Neurocomputing 154 (2015) 225–229

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Short Communication

Neural network based class-conditional probability density function using kernel trick for supervised classifier

E.S. Gopi *, P. Palanisamy

Department of Electronics and Communication Engineering, National Institute of Technology Trichy, Trichy-620015 Tamil Nadu, India

ARTICLE INFO

Article history:
Received 24 December 2013
Received in revised form
28 November 2014
Accepted 30 November 2014
Communicated by S. Mitra
Available online 6 December 2014

Keywords:
Back propagation neural network
Bayes classifier
Kernel trick
Kernel-Linear discriminant analysis
Higher dimensional space

ABSTRACT

The practical limitation of the Bayes classifier used in pattern recognition is computing the class-conditional probability density function (pdf) of the vectors belonging to the corresponding classes. In this paper, a neural network based approach is proposed to model the class-conditional pdf, which can be further used in supervised classifier (e.g. Bayes classifier). It is also suggested to use kernel version (using kernel trick) of the proposed approach to obtain the class-conditional pdf of the corresponding training vectors in the higher dimensional space. This is used for better class separation and hence better classification rate is achieved. The performance of the proposed technique is validated by using the synthetic data and the real data. The simulation results show that the proposed technique on synthetic data and the real data performs well (in terms of classification accuracy) when compared with the classical Fisher's Linear Discriminant Analysis (LDA) and Gaussian based Kernel-LDA.

1. Introduction

In pattern recognition, Bayes classifier is often used as the standard bench mark [1]. We use the upper case bold letter to represent the matrix and the lower case bold letters to represent the vector. Let $f_i(\mathbf{x})$ be the class-conditional probability density function (pdf) of the continuous random vector $\mathbf{x} \epsilon \Re^n$, conditioned on the class variable $\mathbf{i}$, $\forall i = 1, 2, \ldots r$, where r is the number of classes. The vector $\mathbf{u}\ \epsilon \Re^n$ is classified as the one belonging to the class j if $f_i(\mathbf{u})p_i$ $(\forall i = 1, 2, \ldots r)$ is maximum for $i{=}j$, where $p(i)$ is the probability of the ith cluster.

In practice, the class-conditional pdf of the training data is rarely known. For the past few decades, attempts were made to obtain the parametric value of class-conditional pdf either directly or indirectly using Artificial Neural Network. The interpretation of training criteria and structure of the network for pattern recognition applications were explored in [2]. In [3], the Artificial Neural Network was used to estimate conditional variance of the random variable assuming that the conditional probability density function of the random variable as Gaussian. In [4], stochastic neural network for pattern recognition was used to estimate missing data for pattern recognition. Expectation–Maximization algorithm was used to estimate the missing data. Conditional probability density function of the random variable $\mathbf{x}$ given $\mathbf{y}$ was obtained using the sigmoidal method and were compared with Gaussian mixture model in [5]. In [6], posterior probability was estimated by constrained conditional density function using logistic and soft-max architecture. In [7], class-conditional density function was estimated using shared kernel function (multivariate Gaussian density function). There were mostly used for unsupervised learning. Also no attempts were made to obtain class-conditional pdf of the data after mapping to the higher dimensional space using kernel trick.

Kernel trick is the technique of using the kernel function as the replacement of inner-product of two vectors in the mapped higher dimensional space. The advantage is that, we can optimize the higher dimensional space without the explicit mapping function to map the arbitrary vector from the feature dimensional space to the higher dimensional space. Kernel trick was first proposed for fisher discriminant analysis in the higher dimensional space [8]. It was further used in support vector machine (SVM). Recently, usage of the kernel trick has been explored for various applications as mentioned below. In [9], modification of kernel discriminant analysis for higher dimensional data was proposed. The Kernel-based non-linear discriminant analysis was proposed in [10]. Multiple kernel trick for fisher discriminant analysis was used in [11]. An attempt was made on kernel based adaptive principal component analysis [12] for dimensionality reduction. The study on kernel least mean square algorithm was done in [13,14]. In this paper, the Kernel trick to compute the class-conditional pdf of the mapped vectors in the higher dimensional space is proposed. It is modeled as the Back Propagation Neural Network (BPNN) [1] with sigmoidal function.

The class-conditional pdf is usually obtained using the prior probabilistic model (e.g. Gaussian Mixture Model). Estimating the unknown parameters of this model is the difficult task in

* Corresponding author. Tel.: +91 9500423313, +91 4312503314; fax: +91 4312500133.
E-mail addresses: esgopi@nitt.edu (E.S. Gopi), palan@nitt.edu (P. Palanisamy).

http://dx.doi.org/10.1016/j.neucom.2014.11.070
0925-2312/

terms of computational complexity and time. In this paper, we propose a new approach to obtain the class-conditional pdf in both the feature dimensional space and the mapped higher dimensional space (for better class separation) using the kernel trick. In the proposed approach, BPNN is used as the model for estimating class-conditional pdf of the training data. The proposed technique is useful in supervised classifier that uses class-conditional pdf (e.g. Bayes classifier).

2. Idea behind the proposed technique

Let $f_i(\mathbf{u})$ is the class-conditional pdf of the i^{th} class. Given the training set, by intuition, we understand that the probability density function of the arbitrary vector $\mathbf{u}$ belongs to training set, belonging to the particular class j which is defined as follows (refer Fig. 1).

$$f_j(\mathbf{u}) \underline{\Delta} \alpha \sum_{i=1, i \neq j}^{r} \frac{d(\mathbf{u}, \mathbf{c_i})}{d(\mathbf{u}, \mathbf{c_j})} \tag{1}$$

where, $d(\mathbf{u}, \mathbf{c_i})$ is the square of the euclidean distance between the vector $\mathbf{u}$ and the centroid $\mathbf{c_i}$ of the ith class and α is the normalization constant.

The set of training vectors and the corresponding class-conditional pdf values computed using the proposed empirical method (refer (1)) are collected. The class-conditional pdf for every class is modeled using the non-linear sigmoidal BPNN model [1]. The number of neurons in the input layer is equal to the number of attributes in the feature dimensional space and the number of neurons in the output layer is 1. Once class-conditional pdf is obtained, Bayes classifier (described in Section 1) can be formulated to classify the uniformly distributed unknown random vector $\mathbf{u} \in \Re^n$ that is uniformly distributed. The elements of the vector $\mathbf{u}$ are nor;malized $(-1 \text{ to } 1)$ before given as the input to the BPNN for both training and testing.

If the metric d in (1) is measured in the higher dimensional space, the corresponding trained BPNN is modeled as the class-conditional pdf (obtained using the kernel trick) in the higher dimensional space. The euclidean distance measure d in the higher dimensional space using the kernel trick is described in Section 2.1.

2.1. *Euclidean distance d in the higher dimensional space computed using the kernel function*

Let $\mathbf{u_{ij}}$ be the i^{th} vector in the j^{th} class in the lower dimensional space (*LDS*) and $\mathbf{v}_{ij} = \phi(\mathbf{u_{ij}})$ be the i^{th} vector in the j^{th} class in the higher dimensional space (*HDS*), where ϕ is the map from the (*LDS*) to the (*HDS*). Let $d(\mathbf{v}_{ij}, \mathbf{s_q})$, be the euclidean-distance between the vector $\mathbf{v_{ij}}$ to the centroid of the q^{th} class $(\mathbf{s_q})$ in the *HDS* and n_q be the number of vectors in the q^{th} class. The euclidean distance $d(\mathbf{v}_{ij}, \mathbf{s_q})$ in the *HDS* is computed [15] as follows:

$$d(\mathbf{v_{ij}}, \mathbf{s_q}) = \left[\mathbf{v_{ij}} - \left(\frac{1}{n_q}\right)\Sigma_{k=1}^{n_q}\mathbf{v_{ij}}\right]^T \left[\mathbf{v_{ij}} - \left(\frac{1}{n_q}\right)\Sigma_{k=1}^{n_q}\mathbf{v_{ij}}\right] \tag{2}$$

$$= k(\mathbf{u_{ij}}, \mathbf{u_{ij}}) - \left(\frac{2}{n_q}\right)\Sigma_{k=1}^{n_q} k(\mathbf{u_{ij}}, \mathbf{u_{kq}}) + \left(\frac{1}{n_q^2}\right)\Sigma_{k=1}^{n_q}\Sigma_{l=1}^{n_q} k(\mathbf{u_{kq}}, \mathbf{u_{lq}}) \tag{3}$$

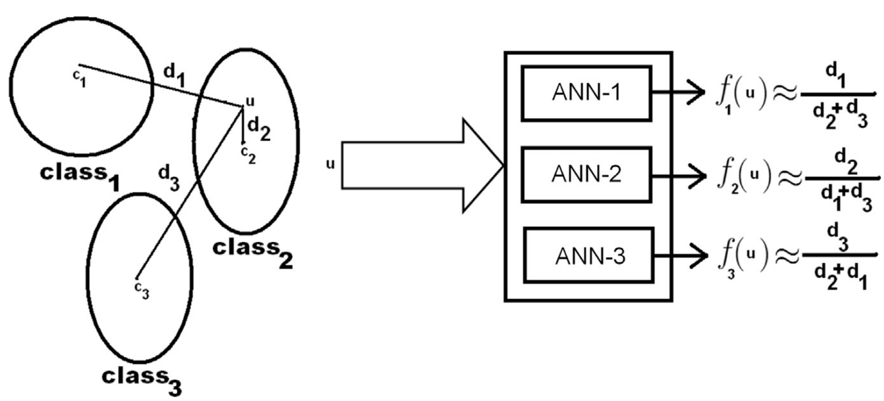

Fig. 1. Illustration of the proposed technique to compute class-conditional pdf in the higher dimensional space for three classes.

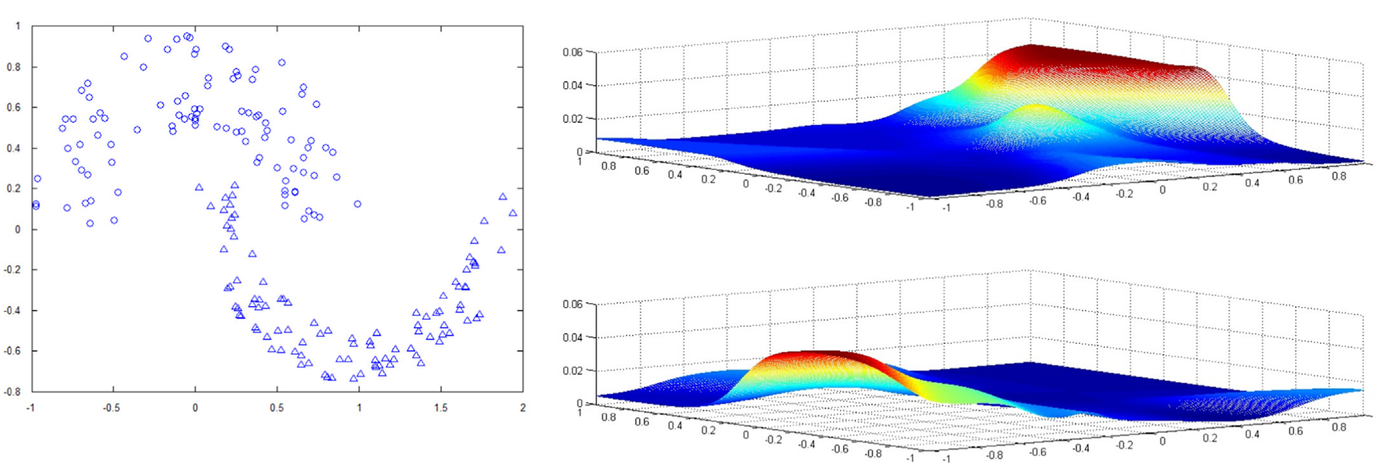

Fig. 2. Class conditional pdf obtained using the trained ANN with kernel trick for 'Half ring' synthetic data.