Visualization in Medicine and Life Sciences (2013)
L. Linsen, H. -C. Hege, and B. Hamann (Editors)

Supervised Kernel Principal Component Analysis for Visual Sample-based Analysis of Gene Expression Data

Tran Van Long [1] and Lars Linsen [2]

[1] University of Transport and Communications, Hanoi, Vietnam
[2] Jacobs University, Bremen, Germany

Abstract

DNA microarray technology has enabled researchers to simultaneously investigate thousands of genes over hundreds of samples. Automatic classification of such data faces the challenge of dealing with smaller number of samples compared to a larger dimensionality. Dimension reduction techniques are often applied to overcome this. Recently, a number of supervised dimension reduction techniques have been developed. We present a novel supervised dimension reduction technique called supervised kernel principal component analysis and demonstrate its effectiveness for visual representation and visual analysis of gene expression data.

1. Introduction

In genomic research, DNA microarray technologies can monitor expression levels for large number of genes (up to 10^6) simultaneously, while the number of samples is usually in the range of hundreds. The small number of samples when compared to the high number of genes makes it challenging to understand and interpret the gene expression data. To explore microarray gene expression data, the data need to be analyzed and presented in a way that biologists can easily understand them. Multivariate data analysis and visualization techniques support this endeavor.

One important aspect is the assigning of samples into different disjoint classes (or categories), such as different types of cancer or healthy vs. non-healthy. Reducing the data dimensionality [Zha06, ZKL08] is considered as one of the most promising directions of research in this context. Principal component analysis (PCA) is the most commonly used technique for unsupervised dimension reduction. PCA linearly projects data onto a set of new coordinates (principal components) that preserve the variance of the data set as much as possible. Linear discriminant analysis (LDA) is one of the most common techniques for supervised dimensionality reduction. LDA is also a linear technique and tries to preserve classes while generating maximal separability between classes. However, it has been shown that both approaches did not successfully separate classes when applied to gene expression data [BKR*10].

In this paper, we propose a novel supervised dimension reduction technique, which is an extension of the unsupervised kernel principal component analysis (KPCA), which itself is an extension of PCA considering non-linear projections. We show that our new method, which we call supervised kernel principal component analysis (SKPCA), clearly separates the classes. Thus, it is an effective method to visually represent clustered gene expression data. Moreover, we also apply our method to new samples that have not been categorized. We show that the samples are dragged towards the respective class, which allows for visual analysis of gene expression data.

2. Related work

Within the last decade, non-linear dimension reduction techniques such as KPCA, ISOMAP, Locally Linear Embedding (LLE), Laplacian Eigenmaps (LE), Diffusion maps, and Maximum Variance Unfolding (MVU) have gained much attention [JAL07]. These techniques can be considered as unsupervised dimension reduction methods. The main problem for applying non-linear dimension reduction techniques to gene expression data is the lack of meaning of the distance in a high-dimensional space [BGRS99]. We propose to use a supervised technique to overcome this issue. Our work is in line with recent advances in supervised dimension reduction [XDCZH05, Agg06, KBH11, YL11, AMDSCD12, SSDJ12]. We present a new method and show its suitability for sample-based gene expression analysis. The main advantage of our approach is that we use the similarity dy-

DOI: 10.2312/PE.VMLS.VMLS2013.055-059

namically, i.e., we change the similarity measurements by assigning supervised kernel matrix. More precisely, we use the class label information to define the similarity through the scalar product in a Hilbert space.

3. Problem Specification

A gene expression data set from a microarray experiment can be expressed as a gene expression matrix [JTZ04]

$$X = \begin{pmatrix} x_{11} & x_{21} & \dots & x_{n1} \\ x_{12} & x_{22} & \dots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1p} & x_{2p} & \dots & x_{np} \end{pmatrix}$$

where each column represents the expressions of al genes for one sample, i.e., one patient, and each row represents the expressions of one gene for all samples, i.e., each entry x_{ij} is the measurement of expression of gene j in sample i. Typically, the number of samples n is in the range of hundreds, while the number of genes is in the range of thousands. Gene expression data analysis can be classify in two categories: gene-based and sample-based. In the gene-based analysis we can consider each gene as a point in an n-dimensional space (large n, small p). In the sample-based analysis we consider each sample as a point in a p-dimensional space (large p, small n). In this paper we focus on sample-based analysis. In sample-based analysis of gene expression data, one typically investigates diseased vs. normal samples. The goals to find the structures or substructures of the samples [JTZ04]. The challenging of sample-based analysis we have a small data size in a high-dimensional space.

4. Unsupervised dimension reduction

In the following we describe the problem of sample-based gene expression analysis using unsupervised PCA and KPCA, which will be extended to a supervised version afterwards.

4.1. PCA

PCA can be considered as an orthogonal projection into a lower dimensional linear space, such that the variance of the projected data is maximized or, equivalently, the mean-squared distance between data points and their projections is minimized. Consider a data set $X = [x_1, x_2, \dots, x_n]$ in a p-dimensional space. Without loss of general, we can assume the data set to be centered, i.e., $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = 0$.

Frequently, PCA is applied to the case of a large number of samples and a lower number of dimensions. When considering it the other way round, the data set lies in a linear space spanned by $x_1, x_2, \dots, x_n$ and the dimensionality of this linear space is always less than n. So, we can define the first principal components as a linear combination of $x_1, x_2, \dots, x_n$, i.e., $v = \sum_{i=1}^{n} \alpha_i x_i$ with $||v||^2 = 1$.

A data point x_i is presented by $\langle x_i, v\rangle$ on the principal component. We find the principal component v that maximizes $\sum_{i=1}^{n}\langle x_i, v\rangle^2$. Let $K_{ij} = \langle x_i, x_j\rangle$, $K = (K_{ij}) = X^T X$, then, we can conclude that $\sum_{i=1}^{n}\langle x_i, v\rangle^2 = \alpha^T K^2 \alpha$ with $||v||^2 = \alpha^T K \alpha$. So we find the first principal component v by finding vector $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ that maximizes $\alpha^T K^2 \alpha$ under constraint $\alpha^T K \alpha = 1$.

For visual representation we find the d largest eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$ of matrix K and corresponding eigenvectors $\alpha_1, \alpha_2, \dots, \alpha_d$ by the power method [GVL96]†. The data set X is presented by $Y = [y_1, y_2, \dots, y_n]$ in a d-dimensional space where $y_i = (\sqrt{\lambda_1}\alpha_{1i}, \dots, \sqrt{\lambda_d}\alpha_{di})$.

4.2. Kernel PCA

KPCA [SS01] defines a kernel is on an input space X by $K : X \times X \longrightarrow \mathbb{R}$. The kernel $K(x, x')$ is satisfying the property of being positive definite. Based on Mercer's theorem, we can define a Hilbert space $\mathbb{H}$ and a feature map $\Phi : X \longrightarrow \mathbb{H}$ such that

$$K(x, x') = \sum_{i=1}^{\infty} \lambda_j \Psi_j(x) \Psi_j(x'),$$

where λ_j and $\Psi_j(x)$ are eigenvalues and eigenvectors of a linear operator $T(f(x)) = \int_X K(x, x') f(x') dx'$. Then, the feature map is $\Phi(x) = (\sqrt{\lambda_j}\Psi_j(x))$ and the scalar product in the Hilbert space $\mathbb{H}$ is defined as $\langle \Phi(x), \Phi(x')\rangle = K(x, x')$.

Considering n observations $X = [x_1, x_2, \dots, x_n]$ in a p-dimensional space. We define a map $\Phi : x \longmapsto \Phi(x)$ in an infinite Hilbert space with an inner product that define by the kernel function $K(x_i, x_j) = K_{ij} = \langle \Phi(x_i), \Phi(x_j)\rangle$. We denote the set of data points in the Hilbert space by $\Phi_i = \Phi(x_i), i = 1, 2, \dots, n$. Centering by the mean vector $\bar{\Phi} = \frac{1}{n}\sum_{i=1}^{n} \Phi_i$, we define $\hat{\Phi}_i = \Phi_i - \bar{\Phi}$ and $\hat{K}_{ij} = \hat{K}(x_i, x_j) = \langle \hat{\Phi}_i, \hat{\Phi}_j\rangle$. Then, we proceed as for the PCA method to find the principal component in the form $V = \sum_{i=1}^{n} \alpha_i \hat{\Phi}_i$ that maximizes the variance $\sum_{i=1}^{n}\langle V, \hat{\Phi}_i\rangle^2 = \sum_{i=1}^{n}\left(\sum_{i=1}^{n} \alpha_j \hat{K}(x_i, x_j)\right)^2 = \alpha^T \hat{K}^2 \alpha$ with $\langle V, V\rangle = \alpha^T \hat{K} \alpha = 1$. The data are presented by vector y with the ith element $y_i = \langle V, \hat{\Phi}_i\rangle = \sum_{j=1}^{n} \hat{K}_{ji}\alpha_j = (\hat{K}\alpha)_i, i = 1, 2, \dots, n$ or $y = \hat{K}\alpha$.

For visual representation we find the d largest eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_d$ and the corresponding unit eigenvectors

† $||\alpha_i||^2 = 1, i = 1, 2, \dots, d$

$\alpha_1, \alpha_2, \ldots, \alpha_d$. To each given point x_i we display the respective point

$$y_i = \left(\sqrt{\lambda}_1\alpha_{1i}, \sqrt{\lambda}_2\alpha_{2i}, \ldots, \sqrt{\lambda}_p\alpha_{di}\right).$$

5. Supervised Kernel PCA

5.1. Visual representation

We propose a supervised version of the KPCA. We assume a classified training set, i.e., the data are classified into k classes with a labeling function $\ell : X \longrightarrow \{1,2,\ldots,k\}$ such that two points x_i and x_j belong to the same class if $\ell(x_i) = \ell(x_j)$. For visualizing the classes well separated in a lower dimensional space, we need to incorporate the class information into the kernel function (or matrix). Without loss of generality, we can assume the range of the kernel function to be $[0,1]$. Then, we define a supervised kernel function that involves the label information classes as follows:

$$K_s(x_i,x_j) = \begin{cases} K(x_i,x_j)+\mu & \text{if } \ell(x_i) = \ell(x_j), \\ K(x_i,x_j) & \text{otherwise} \end{cases}$$

where μ is a positive parameter. In the case $\mu = 0$, we obtain the unsupervised version of KPCA. If $\mu = +\infty$, each class degenerates to a single point, i.e., the data set is visualized as k points.

The algorithm can be summarize as

1. Compute kernel matrix $K = (K(x_i,x_j))$.
2. Compute supervised kernel matrix K_s.
3. Center kernel matrix by $\hat{K} = (I - \frac{1}{n}ee^T)K_s(I - \frac{1}{n}ee^T)$.
4. Find d largest eigenvalues λ_i and corresponding eigenvectors α_i of matrix $\hat{K}$ using power method.
5. Compute $y_i = \left(\sqrt{\lambda}_1\alpha_{1i}, \sqrt{\lambda}_2\alpha_{2i}, \ldots, \sqrt{\lambda}_d\alpha_{di}\right)$.

5.2. Classification

In addition to achieve a visual representation with highly separated classes, SKPCA can also be applied for classification purposes. Here, we assume a classified training set that is used to compute the coefficients α_i and a testing set, where the classes are unknown. More precisely, assuming that data point x_i gets assigned location y_i in the lower-dimensional space, then coefficients α_i are chosen such that $\sum_{i=1}^{n} K(x_i,x_j)\alpha_j = y_i$ for all data points x_i in the training data set. Hence, the coefficients are determined by a linear equation $K\alpha = y$, where kernel matrix K can have full rank [SS01] such that we can find unique coefficients α.

For the testing data set x, we can, then, find the location in the lower-dimensional space via the map $y = f(x) = \sum_{i=1}^{n} K(x,x_i)\alpha_i$. Then, a visual analysis step is involved, where the location of the points y of the testing set are compared with the positions y_i of the training data set to induce a classification of testing data point x. One helpful feature of our proposed method is that we can modify parameter μ and observe the changes in the visual representation, which can help improving the decision making in the classification process.

6. Results

First, we want to investigate the visual representation of classified data. For a proof of concept, we start with the intensively studied Iris data set that includes $n = 150$ data points with $p = 4$ features, which are classified into three classes (each class containing 50 points). Figure 1 shows the result of visually representing the data in a 2D space using KPCA and SKPCA. We use a Gaussian kernel function $K(x_i,x_j) = \exp(-\frac{||x_i-x_j||^2}{2})$. The three classes are encoded using different colors. We can observe that the KPCA method (left image) produces a layout where two of the class are severely overlapping. Our SKPCA approach with parameter $\mu = 1$ manages to clearly separate the three classes.

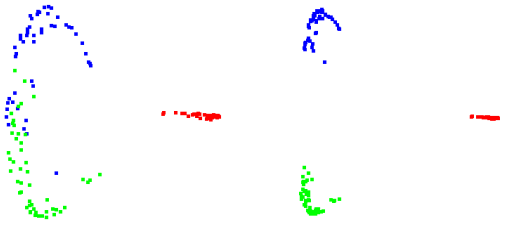

Figure 1: *Iris data set: KPCA (left) produces overlapping classes, while SKPCA (right) manages to separate all classes.*

The second data set we consider is the Colon Cancer data set [ABN*99] containing $n = 62$ samples with $p = 2,000$ genes. The samples are categorized into two classes, namely 40 tumor tissues and 22 normal tissues. For sample-based gene expression data we use the Pearson coefficient to replace Euclidean distance. The Pearson coefficient can be considered as the scalar product kernel function. The data set is centered and normalized. We use the Pearson coefficient kernel function

$$\rho(x_i,x_j) = \frac{p\sum_{k=1}^{p} x_{ik}x_{jk} - \sum_{k=1}^{p} x_{ik}\sum_{k=1}^{p} x_{jk}}{\sqrt{p\sum_{k=1}^{p} x_{ik}^2 - \left(\sum_{k=1}^{p} x_{ik}\right)^2}\sqrt{p\sum_{k=1}^{p} x_{jk}^2 - \left(\sum_{k=1}^{p} x_{jk}\right)^2}},$$

and the scalar kernel function $K(x_i,x_j) = [\rho(x_i,x_j)]^m$ with $m = 2$ (default value). We again compare the results of KPCA with SKPCA in Figure 2. Again, KPCA produces a layout with overlapping classes, while SKPCA clearly separates the two classes.

The third data set we considered is the MLL Leukemia

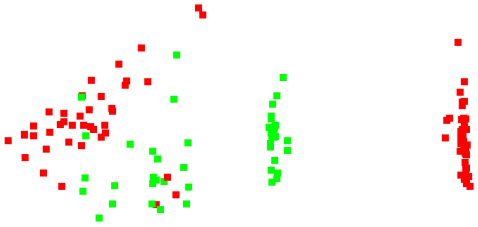

Figure 2: *Sample-based gene expression data (Colon Cancer data set): KPCA (left) produces overlapping classes, while SKPCA (right) manages to separate the two classes.*

data set comprised of $n = 72$ leukemia samples, which can be grouped into the three classes with ALL (24 samples), MLL (20 samples), and AML (28 samples). The number of genes is $p = 12,582$. Here, the dimensionality is really high, while the number of samples is rather low. Again, KPCA produces mixed classes, while SPKCA clearly displays the three classes well separated, see Figure 3.

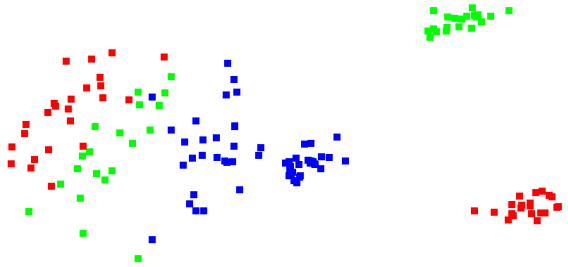

Figure 3: *High-dimensional sample-based gene expression data (MLL Leukemia data set): KPCA (left) produces mixed classes, while SKPCA (right) displays three separated classes.*

Next, we are going to investigate the usefulness of SKPCA for classification in a visual analysis process. We use the Leukemia ALL-AML cancer data set [GST*99]. It that contains $n = 72$ samples with $p = 7129$ genes. The samples are classified as ALL or AML. We perform SKPCA with a training data set that includes 38 samples (27 ALL and 11 AML). The testing data set contains 34 samples (20 ALL and 14 AML). Figure 4 shows the results by dispalying both training and testing data set with different choices of the parameter μ ($\mu = 0.0$, 0.01, 0.1, and 0.2). The smaller dots represent the training data set (blue for AML and red for ALL). The larger dots represent the testing data set (green for AML and pink for ALL). It can be observed that the samples are mixed for KPCA ($\mu = 0.0$), but for increasing μ the training data gets clearly separated and the respective training data is dragged towards those separated clusters of the training data. Hence when increasing μ, one can observe that the testing data start moving in opposite direction, which can be exploited to classify the data.

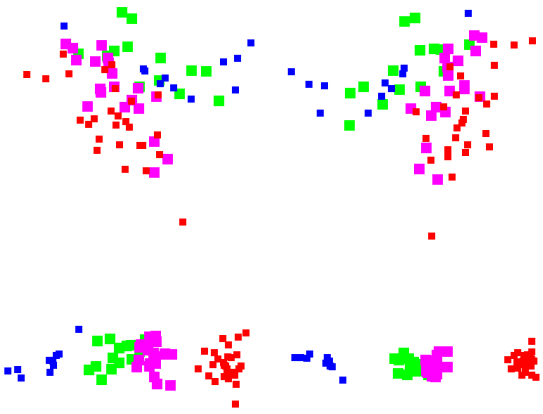

Figure 4: *Classification with SKPCA: training data shown in blue (AML) and red (ALL), testing data shown in green (AML) and pink (ALL). (upper left) KPCA; (upper right) SKPCA with $\mu = 0.01$; (lower left) SKPCA with $\mu = 0.1$; (lower right) SKPCA with $\mu = 0.2$;*

7. Conclusions

We proposed a novel supervised dimension reduction technique called supervised kernel principal component analysis (SKPCA) for visualizing classes of data sets with a relatively small number of samples when compared to a large number of dimensions. We applied our method to the visual representation of classified sample-based gene expression data. All experiments show that SKPCA gets better separation of clusters than the standard KPCA. The method contains a control parameter μ, that can be used to control the spread or shrinkage of clusters. We also applied our method to support the classification of new samples based on a training set (known classification) and a testing set (to be classified). We have shown that our method can also be useful in this regard.

The performance of our methods is reduced when the classes differ much in size. In future work, we want to investigate how this can be addressed. Also, we want to extend our approach to handle hierarchical representation of classes.

Acknowledgements

This work was funded by the Vietnam's National Foundation for Science and Technology Development (NAFOSTED) via a research grant for fundamental sciences, grant number: 102.01-2012.04.

References

[ABN*99] ALON U., BARKAI N., NOTTERMAN D., GISH K., YBARRA S., MACK D., LEVINE A. J.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Nat. Acad. Sci. USA 96* (1999), 6745–6750. 3

[Agg06] AGGARWAL C. C.: A framework for local supervised dimensionality reduction of high dimensional data. In *SIAM Conference on Data Mining* (2006), pp. 360–371. 1

[AMDSCD12] ALVAREZ-MEZA A. M., DAZA-SANTACOLOMA G., CASTELLANOS-DOMINGUEZ G.: Biomedical data analysis by supervised manifold learning. *34th Annual International Conference of the IEEE EMBS San Diego, California USA, 28 August - 1 September, 2012* (2012). 1

[BGRS99] BEYER K., GOLDSTEIN J., RAMAKRISHNAN R., SHAFT U.: When is "nearest neighbor" meaningful? *Database Theory–ICDT 99* (1999), 217–235. 1

[BKR*10] BARTENHAGEN C., KLEIN H.-U., RUCKERT C., JIANG X., DUGAS M.: Comparative study of unsupervised dimension reduction techniques for the visualization of microarray gene expression data. *BMC Bioinformatics 2010,11:567 11* (2010), 567–578. 1

[GST*99] GOLUB T. R., SLONIM D. K., TAMAYO P., HUARD C., GAASENBEEK M., MESIROV J. P., COLLER H., LOH M. L., DOWNING J. R., CALIGIURI M. A., ET AL.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science 286*, 5439 (1999), 531–537. 4

[GVL96] GOLUB G. H., VAN LOAN C. F.: *Matrix computations*, vol. 3. Johns Hopkins University Press, 1996. 2

[JAL07] JOHN A. LEE M. V.: *Nonlinear Dimensionality Reduction*. Springer, 2007. 1

[JTZ04] JIANG D., TANG C., ZHANG A.: Cluster analysis for gene expression data: A survey. *Knowledge and Data Engineering, IEEE Transactions on 16*, 11 (2004), 1370–1386. 2

[KBH11] KERSTIN B., BIEHL M., HAMMER B.: Supervised dimension reduction mappings. *Proc. ESANN. 2011.* (2011). 1

[SS01] SCHÖLKOPF B., SMOLA A. J.: *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2001. 2, 3

[SSDJ12] SU Y., SOFIEN B., DONGWON L., JESSE B.: Semi-supervised dimensionality reduction for analyzing high-dimensional data with constraints. *Neurocomputing 76*, 1 (2012), 114–124. 1

[XDCZH05] XIN G., DE-CHUAN Z., ZHI-HUA Z.: Supervised nonlinear dimensionality reduction for visualization and classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 35*, 6 (2005), 1098–1107. 1

[YL11] YAN C., LIYA F.: A novel supervised dimensionality reduction algorithm: Graph-based fisher analysis. *Pattern Recognition* (2011). 1

[Zha06] ZHANG A.: *Advanced Analysis of Gene Expression Microarray Data*. World Scientific Publishing, 2006. 1

[ZKL08] ZHANG L., KULJIS J., LIU X.: Information visualization for dna microarray data analysis: a critical review. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews 38*, 1 (Jan. 2008), 42–54. 1