ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

ICITCSA 2017

Pioneer College of Arts and Science, Coimbatore

Vol. 6, Special Issue 1, January 2017

Performance Tuning with Different Kernel Function Hyperplane Analysis for Optimal Recognition Rate

P. Rukmanidevi[1], V. Kalaivani[2]

Assistant Professor, Department of Computer Applications, Pioneer College of Arts & Science, Coimbatore, India[1,2]

Abstract- Kernel principal component analysis is presented for kernel feature selection and High dimensional feature extraction to show kernel adaptations for nonlinear features selection of medical image data sets (MIDS). The proposed kernel principal component analysis extracts the salient features from a sample of unclassified patterns by use of a kernel. The kernel principal component analysis iteratively constructs a linear subspace of a high-dimensional feature space by exploiting a variance condition for the nonlinearly transformed samples. The resulting kernel subspace can be first chosen and then be processed for composite kernel subspace through the efficient combination representations used for further reconstruction and classification based on support vector machine.

Keywords: Support Vector Machine, Principal component analysis, data-dependent kernel, nonlinear subspace.

I.INTRODUCTION

The kernel principal component analysis extracts the salient features from a trial of unspecified patterns by use of a various kernel functions. Various cancer data sets are trained. It is very difficult to explicitly center the data in the feature space. But, in the existing work they have used various kernel methods to identify the best kernel to calculate linear composite kernel space. For doing this each kernel methods has been exploited which makes time usage higher and also finding best kernel is also somewhat not possible. To overcome this problem, the center kernel matrix can be accessed. The proposed kernel principal component analysis depends only on the kernel matrix, it makes easier to compute the centered kernel matrix in terms of the non-centered kernel alone and no features need to be accessed. Mainly support vector machine is used for classification in this paper the focus is shown to multi class support vector machine to classify the given datasets. SVM aims to find a hyper plane that linearly separates data points belongs to different classes and also least likely to over fit the training data. Multi class support vector machine has taken to assign labels to instances by using support vector machines, where the labels are drawn from a finite set of several elements in the datasets. Mapping is done so that the separate categories are divided by a clear gap that is as wide as possible. SVMs can efficiently perform a non-linear classification using kernel method, which implicitly mapping their inputs into high-dimensional feature spaces which makes dimensionality reduction. Numerical experiments have been shown with kernel principal component analysis with existing results. It shows proposed work generates more efficient and an effective feature representation, and has yielded a better classification performance for the proposed composite kernel subspace using a multi class SVM classifier.

II. EIGENVECTORS AND EIGENVALUES

When we get a set of data points, like the triangles above, we can deconstruct the set into eigenvectors and eigenvalues above are exist in pairs every eigenvector has a corresponding eigenvalue. An eigenvector is a direction in the example above the direction of the line (vertical, horizontal, 45 degrees etc.). An eigenvalue is a number, how much variance there is in the data in that direction,the number telling us how extend out the data is on the line. The eigenvector with the highest eigenvalue is therefore the principal component.

Okay, so even though in the last example I could point my line in any path, it turns out there are not many eigenvectors/values in a data set. In fact the amount of eigenvectors/values that exist equals the number of dimensions the data set. Measuring age and hours on the internet. There are 2 variables, it's a 2 dimensional data set, and therefore there are 2 eigenvectors/values. If measuring age, hours on internet and hours on mobile phone there's 3 variables, 3-D data set, so 3 eigenvectors/values. The reason for this is that eigenvectors put the data into a fresh set of dimensions, and these new dimensions have to be equal to the original amount of dimensions.

III.KERNEL PRINCIPAL COMPONENT ANALYSIS

The starting point is a random vector $x \varepsilon R^n$ with N observations x_i, $i \varepsilon [1,\ldots N]$. In PCA, data are first centered $x \Leftarrow (x - E\{x\}$. Then PCA diagonalizes the covariance matrix Cx:

$$C_x = \frac{1}{N}\sum_{i=1}^{N} x_i x_i^T$$

This problem leads to solve the eigenvalue equation

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

ICITCSA 2017

Pioneer College of Arts and Science, Coimbatore

Vol. 6, Special Issue 1, January 2017

$\Lambda v = C_x v \quad ||v|| = 1$

where $\lambda \geq 0 \in$ are eigenvalues and v $\in R^n$ are eigenvectors. The projection on the eigenvector vk is done by:

$$x_{pc}^k = v^k . x$$

Now, suppose we first map the data onto another dot product space H:

$$\varphi: R^n \to H \qquad x \to \varphi(x)$$

Here, Φ could be a nonlinear function and H could have infinite dimensionality. PCA can be perform in H with the same procedure as previously: the data is centered and the covariance matrix is defined as:

$$C_{\varphi(x)} = \frac{1}{N}\sum_{i=1}^{N} \varphi(x_i)\varphi(x_i)^T$$

Similarly to PCA, one as to solve:

$$\lambda v_\varphi = C_{\varphi(x)} v_\varphi = \frac{1}{N}\sum_{i=1}^{N} \varphi(x_i)\varphi(x_i)^T v_\varphi$$
$$= \frac{1}{N}\sum_{i=1}^{N} \varphi(x_i).v_\varphi)\ \varphi(x_i)$$

From (6), it is clear that v $_\Phi$ is lying in the span of $\varphi(x_1) \ldots . \varphi(x_N)$ thus each eigenvector can be written as:

$$v_\varphi = \sum_{i=1}^{N} \propto_i \varphi(x_i)$$

By multiplying (6) with $\Phi(x_k$)from the left and substituting (7) into it, we get:

$$\lambda \sum_{i=1}^{N} \propto_i \varphi(x_k).\varphi(x_i)) =$$
$$\frac{1}{N}\sum_{i=1}^{N} \propto_i (\varphi(x_k).\sum_{j=1}^{N} \varphi(x_j).\varphi(x_i))\ \varphi(x_j))$$

for k $\in$ [1,N]

Defining the N×N Gram matrix K by

$$K_{ij} := \varphi(x_i))\ \varphi(x_j))$$

the above equation turns to:

$$\lambda K \propto = \frac{1}{N} k^2 \propto$$

Where $\propto = (\propto_1, \ldots \ldots \propto_N)^T$ .The solution of (9) is found by solving the eigen value problem:

$$N\lambda \propto = K \propto$$

for nonzero eigenvalues. Clearly, all solutions of (10) satisfy. However, it does not give all the solutions, eigenvector associate to zero eigenvalue is solution of (9) which is not a solution of (10). But, it can be shown that these solutions lead to null expansion of (7) and thus are irrelevant for the considered problem. Finally, to solve $C_{\Phi(x)}$'s eigenvalue equation is equivalent to solve K's eigenvalue equation. The unitary norm condition from (2) is translated in H into $\lambda_K(\alpha^k.\alpha^k) = 1$ The projection in H is simply done by:

$$\varphi(x)_{kpc}^k = v_\varphi^k.\varphi(x) = \sum_{i=1}^{N} \alpha_i^k \varphi(x_i)\varphi(x))$$

However, compute PCA in H has a high computational cost. Using kernel trick, it is possible to work implicitly in H while all computations is finished in the input space. Using kernel function, the dot product in feature space is reduced to a (possibly nonlinear) function in input space.

$$\varphi(x_i).\varphi(x_j) = k(x_i, x_j)$$

The kernel function has to satisfy the Mercer's theorem to ensure that it is possible to create a mapping into a space where K acts as a dot product. The polynomial kernel and the Gaussian kernel are ones of the most used kernel:

$$k_{poly}(x_i, x_j) = (x_i.x_j + r)^d$$
$$k_{gauss}(x_i, x_j) = exp(-\gamma||x_i - x_j||^2$$

When builds with kernel functions, Gram matrix is also called as Kernel Matrix. Finally, the KPCA is done in the original space as follows:

1. Compute the Kernel Matrix:

$$k_{ij} = k(x_i, x_j)$$

2. Center K

$$K_c = K - 1_N K - K 1_N + 1_N K 1_N$$

Where 1_N is a N square matrix for which $(1_N)_{ij}$ = 1 /N , for all (i,j) in [1, . . . ,N].

3. Diagonalize K_c and normalize eigenvectors:

$$\lambda_k(\alpha^k.\alpha^k) = 1$$

4. Extract the k first principal components:

$$\varphi(x)_{kpc}^k = \sum_{i=1}^{N} \alpha_i^k \varphi(x_i).\varphi(x))$$

A. Training

Separating data into training and testing sets is an important part of evaluating data mining models. Typically, when you separate a data set into a training set, most of the data is used for training. Analysis Services randomly samples the data to help ensure that the testing and training sets are parallel. By using analogous data for training and testing, you can minimize the effects of data discrepancies and enhanced comprehend the characteristics of the model.

Following a model has been processed by using the training set; you test the model by building predictions next to the test set.

B. KPCA kernel function selection

Kernel selection is heavily dependent on the specific data set. For kernel-based learning algorithms, the key challenge lies in the selection of kernel parameters and

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

ICITCSA 2017

Pioneer College of Arts and Science, Coimbatore

Vol. 6, Special Issue 1, January 2017

regularization parameters. KPCA can be seen as an overview of PCA where nonlinearities in the data are taken into account by nonlinear mapping of the data, using a kernel function, into a higher dimensional space in which PCA is carried out. KPCA is utilized to first transform the original sample space to a nonlinear feature space via the appropriate kernel function, and then perform principal component analysis (PCA).

In n dimensional training samples of the given data, some of them are represented as class labels of the samples. We have developed a data dependent kernel to capture the relationship among the data in this classification task. The linear, polynomial and RBF kernels are selected.

C. Linear Kernel

The Linear kernel is the simplest kernel function. It is given by the inner product <x,y> plus an optional constant **c**. Kernel algorithms using a linear kernel are frequently equivalent to their non-kernel counterparts, i.e. KPCA with linear kernel is the same as standard PCA.

$$k(x, y) = x^T y + c$$

D.Polynomial Kernel

The Polynomial kernel is a non-stationary kernel. Polynomial kernels are well-matched for problems where all the training data is normalized.

$$k(x, y) = (\alpha x^T y + c)^d$$

Adjustable parameters are the slope **alpha**, the constant term **c** and the polynomial degree **d** and it is mainly exploited for performance analysis.

E. Radial Basis Function Kernel

The B-Spline kernel is defined on the interval [−1, 1]. It is given by the recursive formula:

$$k(x, y) = B_{2p+1}(x - y)$$

Where Pєn with

$$B_{i+1} := B_i \otimes B_0$$

$$k(x, y) = \prod_{p=1}^{d} B_{2n+1}(x_p - y_p)$$

Where x_+ is defined as the truncated power function:

$$x_+^d = \begin{cases} x^d & if\ x > 0 \\ 0, & otherwise \end{cases}$$

IV. IMPLEMENTATION

The performance measure has been evaluated for proposed SVM with existing Knn. It indicates that proposed work shows accurate classification results in selected medical data sets. KPCA with multi class svm has been exploited in order to show the superiority of feature selection, has applied to medical datasets comprising of colon data set, leukemia data set, lung data set and Prostate datasets. In classification stage multi class svm have been used. The classification accuracy, Performance for SVM and KNN, error rate and elapsed time has been evaluated in proposed method.

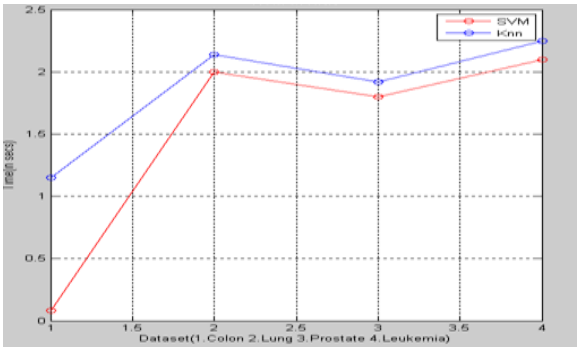

Fig.1 Performance measure for KPCA_SVM Vs KNN

Fig. 2 Performance measure for KPCA_SVM Vs KNN with error rate

The obtained results from KPCA with muti class svm with existing knn algorithms error rate of medical cancer dataset has been shown. This data set consisted of true-positive (TP) and falsepositive (FP) detections has been obtained with portion, data portion and size of each data set respectively. Elaspsed time for colon is 0.025968 seconds, Leukemia data set is 0.028987 seconds, Lung data set is 0.030070 seconds and Prostate data set is 0.030943 seconds.

KPCA_SVM VS KPCA_KNN In scents

KPCA_SVM VS KPCA_KNN (error rate %)

V. CONCLUSION & FUTURE DIRECTIONS

This paper describes KPCA with multiclass SVM, a faster and more efficient feature extraction of medical data sets. Numerical experiments have been shown with kernel principal component analysis with existing results. It shows proposed work generates more efficient and an effective feature representation, and has yielded a better classification performance for the selected kernel subspace using a multi class SVM algorithm. Best three linear, polynomial and RBF kernel function has chosen which is adapted to the data set, as well as extending this process of dimensionality reduction and feature extraction. The experimental result is tested with four data sets for evaluating the error rate, classification accuracy using multi class svm algorithm, and elapsed time has been calculated.

In future the attention can be shown to various data sets collection more reasonably and effectively and various measurement functions such as correlation, covariance can

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

ICITCSA 2017

Pioneer College of Arts and Science, Coimbatore

Vol. 6, Special Issue 1, January 2017

be exploited to measure the distinguishing between two classes.

REFERENCES

[1] M. Pechenizkiy, S. Puuronen, A. Tsymbal, "Feature Extraction For Classification In The Data Mining Process" International Journal "Information Theories & Applications" Vol.10.
[2] MaxWelling, "Kernel Principal Components Analysis".
[3] Bernhard Scholkopf, Alexander Smola and Klaus Robert Muller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem"
[4] Mathieu Fauvel , Jocelyn Chanussot , Jon Atli Benediktsson, "kernel principal component analysis for feature reduction in hyperspectrale images analysis".
[5] James T. Kwok, Brian Mak and Simon Ho, "Eigenvoice Speaker Adaptation via Composite Kernel PCA".
[6] Quan Wang, "Kernel Principal Component Analysis and its Applications in Face Recognition and Active Shape Models".
[7] Sebastian Mika, Bernhard Scholkopf, Alex Smola, Klaus-Robert Muller, Matthias Scholz, Gunnar Ratsch , "Kernel PCA and De-Noising in Feature Spaces".
[8] Klaus-Robert Müller, Sebastian Mika, Gunnar Rätsch, Koji Tsuda, and Bernhard Schölkopf, "An Introduction to Kernel-Based Learning Algorithms" IEEE Transactions on neural networks, vol. 12, no. 2, march 2001.
[9] Ludmila I Kuncheva, and William J Faithfull, "PCA Feature Extraction for Change Detection in Multidimensional Unlabelled Data" IEEE Transactions on neural networks and learning systems.
[10] Mykola Pechenizkiy, Alexey Tsymbal, Seppo Puuronen, "PCA-based Feature Transformation for Classification: Issues in Medical Diagnostics".