minimization is introduced so as to minimize the RMSE generated by the architecture. The dataset used for simulation and comparison is Boston housing data. The RMSE is compared with all the existing method to study the efficiency of the proposed algorithm.

3. Methodology of Designing Composite Architecture

This section starts with a brief introduction of neural network construction and the associated training algorithm. We will also discuss some aspects of fuzzy system which will be required for our current problem formulation. The next subsection contains the two proposed method.

3.1. Maximum likelihood approach to neural network

The present architecture is supposed to handle the continuous input feature space through a feed forward neural network. Once the processing for both the continuous and discrete feature space is finished, the controller will try to optimize the maximum likelihood weights and the bias derived from a likelihood cost function.

Let $M$ be the output of $k$ independent data processing system. According to the theory of ensemble of network systems, $M = \Sigma w_k z_i$. The problem is to determine the weight vectors in such a way that the errors are minimal. There are three approaches of likelihood cost function named “bagging”, "balancing" and “bumping”. The main objective to use maximum likelihood cost function is to improve the overall performance of the ensemble members. A brief outline of the approach is highlighted in the subsequent paragraph.

The network architecture can be thought to be a prototype of the classical regression model. The variance of the error is equal to $W^T \sigma W$ under the constraint $\Sigma w_k = 1$ where $\sigma$ is the covariance matrix. It is important to note that the weight may not necessarily be positive. The next step is to search for function which yields an optimal weight. The concept of Bayesian analysis thus results in Maximum Likelihood Cost function ($E_{ml}$) in terms of error.

$$E_{ml} = \frac{1}{2} \left[ N \log(W^T \sigma W) + \frac{1}{W^T \sigma W} \sum e^2 \right]$$ (1)
The above objective function is to be minimized under the constraint that \( \sum w_k = 1 \) where \( \varepsilon^2 \) is the squared error term which can be expressed as \( (\sum w_k z - y)^2 \).

The current controller initially processes the data through feed forward system followed by optimized maximum likelihood cost function to get the least RMSE.

3.2. Modelling technique for discrete input features: ANFIS

Categorical data is processed using fuzzy inference system (FIS) in our proposed method. Fuzzy logic is capable of modeling non-linear arbitrary relationships and is highly tolerant of imprecise data in real life situation [17-20]. Zadeh proposed the concept of linguistic variables as an alternative approach to model human thinking in terms of fuzzy sets in place of crisp numbers [21]. Within fuzzy logic, neuro-fuzzy systems play a particularly important role in the induction of rules from observations. The prediction process can be made much more precise by using this effective method, developed by Jang and called ANFIS (Adaptive Neuro-Fuzzy Inference System), which combines elements of experts’ prior knowledge from FIS and further refines the results of FIS through the artificial learning process to develop the final relationship [22-24]. Ultimately, it constructs a FIS whose membership function (MF) parameters are adjusted using certain learning algorithm according to a chosen error criterion. For large data set, fine-tuning of MF parameters is recommended, since the human-determined MFs are seldom optimal in terms of reproducing desired outputs. So, we have used only backward pass for gradient descent method to update premise parameters with the help of error signals where consequent parameters are kept fixed. This type of modeling yields best results if the training data presented to the ANFIS for training membership function parameters is fully representative of the features of the data that the trained FIS is intended to model. The following subsection presents the algorithmic steps for processing categorical data in an adaptive neuro-fuzzy inference system.

### Steps for processing categorical data

1. Start the process with fuzzification of inputs.
2. Choose appropriate membership functions with initial parameters.
3. Determine the output membership functions.
4. Determine the If-Then rules depending on the subject knowledge.
5. Defuzzify the output based on centroid of area method (COA).
6. Train the parameters of the membership functions using ANFIS.

3.3. Composite fuzzy-neural architecture

We now present the block diagram of the composite architecture of the proposed fuzzy-neural system in Fig. 1. The algorithm of processing mixed variables through this model for developing input-output relationship is as follows.
Step 1. Specify the number of categorical and continuous variables
Step 2. Specify the number of sub-categories (linguistic terms) for each of the categorical variables
Step 3. Process the continuous inputs by neural network architecture (see Sec. 3.1)
Step 4. Train the neural network with the help of gradient descent algorithm
Step 5. Optimize the architecture from the $R^2$ and MSE considering training and testing data.
Step 6. Process the categorical inputs by fuzzy inference system (FIS). (see Sec. 3.2)
Step 7. Train the parameters of the membership functions using ANFIS.
Step 8. Defuzzify the output obtained from the ANFIS.
Step 9. Create a diagonal matrix with the elements of the defuzzified output.
Step 10. Perform a matrix multiplication operation of two matrixes, one obtained from the neural systems and one obtained from step 9.
Step 11. The crisp output from the fuzzy system and the continuous output from the neural network architecture is then put together in a linear form with known estimates of regression coefficients (which essentially is the fuzzy output) and the unbiased estimate of the intercept (calculated from Likelihood cost function).

Step 12. Obtain the final response and compare with the target value to obtain the MSE.

Difference with other approaches

The novelty in this approach lies in the design of the controller. Neuro Fuzzy Hybridizations are broadly done in two different ways. Fuzzy-neural network or FNN considers the input signals/weight matrix and output space to be fuzzy or a set of membership to fuzzy set [25]. Neural-Fuzzy System or NFS identifies the fuzzy rules based on different learning algorithms. Incorporation of Fuzzy (in the form fuzzy sets) in ANN increases the efficiency of the system by a knowledge mechanism. But the encoded fuzzy rules when feed into the neural network with the mixed input space could affect the efficiency of processing the continuous
feature space. Thus the knowledge-based network with a separate mechanism of handling the fuzzy subsets or a set of membership values to fuzzy sets has been optimally designed. The error minimization mechanism considers inclusion of maximum likelihood cost function method which increases the cost function by minimizing the weighted squared error term. As this error optimization problem assumes the probability density distribution of a particular observation indicating likelihood of measurement to come from same feature space, the accuracy of the model increases significantly compared to other methods.

4. About the Database

To verify the performance of the model proposed here, we have considered a standard Boston-housing data originally published by Harrison and Rubinfeld [26]. This dataset contains information collected by the US Census Service concerning housing in the area of Boston Massachusetts. It was obtained from the StatLib archive (http://lib.stat.cmu.edu/datasets/boston). The dataset has 506 instances. The total number of input variables in this dataset is 13 out of which 2 are categorical (CHAS and RAD) in nature. The database has a single continuous output variable. The relevant information of the features are as follows.

a) CRIM: per capita crime rate by town
b) ZN: proportion of residential land zoned for lots over 25,000 sq.ft.
c) INDUS: proportion of non-retail business acres per town
d) CHAS [categorical]: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
e) NOX: nitric oxides concentration (parts per 10 million)
f) RM: average number of rooms per dwelling
g) AGE: proportion of owner-occupied units built prior to 1940
h) DIS: weighted distances to five Boston employment centres
i) RAD [categorical]: index of accessibility to radial highways
j) TAX: full-value property-tax rate per $10,000
k) PTRATIO: pupil-teacher ratio by town
l) B: 1000(B_k - 0.63)^2 where B_k is the proportion of blacks by town
m) LSTAT: % lower status of the population
n) MEDV [response]: Median value of owner-occupied homes in $1000's.

5. Results and Discussion

The algorithm is tested on the above-mentioned data set and 106 cases have been used for testing purpose to check the generalization capability of the composite architecture.

5.1. Training of the neural network architecture

The feed-forward multilayer perceptron network consists of 11 nodes (representing 11 continuous input features) in the input layer and 1 node in the output layer representing the output variable MEDV. The number of hidden nodes in a single
hidden layer was varied from 10 to 30 along with different standard transfer functions and gradient based training algorithms. For each of these combinations of transfer functions and training algorithms, a particular architecture was trained with 300 epochs, repeated 20 times and compared its performance based on the statistics (mean, standard deviation, CV) of MSE/RMSE values on the output data set. A random combination of the training vectors is used for each epoch for the purpose of modeling a stable system. This process is continued by varying hidden nodes to see the effectiveness of a single hidden layer. The learning rate and performance goal were kept fixed at 0.05 and $10^{-5}$ respectively for all the combinations. It is observed over different trial runs and computing overall consistency of MSE/RMSE that the hidden layer of 26 nodes produces minimum RMSE (see Fig. 2). It is also found that, tansigmoid transfer function and scaled conjugate Gradient (SCG) training algorithm result in minimum RMSE (see Table 1).

![Fig. 2. Pattern of RMSE over Hidden Nodes in ANN Architecture.](image)

### Table 1. MSE for Different Transfer Functions and Training Conditions (# of hidden nodes: 26)

<table>
<thead>
<tr>
<th>Transfer Functions</th>
<th>Training Algorithm</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logsigmoid</td>
<td>SCG</td>
<td>30.5986</td>
</tr>
<tr>
<td>Logsigmoid</td>
<td>LM</td>
<td>31.4661</td>
</tr>
<tr>
<td>Tansigmoid</td>
<td>SCG</td>
<td>19.5746</td>
</tr>
<tr>
<td>Tansigmoid</td>
<td>GDM</td>
<td>21.6431</td>
</tr>
<tr>
<td>Tansigmoid</td>
<td>LM</td>
<td>30.2953</td>
</tr>
<tr>
<td>Tansigmoid</td>
<td>GD</td>
<td>22.4198</td>
</tr>
<tr>
<td>Tansigmoid</td>
<td>GDA</td>
<td>31.3781</td>
</tr>
</tbody>
</table>

5.2. Training using ANFIS

First, we have hypothesized a parameterized model structure (relating inputs to membership functions to rules to outputs to membership functions, and so on). The membership functions we have used are of Gaussian type and for the Charles River dummy variable we have used two MFs (fuzzy sets) whereas for index of accessibility to radial highways we have used 9 MFs. A few results regarding the training of the parameters for the Charles River Dummy Variable is shown in Fig. 3.

A few results regarding the training of the parameters for the Index of Accessibility to radial Highways is shown in Fig. 4.
5.3. Estimation of bias using log-likelihood cost function

As stated in the methodology we will use a linear filter to obtain the predicted response values. For this purpose we have to obtain a suitable bias \( b \) that will lead to actual prediction. We have used the crisp values of fuzzy system outputs as the weight matrix and the output of the above-mentioned architecture as the input space for the linear filter. The bias can be estimated at the last layer using the log-likelihood cost function.

From the fuzzy response we are fixing the weight matrix of the linear filter. Let it be \( w \) and the response from the neural network architecture is \( y \). Now, from the principle of log-likelihood cost function we can say that,

\[
\text{Cost-function } (E_{LL}) = -\sum_i \{y_i \ln(\mu_i) + (1-\mu_i)\ln(1-\mu_i)\},
\]

where, \( E_{LL} \) is the actual output and \( y_i \) is the output from neural network architecture. Now, substituting \( \mu_i = wy_i + b \), we get,

\[
E_{LL} = -\sum_i \{y_i \ln(wy_i + b) + (1- wy_i)\ln(1- (wy_i + b))\}
\]

(2)

So, with the help of least square method, we can get an estimate of bias \( b \) from Eq. (2) as
\[
\frac{\partial E_{LL}}{\partial b} = -\sum \frac{y_i}{\ln(wy_i + b)} + \sum \frac{(1 - y_i)}{\ln(1 - wy_i - b)} = 0
\]  

(3)

We have repeatedly performed the operations in neural network to get a distribution of the mean square error for the current architecture, the corresponding frequency distribution of which is shown in Fig. 5. The average MSE and RMSE are estimated as 19.40332 and 4.40492 respectively. The regression plot (see Fig. 6) shows an $R^2$ value near 0.93 which is quite acceptable considering such a real life data set. The predicted results are well within the confidence band except a few outliers which around 0.39% for the Boston-housing data set. A comparative performance of the proposed model along with two other methods is done with respect to RMSE on the same dataset. As observed from Table 2, the composite architecture proposed here performs better for a very well-known dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brower’s Separation Methods</td>
<td>8.1</td>
</tr>
<tr>
<td>Conventional Method with $\alpha$-numeric coding</td>
<td>9.1</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>4.40492</td>
</tr>
</tbody>
</table>

Table 2. Comparative Study.

The maximum likelihood cost function aims at optimization of errors/noise treating it as a classical weighted assignment problem. It has been stated in the preceding sections that the method minimizes the normalized squared correspondence errors and looks for the optimal estimates of the bias in terms of weights. Classical machine learning approaches implicate the error minimization algorithms in image classification techniques for noise reductions for accurate classification of horizontal and vertical sequence. But the current approach assuming the distribution of the variables apply the concept of likelihood for correct estimation of bias that appear to be a principal driver influencing the mean square error of the networks. The usage of such techniques implies a drastic reduction in RMSE than the available methods.

A most common approach to handle the categorical feature space is by Adaptive Neuro-fuzzy system. The proposed architecture takes into consideration the simultaneous functionality of ANN and fuzzy system. The important part of
this controller optimizes the error function and rapidly enhances the accuracy rate. Error optimization has been done in a different way and quite a lot of researches have been carried out so far in order to improve the performance rate. The maximum likelihood cost function is generally applied in a series of neural network architecture having different errors for each of the networks. Linear ensemble of regression neural networks usually uses Bates-Granger method which often suffers severely from numerical instability problem. The current controller thus avoids the matrix inversion problems involved in Bates-Granger method and includes likelihood methods (simulated annealing) to optimize the error rate.

6. Concluding Remarks

Both Brower’s separation method and alpha-numeric coding do not ensure the interaction effect of multiple categorical variables which is major drawback of both the methods. In context of real-life predictive problems, complex interactions among categorical variables play a significant role. The existing method can handle this issue effectively and on the other hand, additional cost minimization module to minimize error function drastically reduces RMSE. One disadvantage of the existing methodology is the use of rule based fuzzy system. Improper if-then rule might hamper the overall accuracy of the model. Also the existing methodology doesn’t ensure the interaction between continuous and categorical variable and a further extension to this approach may include fuzzy clustering and higher order TSK model to overcome the limitations of the current system. The work is already in progress.

References


