REVIEW OF SIGNAL RECONSTRUCTION USING KPCA

Ramasubramanian Sundaram

Department of Electrical and Computer Engineering
Mississippi State University
Mississippi State, MS 39762 USA
email: sundaram@isip.mstate.edu

ABSTRACT

Principal Component Analysis is a technique used to linearly transform an original set of variables into a set of uncorrelated variables of smaller dimension that represents most of the information. It is also possible to transform variables in a nonlinear fashion. One such method namely Kernel Principal Component Analysis is a nonlinear extension of PCA where the principal components are computed in a high dimensional feature space which is nonlinearly related to the input space. This nonlinear transformation is performed using Kernel functions. The underlying assumption being that since a PCA in a high dimensional feature space can be formulated in terms of dot products, it can also be performed using the Kernel functions. In KPCA, the input data is first transformed to a high dimensional feature space via a nonlinear mapping and linear PCA is performed in this feature space. This paper will focus on analyzing the theory behind the KPCA technique and its merits and demerits.

1. INTRODUCTION

Kernel methods have been extensively used if the elements of the domain interact through inner products. This is possible because dot products can be obtained in the high dimensional feature spaces when the transformation is defined by the Kernel functions. The transformations are nonlinear in nature and is done so that a better classification is obtained. This forms the basis for Kernel Principal Components Analysis where the analysis is performed in a high dimensional feature space by a nonlinear transformation. This idea has gained momentum in recent times.

The objective of this paper is twofold. First, the theory behind the KPCA algorithm and the procedures involved in denoising are explored. This paper will also provide a critical review of the paper "The Signal Reconstruction of Speech by KPCA" by H Yan, X.G. Zhang and Y.D. Li published in the *Proceedings of the ICSLP,* October, 2000 in which KPCA has been proposed for denoising a speech signal. The critique will focus on the merits and demerits of the paper with respect to the KPCA algorithm and the experiments that were performed to substantiate the claim that KPCA performs better.

2. KERNEL THEORY

If the form of the class-conditional densities $p(x/\omega_i)$ are known then discrimination and classification problems can be solved by comparing the scores with thresholds. Usually, these functions are not known and non-parametric methods like the histograms, kernel methods, k-nearest neighbor are used [1].

2.1. Kernel Estimators

Kernels are used to estimate the densities when the form of the distribution is not known. A simple kernel to estimate the density function for a univariate case can be written as:

$$\hat{p}(x/\omega_m) = \frac{1}{n_m h} \sum_{i=1}^{n_m} K_0\left(\frac{x - x_i}{h}\right) \qquad (1)$$

where n_m is the total sample size for the class ω_m, x_i, i = 1, 2,..., n_m are the samples and K_0 is the Kernel function. The Kernel function can be defined as:

$$K_0(z) = \begin{cases} 0, \text{ for } |z| > 1 \\ 1, \text{ for } |z| < 1 \end{cases} \tag{2}$$

It is obvious that any point in the interval (x-h, x+h) would contribute to the estimate at x and at any point outside the interval contribution is zero [2]. Since such a harsh weighting is impractical, one can define a smoother kernel functions like,

$$K_0(z) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}z^2\right\} \tag{3}$$

so that all the sample points of a class contribute to the estimate but in an inversely proportional manner. The variable *h* determines how much each sample point contributes to the estimate and is called the smoothing parameter. Kernels like the one in (2) are called Gaussian Kernels. The choice of kernel function depends upon the accuracy needed and the computational cost. Several kernel mappings like Mercer Kernel [3], Reproducing Kernel map [3] exists that map the input patterns to the high dimensional feature space.

2.2. KPCA

Nonlinear algorithms can be reduced to linear ones in some high dimensional feature space F nonlinearly related to the input space. Using a kernel function instead of a dot product corresponds to mapping the data to high dimensional space F by a nonlinear mapping $\Phi(x)$ and taking the dot product there.

$$k(x, y) = (\Phi(x) \cdot \Phi(y)) \tag{4}$$

KPCA [3] is a one such feature space algorithm that where the dot products can be replaced *a priori* chosen kernel. KPCA carries out linear PCA in the feature space F. The extracted features are related to the input samples by the nonlinear relation given by

$$f_k(x) = \sum_{i=1}^{l} \alpha_i^k k(x_i, x) \tag{5}$$

where α_i^k are the components of the k^{th} eigenvector of the matrix $(k(x_i, x_j))_{ij}$. For PCA in F, we need to compute the eigenvectors ν and the eigenvalues λ of the covariance matrix C in the feature space where

$$C = \frac{1}{l}\sum_{i=1}^{l} \Phi(x_i) \cdot \Phi(x_i)' \tag{6}$$

Since computing dot products in F can be computationally costly kernels are used as in (5).

Such kernel transformation are not one-to-one because of their nonlinearity. It is generally a many-to-one mapping. This causes a preimage problem in that not all the points in the high dimensional space can be expressed as the image to single input pattern. Hence it is not possible to find the preimage of the mapped data and is difficult to find what input samples corresponds to the point in the mapped data. But it is possible to find approximate preimages using the principal components in the high dimensional space. If *z* the approximate preimage in the input space for the mapped test data T in the high dimensional space then

$$\delta(z) = \|T - \Phi(z)\|^2 \tag{7}$$

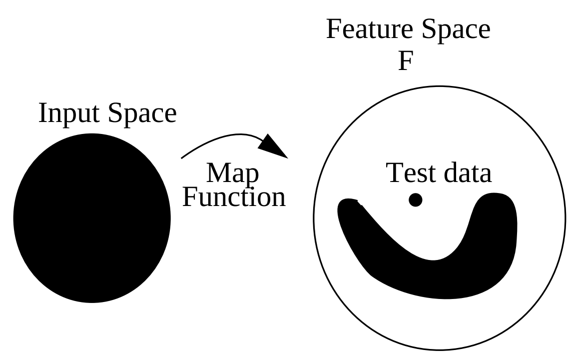

Figure 1: The preimage problem. Not each point in the span of the mapped data is necessarily the image of some input pattern.

is small. If P_n is the n-dimensional projection minimizing (7) then

$$\delta(z) = \sum_{i=1}^{l} \left\| P_n \Phi(z) - \Phi(z) \right\|^2 \quad (8)$$

This method is effectively used for denoising the input data. Given a noisy set of samples x, map it on to the high dimensional space, discard the higher components and then compute the preimage *z*. Here, the assumption is that the main structure in the data set is captured by the major principal components and the remaining components pick up the noise. Hence *z* can be considered to be the denoised version of *x*.

To find the best possible preimage for a point in the mapped data, iterative methods are used with the Gaussian kernels. The distance between the mapped test data point and its orthogonal projection is minimized over several iterations to get the preimage [3].

3. RESULTS AND ANALYSIS

Theory suggests that kernel PCA algorithm can be used for denoising the input data by nonlinear mapping. The paper being reviewed uses the KPCA method for denoising the input speech signal by following the iterative procedure as discussed in the previous section.

[3] suggests that KPCA works well with respect to denoising and hence can be used for the same. The paper under consideration uses the spectrogram as the way of showing an improvement in performance with respect to a noisy speech signal. It does not reveal anything about how noisy the data before using KPCA and by what factor the noise was reduced? All these makes the results looks inconclusive even though KPCA could have actually denoised the signal properly.

Several experiments could have been done by employing different kernels and analyzing their performance on the denoising problem. Can polynomial kernels be used for the same problem and if so would the effect be the same? This would have helped in the process of understanding the robustness of the algorithm across different kernels. Also, nothing is mentioned about the noise used in the signal. Were they manually added using noise encoders or were they present in the channel? Also no information about the SNR before and after the denoising process is provided which would have enabled the readers to understand the performance of the algorithm in a better manner.

An iterative process is chosen to find the approximate preimage for an input test data. The convergence criteria for the iterative process is not discussed. Theoretically, minimizing the distance measure as in (8) iteratively should give us the approximate preimage. Will there be a convergence for all kernel mappings and if so what is the initial value or estimate for the preimage to start with? Also, the paper does not discuss about the number of iterations before convergence can be expected for a reasonably good estimate.

Choice of high dimensional feature spaces plays an important role in defining the accuracy and computational costs of using kernels. The number of feature vectors in the high dimensional space determine the performance while denoising. The number of feature vectors could be equal to the number of training samples and reconstruction or denoising is done better if we choose more number of features in the feature space. Evidence of the training samples for KPCA or the number of features is not provided in the paper. These make the results difficult to comprehend as the performance is directly proportional to the number of features and so is the computational costs.

The paper focusses on the performance of KPCA as though it is the only algorithm that can solve the denoising problem. There are other algorithms like PCA, SVMs [4] etc. that are known to solve the same problem. The paper ought to have compared and contrasted the performance of KPCA with other algorithms. For example, analysis could have been performed to prove that kernel PCA works in some cases where linear PCA fails. This is true when the structures that needs to be extracted are nonlinear. Performance measures in terms of Signal to Noise Ratio could have been provided for various feature space based algorithms.

Another important factor that needs to be addressed is the problem of computational complexity of KPCA. If the margin of improvement is less when compared with linear PCA then is it worth pursuing with KPCA since the computational costs are more in the mapped feature space. This gives rise to another aspect of the algorithm about whether KPCA works for all cases and if it fails then the reasons for the failure. Also, with respect of denoising of an input signal the kernel PCA only provides us with a means of mapping points to their denoised versions. So the process of obtaining the denoised version of the input data still needs to be done. The paper does not mention any methods to obtain the final denoised data and the computations involved in it. Other feature space methods like Nonlinear Autoencoder, Principal Curves [3] provide an explicit one-dimensional parametrization of the denoised data directly without any postprocessing being involved. The reason behind pursuing kernel PCA method even though it provides only an approximate preimage of the mapped data needs to be addressed.

The paper shows that KPCA works better when the number of Principal Components considered for denoising is increased. This is obvious from the fact that most of the information about the data is in the major principal components and the noise is projected along the less important components. But the information about how many principal components needs to be considered for an optimal denoising performance is missing in the paper. This is an important factor since considering less number of principal components would not give a good estimate of the data and considering too many principal components would lead to degradation in denoising.

4. CONCLUSIONS

Denoising a speech signal invokes interest in the mind of the reader interested in Speech processing and recognition. Since noise does affect the performance of a system, it is important that we look into algorithms that does denoising. Kernel PCA looks like a good starting point for the same because it performs denoising and at the same time can provide a better classification of data withe the projection of the input data on to a high dimensional feature space. The drawback of KPCA method is that the interesting directions in feature space are defined in terms of all the points used to create the feature space. This is both computationally expensive and makes comprehension of the underlying structure of the data spaces more difficult.

This paper presented a review on the Kernel PCA algorithm with respect to denoising of a speech signal. While KPCA as a denoising algorithm looks promising, the paper reviewed leaves a lot to be desired. The paper does not substantiate the claims that KPCA works well for denoising a speech signal. The experimental set up is rather weak and the more work needs to be done to make sure that the algorithm is robust for all cases. Further work on KPCA should concentrate providing an estimate about the robustness of the algorithm. This involves analysis of performance of KPCA with respect to different kernels namely polynomial kernels, Mercer kernels etc. Also, comparisons should be made between various types of feature space algorithms to conclude that KPCA does really solve the denoising problem. The comparisons should be made by means of SNR's before and after denoising for all the methods. Also, since finding the preimage for a mapped input data is an iterative process, the convergence criteria and the initial estimate of the preimage needs to be established firmly.

5. REFERENCES

[1] D. J. Hand, *Discrimination and Classification*, Wiley Series in Probability and Mathematical Statistics, 1981.

[2] D. J. Hand, *Kernel Discriminant Analysis*, Research Studies Press, 1982.

[3] B. Scholkopf, S. Mika, C. Burges, P. Knirsch, K. R. Miller, G. Ratsch and A. Smola, "Input space vs. feature space in kernel-based methods," *IEEE Transactions on Neural Networks*, vol. 10, pp. 1000-1017, Sept. 1999.

[4] Alberto Ruiz and Pedro E. Lopez, "Nonlinear Kernel-Based Statistical Pattern Analysis", *IEEE Transactions on Neural Networks*, vol. 12, No. 1, January 2001.