PCA and Kernel PCA

Ezukwoke K.I[1], Samaneh M.J[2]

[1, 2] Department of Computer Science
Machine Learning and Data Mining
{ifeanyi.ezukwoke, samaneh}@etu.univ-st-etienne.fr
University Jean Monnet, Saint-Etienne, France

Abstract

Kernel principal component analysis (kernel-PCA) is a prominent non-linear extension of one of the most used classiical dimensionality reduction algorithms (PCA-Principal Component Analysis). In this paper, we present an inhaustive comparison between the classical PCA and the kernel-PCA. We compare the performance of PCa with kernel version using different kernels including (**rbf, cosine, correlation, polynomial**).

Keywords
Dimensionality reduction, PCA, kernel-PCA

1 INTRODUCTION

Principal Component Analysis (PCA) is an unsupervised dimension reduction algorithm for scaling high dimensional data to low dimensional data. The idea was initially proposed by Karl Pearson [1] as a statistical method to find lines or planes of best fit in the context of regression. It hase been subsequently cited and reproduced in various context and expecially in machine learning for **denoising**.

This implcitly means it is used for feature extraction since some of the features in the original space may not be required when using the eigenvectors corresponding to the largest eigenvalue.

2 DIMENSION REDUCTION

2.1 Principal Component Analysis (PCA)

The Classical PCA algorithm is to find a linear subspace of lower dimensionality than the original space. The PCA algorithm is described below in stepwise order. Note that $\sum$ is used to denote the **covariance**, which is different from summation $\sum_{i=i}^{N}$.

Algorithm 1: PCA Algorithm

Input : $x \in X$ where $x \in \mathbb{R}^D$
Output : $\hat{x} \in \mathbb{R}^k$ where $k << D$

begin
$\hat{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$;
$dx_i = x_i - \hat{x}$;
$\sum = \frac{1}{N} dx_i dx_i^T$;
$\sum \mathbf{u}_k = \lambda_k \mathbf{u}_k$;
$\mathbf{u}_k = \arg\mathrm{sort}(\mathbf{u}_k)$;
$\hat{x} = \mathbf{u}_k \cdot dx_i$;
end

PCA is limited in use to finding a linear subspace, by taking advantage of the linear dependence between the feature vectors. This limitation makes the classical PCA unsuitable for non-linearly seperable data (*makemoondataset*).

2.2 Kernel Principal Component Analysis (kernel-PCA)

The Kernel PCA is seeks to address the limitation of the classical PCA algorithm. By transforming the feature space x_i into a higher dimension space $\phi(x_i)$, the kernelized version is able to capture efficiently a subspace that reduces the dimension of the original feature space.

Algorithm 2: Kernel PCA Algorithm

Input : $\phi(x) \in \kappa(x,x_j)$ where $\phi(x) \in \mathbb{R}^D$. Let $\kappa = \kappa(x_i, x_j)$

Output : $\hat{\phi(x)} \in \mathbb{R}^k$ where $k << D$

begin
- Select a kernel κ;
- Construct Normalized kernel $\hat{\kappa} = \kappa - 2\kappa 1_{1/N} + 1_{1/N}\kappa 1_{1/N}$;
- Solve eigen problem $\kappa\alpha_{\mathbf{i}} = \lambda\alpha_{\mathbf{i}}$;
- Project data in new space $y_i = \sum_{i=1}^{N} \alpha\kappa$;

end

The kernel PCA algorithm is expressed only in terms of dot products, this trick allows us to construct different nonlinear versions of the classical PCA algorithm.

3 EXPERIMENTAL RESULT

3.1 Dataset

3.2 Performance analysis

4 CONCLUSION

References

[1] Pearson K. On lineas and Planes of closest fit to systems of points in space. *Philos Mag A*, 6:559–572, 1901.

[2] Zhu J, Hastie T. Kernel logistic regression and import vector machine. *J Comput Graphic Stat*, 14:185–205, 2005.

[3] Saunders C., Gammerman A., Vovk V., Ridge Regression Learning Algorithm in Dual Variables. *ICML 15th Internationl Conforence on Machine Learning.*, 1998.