Information Sciences 274 (2014) 95–107

Contents lists available at ScienceDirect

Information Sciences

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

Evolutionary combination of kernels for nonlinear feature transformation

Behzad Zamani [a,*], Ahmad Akbari [a], Babak Nasersharif [a,b]

[a] Audio & Speech Processing Lab, Computer Engineering Department, Iran University of Science & Technology, Tehran, Iran
[b] Electrical and Computer Engineering Department, K.N. Toosi University of Technology, Tehran, Iran

ARTICLE INFO

Article history:
Received 23 May 2012
Received in revised form 27 November 2013
Accepted 9 February 2014
Available online 5 March 2014

Keywords:
Kernel principal component analysis (KPCA)
Kernel linear discriminant analysis (KLDA)
Genetic algorithm
Genetic programming
Kernel combination

ABSTRACT

The performance of kernel-based feature transformation methods depends on the choice of kernel function and its parameters. In addition, most of these methods do not consider the classification information and error for the mapping features. In this paper, we propose to determine a kernel function for kernel principal components analysis (KPCA) and kernel linear discriminant analysis (KLDA), considering the classification information. To this end, we combine the conventional kernel functions using genetic algorithm and genetic programming in linear and non-linear forms, respectively. We use the classification error and the mutual information between features and classes in the kernel feature space as evolutionary fitness functions. The proposed methods are evaluated on the basis of the University of California Irvine (UCI) datasets and Aurora2 speech database. We evaluate the methods using clustering validity indices and classification accuracy. The experimental results demonstrate that KPCA using a nonlinear combination of kernels based on genetic programming and the classification error fitness function outperforms conventional KPCA using Gaussian kernel and also KPCA using linear combination of kernels.

1. Introduction

The feature extraction is a crucial step in the pattern recognition process which greatly affects the performance of pattern recognition systems. In this step, the useful discriminative information should be extracted from the pattern in such a way that a classifier can recognize different patterns. Several methods have been proposed to make the most discriminative features and also robust to noise. A group of these methods are based on the feature mapping using linear or non-linear transformation approaches.

Some well-known examples of the linear transformation methods are Principal Component Analysis (PCA) [8], Linear Discriminant Analysis (LDA) [8] and its family including Heteroscedastic LDA (HLDA) [22], pairwise LDA (PLDA) [23], null space based linear discriminant analysis (NLDA) [24] and evolutionary based LDA [28]. These methods project the original feature vectors into a new feature space via a linear transformation based on different mapping criteria. For example, PCA finds the large variance in the original feature space for the mapping data. On the other hand, LDA maximizes the ratio of the between-class variation and the within-class variation for projecting features into a subspace. However, the drawback of these transformations is that their mapping criteria are different with the classifier error criterion [37,40] and can

* Corresponding author. Address: Department of Computer Engineering, Iran University of Science and Technology, University Road, Hengam Street, Resalat Square, Tehran, Iran. Tel.: +98 (21) 77491192, mobile: +98 (913) 3839249.

E-mail address: bzamani@iust.ac.ir (B. Zamani).

http://dx.doi.org/10.1016/j.ins.2014.02.140
0020-0255/

potentially corrupt the classifier performance. There are several ways to overcome this drawback [19]. A solution is proposed by the authors in [40] to improve these transformations based on minimizing Hidden Markov Model (HMM) classification error.

In the nonlinear feature transformation approaches, based on the Cover's theorem, if the input feature space is mapped nonlinearly to a high-dimensional space, non-separable patterns can become linearly separable in the transformed space [18]. This mapping is usually done via a kernel-based transformation such as kernel PCA (KPCA) [18,35], kernel LDA (KLDA) [18,27] and kernel class-wise locality preserving projection (KCLPP) [20] or a neural network likes as nonlinear PCA (NLPCA) [25].

However, the performance of the kernel-based transformation methods and the kernel-based classification methods depend on the choice of kernel function and its parameters. So, there are many approaches to the determination of a suitable kernel function especially for kernel-based classifiers. The best known methods are kernel estimation [36], kernel parameters optimization [21,43], multiple kernel learning [2,11,12,42], determining kernel-based on mutual information [5] and kernel optimization using evolutionary algorithms and convex optimization methods [15,16]. As another method, genetic programming (GP) is used to improve the support vector machine (SVM) kernel function for higher classification accuracy [10,14,34]. In [14], the authors find a near optimal kernel function using strongly typed genetic programming (STGP [29]). Authors propose a GP based kernel construction method for the relevance vector machine (RVM) [3]. In [26,30,31], authors obtain SVM kernel functions and their parameters using genetic algorithm where the fitness function is SVM classification error.

In this paper, we propose two methods to obtain more suitable kernel functions for kernel-based feature transformation methods (KPCA and KLDA) with attention to this reality that their mapping criteria do not consider the classifier error criterion. To this end, we consider the classification error and also the mutual information between features and classes in the kernel feature space as criteria for determining the kernel function. For this purpose, we determine our new kernel functions based on linear combination and also non-linear combination of basic kernel functions. Linear combination is performed using genetic algorithm. On the other hand, the genetic programming is used for non-linear combination of kernel functions. The classification error and the mutual information between features and classes are used as fitness functions in the mentioned genetic algorithm and genetic programming.

The rest of the paper is organized as follows. In Section 2, we briefly explain the kernel idea, KPCA and KLDA. Section 3 explains kernel combination and our method for combining linear and nonlinear kernels using genetic algorithm and genetic programming, respectively. Section 4 contains our experiments and results. Finally, we give our conclusion in Section 5.

2. Kernel-based feature mapping

Kernel-based nonlinear feature transformation and classification are a new research area in the machine learning. Besides using kernel-based classifiers such as the well-known SVM, kernel-based-methods are used to transform and map the feature space. KPCA and KLDA are well-known examples of kernel-based feature transformation methods.

Kernel functions allow us to compute the dot products in the mapped higher dimensional spaces without explicit mapping in these spaces. Based on Mercer's theorem, if k is a kernel function, then a dot product space F exists with a mapping Φ: $X \to F$ such that [18]:

$$\forall x, z \in X \quad k(x,z) = \phi(x)\cdot\phi(z) = \langle\phi(x),\phi(z)\rangle \tag{1}$$

where X is the original feature space, F is named as the kernel feature space, Φ is the feature map and dot indicates the dot product. When Φ is the identity, the function k is symmetric, continuous and positive-definite, so it constitutes a proper Mercer kernel [17]. Polynomial kernel and Gaussian radial basis function (RBF) kernel are examples of Mercer kernel defined as Eqs. (2) and (3), respectively [18,34].

$$k(x,z) = (\langle x,z\rangle + 1)^d, \quad d \in N \tag{2}$$

$$k(x,z) = \exp\left(\frac{-\|x-z\|^2}{\gamma}\right), \quad \gamma \in R^+ \tag{3}$$

2.1. Kernel principal component analysis (KPCA)

The KPCA technique applies the kernel function to PCA in order to obtain the representation of PCA in a higher dimensional space. Consider the mapped data as $\varphi(X_1),\ldots,\varphi(X_N)$. The centered points used in the algorithm evaluation are expressed as [35]:

$$\hat{\varphi}(X_k) = \varphi(X_k) - \frac{1}{N}\sum_{j=1}^{N}\varphi(X_j) \tag{4}$$

and the covariance matrix is given as