1

# Kernel Methods for Dimensionality Reduction Applied to the «Omics» Data

Ferran Reverter, Esteban Vegas and Josep M. Oller
*Department of Statistics, University of Barcelona*
*Spain*

## 1. Introduction

Microarray technology has been advanced to the point at which the simultaneous monitoring of gene expression on a genome scale is now possible. Microarray experiments often aim to identify individual genes that are differentially expressed under distinct conditions, such as between two or more phenotypes, cell lines, under different treatment types or diseased and healthy subjects. Such experiments may be the first step towards inferring gene function and constructing gene networks in systems biology.

The term ”gene expression profile” refers to the gene expression values on all arrays for a given gene in different groups of arrays. Frequently, a summary statistic of the gene expression values, such as the mean or the median, is also reported. Dot plots of the gene expression measurements in subsets of arrays, and line plots of the summaries of gene expression measurements are the most common plots used to display gene expression data (See for example Chambers (1983) and references therein).

An ever increasing number of techniques are being applied to detect genes which have similar expression profiles from microarray experiments. Techniques such clustering (Eisen et al. (1998)), self organization map (Tamayo et al. (1999)) have been applied to the analysis of gene expression data. Also we can found several applications on microarray analysis based on distinct machine learning methods such as Gaussian processes (Chu et al. (2005); Zhao & Cheung (2007)), Boosting (Dettling (2004)) and Random Forest (Diaz (2006)). It is useful to find gene/sample clusters with similar gene expression patterns for interpreting the microarray data.

However, due to the large number of genes involved it might be more effective to display these data on a low dimensional plot. Recently, several authors have explored dimension reduction techniques. Alter et al. (2000) analyzed microarray data using singular value decomposition (SVD), Fellenberg et al. (2001) used correspondence analysis to visualize genes and tissues, Pittelkow & Wilson (2003) and Park et al. (2008) used several variants of biplot methods as a visualization tool for the analysis of microarray data. Visualizing gene expression may facilitate the identification of genes with similar expression patterns.

Principal component analysis has a very long history and is known to very powerful for the linear case. However, the sample space that many research problems are facing, especially the

sample space of microarray data, are considered nonlinear in nature. One reason might be that the interaction of the genes are not completely understood. Many biological pathways are still beyond human comprehension. It is then quite naive to assume that the genes should be connected in a linear fashion. Following this line of thought, research on nonlinear dimensionality reduction for microarray gene expression data has increased (Zhenqiu et al. (2005), Xuehua & Lan (2009) and references therein). Finding methods that can handle such data is of great importance if as much information as possible is to be gleaned.

Kernel representation offers an alternative to nonlinear functions by projecting the data into a high-dimensional feature space, which increases the computational power of linear learning machines, (see for instance Shawe-Taylor & Cristianini (2004); Scholkopf & Smola (2002)).

Kernel methods enable us to construct different nonlinear versions of any algorithm which can be expressed solely in terms of dot products, known as the kernel trick. Thus, kernel algorithms avoid the explicit usage of the input variables in the statistical learning task. Kernel machines can be used to implement several learning algorithms but they usually act as a black-box with respect to the input variables. This could be a drawback in biplot displays in which we pursue the simultaneous representation of samples and input variables.

In this work we develop a procedure for enrich the interpretability of Kernel PCA by adding in the plot the representation of input variables. We used the radial basis kernel (Gaussian kernel) in our implementation however, the procedure we have introduced is also applicable in cases that may be more appropriated to use any other smooth kernel, for example the Linear kernel which supplies standard PCA analysis. In particular, for each input variable (gene) we can represent locally the direction of maximum variation of the gene expression. As we describe below, our implementation enables us to extract the nonlinear features without discarding the simultaneous display of input variables (genes) and samples (microarrays).

## 2. Kernel PCA methodology

KPCA is a nonlinear equivalent of classical PCA that uses methods inspired by statistical learning theory. We describe shortly the KPCA method from Scholkopf et al. (1998).

Given a set of observations $\mathbf{x}_i \in \mathbb{R}^n$, $i = 1, \ldots, m$. Let us consider a dot product space $F$ related to the input space by a map $\phi : \mathbb{R}^n \to F$ which is possibly nonlinear. The feature space $F$ could have an arbitrarily large, and possibly infinite, dimension. Hereafter upper case characters are used for elements of $F$, while lower case characters denote elements of $\mathbb{R}^n$. We assume that we are dealing with centered data $\sum_{i=1}^{m} \phi(\mathbf{x}_i) = 0$.

In $F$ the covariance matrix takes the form

$$\mathbf{C} = \frac{1}{m} \sum_{j=1}^{m} \phi(\mathbf{x}_j)\phi(\mathbf{x}_j)^{\intercal}.$$

We have to find eigenvalues $\lambda \geq 0$ and nonzero eigenvectors $\mathbf{V} \in F \backslash \{0\}$ satisfying

$$\mathbf{CV} = \lambda \mathbf{V}.$$

As is well known all solutions $\mathbf{V}$ with $\lambda \neq 0$ lie in the span of $\{\phi(\mathbf{x}_i)\}_{i=1}^m$. This has two consequences: first we may instead consider the set of equations

$$\langle \phi(\mathbf{x}_k), \mathbf{C}\mathbf{V} \rangle = \lambda \langle \phi(\mathbf{x}_k), \mathbf{V} \rangle , \tag{1}$$

for all $k = 1, \ldots, m$, and second there exist coefficients $\alpha_i$, $i = 1, \ldots, m$ such that

$$\mathbf{V} = \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i). \tag{2}$$

Combining (1) and (2) we get the dual representation of the eigenvalue problem

$$\frac{1}{m} \sum_{i=1}^{m} \alpha_i \left\langle \phi(\mathbf{x}_k), \sum_{j=1}^{m} \phi(\mathbf{x}_j) \left\langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_i) \right\rangle \right\rangle = \lambda \sum_{i=1}^{m} \alpha_i \langle \phi(\mathbf{x}_k), \phi(\mathbf{x}_i) \rangle ,$$

for all $k = 1, \ldots m$. Defining a $m \times m$ matrix $K$ by $K_{ij} := \left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \right\rangle$, this reads

$$K^2 \boldsymbol{\alpha} = m\lambda K \boldsymbol{\alpha}, \tag{3}$$

where $\boldsymbol{\alpha}$ denotes the column vector with entries $\alpha_1, ..., \alpha_m$. To find the solutions of (3), we solve the dual eigenvalue problem

$$K\boldsymbol{\alpha} = m\lambda \boldsymbol{\alpha}, \tag{4}$$

for nonzero eigenvalues. It can be shown that this yields all solutions of (3) that are of interest for us. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$ the eigenvalues of $K$ and $\boldsymbol{\alpha}^1, ..., \boldsymbol{\alpha}^m$ the corresponding set of eigenvectors, with $\lambda_r$ being the last nonzero eigenvalue. We normalize $\boldsymbol{\alpha}^1, ..., \boldsymbol{\alpha}^r$ by requiring that the corresponding vectors in $F$ be normalized $\left\langle \mathbf{V}^k, \mathbf{V}^k \right\rangle = 1$, for all $k = 1, ..., r$. Taking into account (2) and (4), we may rewrite the normalization condition for $\boldsymbol{\alpha}^1, ..., \boldsymbol{\alpha}^r$ in this way

$$1 = \sum_{i,j}^{m} \alpha_i^k \alpha_j^k \left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \right\rangle = \sum_{i,j}^{m} \alpha_i^k \alpha_j^k K_{ij} = \left\langle \boldsymbol{\alpha}^k, K\boldsymbol{\alpha}^k \right\rangle = \lambda_k \left\langle \boldsymbol{\alpha}^k, \boldsymbol{\alpha}^k \right\rangle . \tag{5}$$

For the purpose of principal component extraction, we need to compute the projections onto the eigenvectors $\mathbf{V}^k$ in $F$, $k = 1, ..., r$. Let $\mathbf{y}$ be a test point, with an image $\phi(\mathbf{y})$ in $F$. Then

$$\left\langle \mathbf{V}^k, \phi(\mathbf{y}) \right\rangle = \sum_{i=1}^{m} \alpha_i^k \langle \phi(\mathbf{x}_i), \phi(\mathbf{y}) \rangle , \tag{6}$$

are the nonlinear principal component corresponding to $\phi$.

### 2.1 Centering in feature space

For the sake of simplicity, we have made the assumption that the observations are centered. This is easy to achieve in input space but harder in $F$, because we cannot explicitly compute the mean of the mapped observations in $F$. There is, however, a way to do it.

Given any $\phi$ and any set of observations $\mathbf{x}_1, ..., \mathbf{x}_m$, let us define

$$\bar{\phi} := \frac{1}{m} \sum_{i=1}^{m} \phi(\mathbf{x}_i)$$

then, the points

$$\tilde{\phi}(\mathbf{x}_i) = \phi(\mathbf{x}_i) - \bar{\phi} \tag{7}$$

will be centered. Thus the assumption made above now hold, and we go on to define covariance matrix and dot product matrix $\tilde{K}_{ij} = \left\langle \tilde{\phi}(\mathbf{x}_i), \tilde{\phi}(\mathbf{x}_j) \right\rangle$ in $F$. We arrive at our already familiar eigenvalue problem

$$m\tilde{\lambda}\tilde{\boldsymbol{\alpha}} = \tilde{K}\tilde{\boldsymbol{\alpha}}, \tag{8}$$

with $\tilde{\boldsymbol{\alpha}}$ being the expansion coefficients of an eigenvector (in $F$) in terms of the centered points (7)

$$\tilde{\mathbf{V}} = \sum_{i=1}^{m} \tilde{\alpha}_i \tilde{\phi}(\mathbf{x}_i). \tag{9}$$

Because we do not have the centered data (7), we cannot compute $\tilde{K}$ explicitly, however we can express it in terms of its noncentered counterpart $K$. In the following, we shall use $K_{ij} = \left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \right\rangle$. To compute $\tilde{K}_{ij} = \left\langle \tilde{\phi}(\mathbf{x}_i), \tilde{\phi}(\mathbf{x}_j) \right\rangle$, we have:

$$\begin{aligned} \tilde{K}_{ij} &= \left\langle \phi(\mathbf{x}_i) - \bar{\phi}, \phi(\mathbf{x}_j) - \bar{\phi} \right\rangle \\ &= K_{ij} - \frac{1}{m} \sum_{t=1}^{m} K_{it} - \frac{1}{m} \sum_{s=1}^{m} K_{sj} + \frac{1}{m^2} \sum_{s,t=1}^{m} K_{st}. \end{aligned}$$

Using the vector $\mathbf{1}_m = (1, ..., 1)^\intercal$, we get the more compact expression

$$\tilde{K} = K - \frac{1}{m} K \mathbf{1}_m \mathbf{1}_m^\intercal - \frac{1}{m} \mathbf{1}_m \mathbf{1}_m^T K + \frac{1}{m^2} (\mathbf{1}_m^\intercal K \mathbf{1}_m) \mathbf{1}_m \mathbf{1}_m^\intercal.$$

We thus can compute $\tilde{K}$ from $K$ and solve the eigenvalue problem (8). As in equation (5), the solution $\tilde{\alpha}^k$, $k = 1, ..., r$, are normalized by normalizing the corresponding vector $\tilde{\mathbf{V}}^k$ in $F$, which translates into $\tilde{\lambda}_k \left\langle \tilde{\alpha}^k, \tilde{\alpha}^k \right\rangle = 1$.

Consider a test point $\mathbf{y}$. To find its coordinates we compute projections of centered $\phi$-images of $\mathbf{y}$ onto the eigenvectors of the covariance matrix of the centered points,

$$\begin{aligned} \left\langle \tilde{\phi}(\mathbf{y}), \tilde{\mathbf{V}}^k \right\rangle &= \left\langle \phi(\mathbf{y}) - \bar{\phi}, \tilde{\mathbf{V}}^k \right\rangle = \sum_{i=1}^{m} \tilde{\alpha}_i^k \left\langle \phi(\mathbf{y}) - \bar{\phi}, \phi(\mathbf{x}_i) - \bar{\phi} \right\rangle \\ &= \sum_{i=1}^{m} \tilde{\alpha}_i^k \big( \langle \phi(\mathbf{y}), \phi(\mathbf{x}_i) \rangle - \langle \bar{\phi}, \phi(\mathbf{x}_i) \rangle - \langle \phi(\mathbf{y}), \bar{\phi} \rangle + \langle \bar{\phi}, \bar{\phi} \rangle \big) \\ &= \sum_{i=1}^{m} \tilde{\alpha}_i^k \Big\{ K(\mathbf{y}, \mathbf{x}_i)) - \frac{1}{m} \sum_{s=1}^{m} K(\mathbf{x}_s, \mathbf{x}_i)) - \frac{1}{m} \sum_{s=1}^{m} K(\mathbf{y}, \mathbf{x}_s) + \frac{1}{m^2} \sum_{s,t=1}^{m} K(\mathbf{x}_s, \mathbf{x}_t) \Big\}. \end{aligned}$$

Introducing the vector

$$\mathbf{Z} = \Big(K(\mathbf{y}, \mathbf{x}_i)\Big)_{m\times 1}. \tag{10}$$

Then,

$$\begin{aligned}\Big(\Big\langle \tilde{\phi}(\mathbf{y}), \tilde{\mathbf{V}}^k \Big\rangle\Big)_{1\times r} &= \mathbf{Z}^\intercal \tilde{\mathbf{V}} - \frac{1}{m}\mathbf{1}_m^\intercal K\tilde{\mathbf{V}} - \frac{1}{m}(\mathbf{Z}^\intercal \mathbf{1}_m)\mathbf{1}_m^\intercal \tilde{\mathbf{V}} + \frac{1}{m^2}(\mathbf{1}_m^\intercal K\mathbf{1}_m)\mathbf{1}_m^\intercal \tilde{\mathbf{V}} \\ &= \mathbf{Z}^\intercal\Big(\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\intercal\Big)\tilde{\mathbf{V}} - \frac{1}{m}\mathbf{1}_m^\intercal K\Big(\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\intercal\Big)\tilde{\mathbf{V}} \\ &= \Big(\mathbf{Z}^\intercal - \frac{1}{m}\mathbf{1}_m^\intercal K\Big)\Big(\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\intercal\Big)\tilde{\mathbf{V}}, \end{aligned} \tag{11}$$

where $\tilde{\mathbf{V}}$ is a $m \times r$ matrix whose columns are the eigenvectors $\tilde{\mathbf{V}}^1, ..., \tilde{\mathbf{V}}^r$.

Notice that the KPCA uses only implicitly the input variables since the algorithm formulates the reduction of the dimension in the feature space through the kernel function evaluation. Thus KPCA is usefulness for nonlinear feature extraction by reducing the dimension but not to explain the selected features by means the input variables.

## 3. Adding input variable information into Kernel PCA

In order to get interpretability we add supplementary information into KPCA representation. We have developed a procedure to project any given input variable onto the subspace spanned by the eigenvectors (9).

We can consider that our observations are realizations of the random vector $X = (X_1, ..., X_n)$. Then to represent the prominence of the input variable $X_k$ in the KPCA. We take a set of points of the form $\mathbf{y} = \mathbf{a} + s\mathbf{e}_k \in \mathbb{R}^n$ where $\mathbf{e}_k = (0, ..., 1, ..., 0) \in \mathbb{R}^n$, $s \in \mathbb{R}$, where $k$-th component is equal 1 and otherwise are 0. Then, we can compute the projections of the image of these points $\phi(\mathbf{y})$ onto the subspace spanned by the eigenvectors (9).

Taking into account equation (11) the induced curve in the eigenspace expressed in matrix form is given by the row vector:

$$\sigma(s)_{1\times r} = \Big(\mathbf{Z}_s^\intercal - \frac{1}{m}\mathbf{1}_m^\intercal K\Big)\Big(\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\intercal\Big)\tilde{\mathbf{V}}, \tag{12}$$

where $\mathbf{Z}_s$ is of the form (10).

In addition we can represent directions of maximum variation of $\sigma(s)$ associated with the variable $X_k$ by projecting the tangent vector at $s = 0$. In matrix form, we have

$$\frac{d\sigma}{ds}\Big|_{s=0} = \frac{d\mathbf{Z}_s^\intercal}{ds}\Big|_{s=0}\Big(\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\intercal\Big)\tilde{\mathbf{V}} \tag{13}$$

with

$$\frac{d\mathbf{Z}_s^\intercal}{ds}\Big|_{s=0} = \Big(\frac{d\mathbf{Z}_s^1}{ds}\Big|_{s=0}, ..., \frac{d\mathbf{Z}_s^m}{ds}\Big|_{s=0}\Big)^\intercal$$

and, with

$$\begin{aligned}\frac{d\mathbf{Z}_s^i}{ds}\Big|_{s=0} &= \frac{dK(\mathbf{y},\mathbf{x}_i)}{ds}\Big|_{s=0} \\ &= \Big(\sum_{t=1}^{m}\frac{\partial K(\mathbf{y},\mathbf{x}_i)}{\partial y_t}\frac{dy_t}{ds}\Big)\Big|_{s=0} \\ &= \sum_{t=1}^{m}\frac{\partial K(\mathbf{y},\mathbf{x}_i)}{\partial y_t}\Big|_{\mathbf{y}=\mathbf{a}}\delta_t^k = \frac{\partial K(\mathbf{y},\mathbf{x}_i)}{\partial y_k}\Big|_{\mathbf{y}=\mathbf{a}}\end{aligned}$$

where $\delta_t^k$ denotes the delta of Kronecker. In particular, let us consider the radial basis kernel: $k(\mathbf{x},\mathbf{z}) = \exp(-c\,\|\mathbf{x}-\mathbf{z}\|^2)$ with $c>0$ a free parameter. Using above notation, we have

$$K(\mathbf{y},\mathbf{x}_i) = \exp(-c\,\|\mathbf{y}-\mathbf{x}_i\|^2) = \exp(-c\sum_{t=1}^{n}(y_i - x_{it})^2)$$

When we consider the set of points of the form $\mathbf{y} = \mathbf{a} + s\mathbf{e}_k \in \mathbb{R}^n$,

$$\begin{aligned}\frac{d\mathbf{Z}_s^i}{ds}\Big|_{s=0} &= \frac{\partial K(\mathbf{y},\mathbf{x}_i)}{\partial y_k}\Big|_{\mathbf{y}=\mathbf{a}} \\ &= -2cK(\mathbf{a},\mathbf{x}_i)(a_k - x_{ik})\end{aligned}$$

In addition, if $\mathbf{a} = \mathbf{x}_\beta$ (a training point) then

$$\frac{d\mathbf{Z}_s^i}{ds}\Big|_{s=0} = -2cK(\mathbf{x}_\beta,\mathbf{x}_i)(x_{\beta k} - x_{ik})$$

Thus, by applying equation (12) we can locally represent any given input variable in the KPCA plot. Moreover, by using equation (13) we can represent the tangent vector associated with any given input variable at each sample point. Therefore, we can plot a vector field over the KPCA that points to the growing directions of the given variable.

We used the radial basis kernel in our implementation however the procedure we have introduced is also applicable to any other smooth kernel, for instance the Linear kernel which supplies standard PCA analysis.

## 4. Validation

In this section we illustrate our procedure with data from the leukemia data set of Golub et al. (1999) and the lymphoma data set Alizadeh et al. (2000).

In these examples our aim is to validate our procedure for adding input variables information into KPCA representation. We follow the following steps. First, in each data set, we build a list of genes that are differentially expressed. This selection is based in accordance with previous studies such as (Golub et al. (1999), Pittelkow & Wilson (2003), Reverter et al. (2010)). In addition we compute the expression profile of each gene selected, this profile confirm the evidence of differential expression.

Second, we compute the curves through each sample point associated with each gene in the list. These curves are given by the $\phi$-image of points of the form:

$$\mathbf{y}(s) = \mathbf{x}_i + s\mathbf{e}_k$$

where $\mathbf{x}_i$ is the $1 \times n$ expression vector of the $i$-th sample, $i = 1, ..., m$, $k$ denotes the index in the expression matrix of the gene selected to be represented, $\mathbf{e}_k = (0, ..., 1, ..., 0)$ is a $1 \times n$ vector with zeros except in the $k$-th. These curves describe locally the change of the sample $x_i$ induced by the change of the gene expression.

Third, we project the tangent vector of each curve at $s = 0$, that is, at the sample points $\mathbf{x}_i$, $i = 1, ..., m$, onto the KPCA subspace spanned by the eigenvectors (9). This representation capture the direction of maximum variation induced in the samples when the expression of gene increases.

By simultaneously displaying both the samples and the gene information on the same plot it is possible both to visually detect genes which have similar profiles and to interpret this pattern by reference to the sample groups.

### 4.1 Leukemia data sets

The leukemia data set is composed of 3051 gene expressions in three classes of leukemia: 19 cases of B-cell acute lymphoblastic leukemia (ALL), 8 cases of T-cell ALL and 11 cases of acute myeloid leukemia (AML). Gene expression levels were measured using Affymetrix high-density oligonucleotide arrays.

The data were preprocessed according to the protocol described in Dudoit et al. (2002). In addition, we complete the preprocessing of the gene expression data with a microarray standardization and gene centring.

In this example we perform the KPCA , as detailed in the previous section, we compute the kernel matrix with using the radial basis kernel with $c = 0.01$, this value is set heuristically. The resulting plot is given in Figure 1. It shows the projection onto the two leading kernel principal components of microarrays. In this figure we can see that KPCA detect the group structure in reduced dimension. AML, T-cell ALL and B-cell ALL are fully separated by KPCA.

To validate our procedure we select a list of genes differentially expressed proposed by (Golub et al. (1999), Pittelkow & Wilson (2003), Reverter et al. (2010)) and a list of genes that are not differentially expressed. In particular, in Figures 2, 3, 4 and 5 we show the results in the case of genes: `X76223_s_at`, `X82240_rna1_at`, `Y00787_s_at` and `D50857_at`, respectively. The three first genes belong to the list of genes differentially expressed and the last gene is not differentially expressed.

Figure 2 (top) shows the tangent vectors associated with `X76223_s_at` gene, attached at each sample point. This vector field reveals upper expression towards T-cell cluster as is expected from references above mentioned. This gene is well represented by the second principal component. The length of the arrows indicate the strength of the gene on the sample position despite the dimension reduction. Figure 2 (bottom) shows the expression profile of

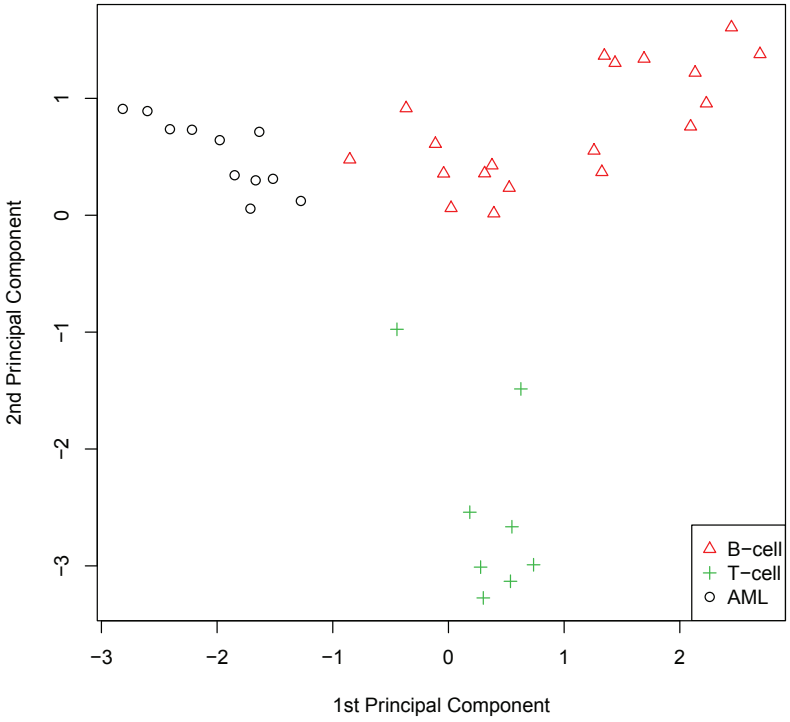



Fig. 1. Kernel PCA of Leukemia dataset.

X76223_s_at gene. We can observe that X76223_s_at gene is up regulated in T-cell class. This profile is agree with our procedure because the direction in which the expression of the X76223_s_at gene increases points to the T-cell cluster.

Figure 3 (top) shows the tangent vectors associated with X82240_rna1_at gene attached at each sample point. This vector field reveals upper expression towards B-cell cluster as is expected from references above mentioned. Figure 3 (bottom) shows the expression profile of X82240_rna1_at gene. We can observe that X82240_rna1_at gene is up regulated in B-cell class. This profile is agree with our procedure because the direction in which the expression of the X82240_rna1_at gene increases points to the B-cell cluster.

Figure 4 (top) shows the tangent vectors associated with Y00787_s_at gene attached at each sample point. This vector field reveals upper expression towards AML cluster as is expected from references above mentioned. Figure 4 (bottom) shows the expression profile of Y00787_s_at gene. We can observe that Y00787_s_at gene is up regulated in AML class. This profile is agree with our procedure because the direction in which the expression of the Y00787_s_at gene increases points to the AML cluster.

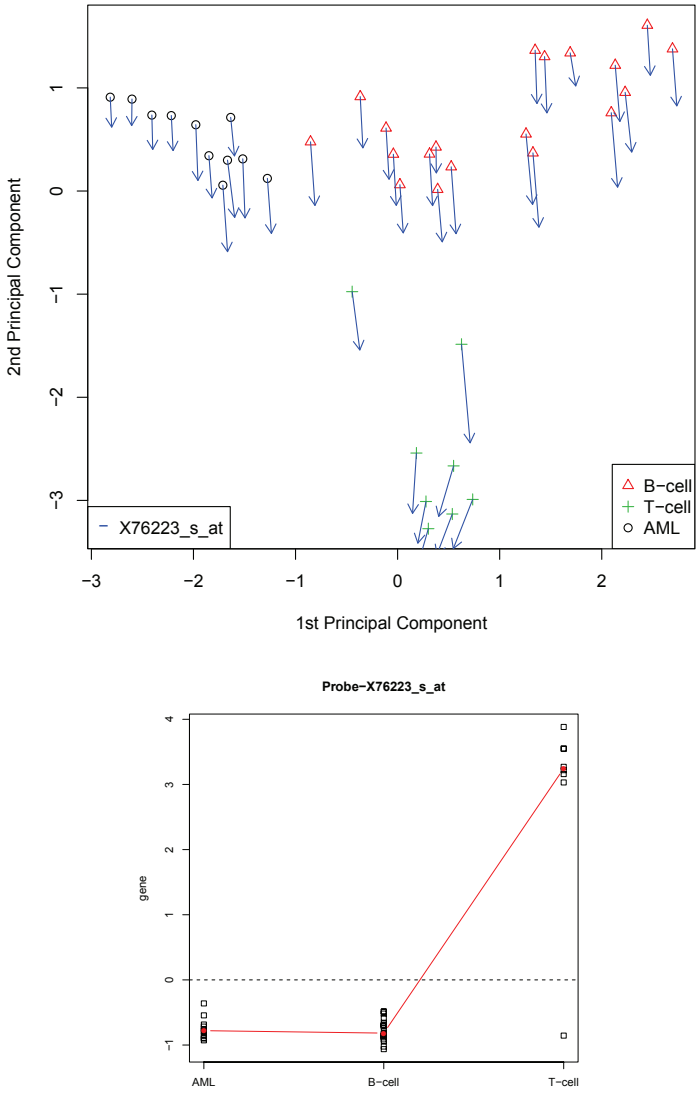



Fig. 2. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with X76223-s-at gene at each sample point. Vector field reveals upper expression towards T-cell cluster. (Bottom) Expression profile of X76223-s-at gene confirms KPCA plot enriched with tangent vectors representation.

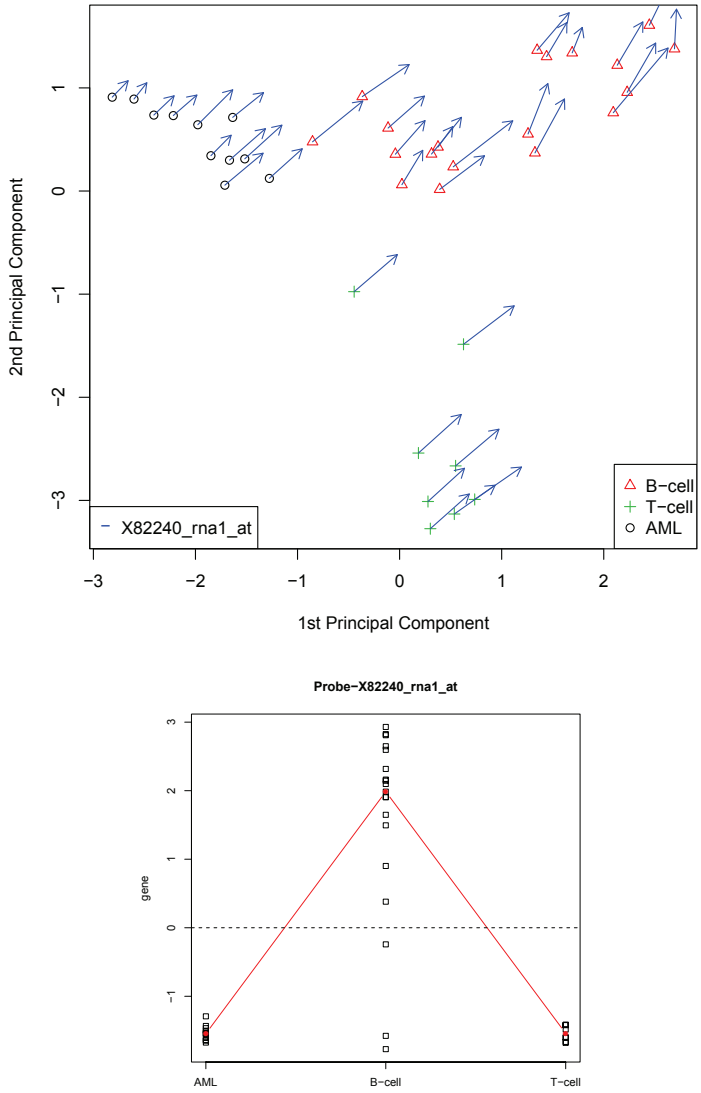



Fig. 3. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with X82240-rna1-at gene at each sample point. Vector field reveals upper expression towards B-cell cluster. (Bottom) Expression profile of X82240-rna1-at gene confirms KPCA plot enriched with tangent vectors representation.

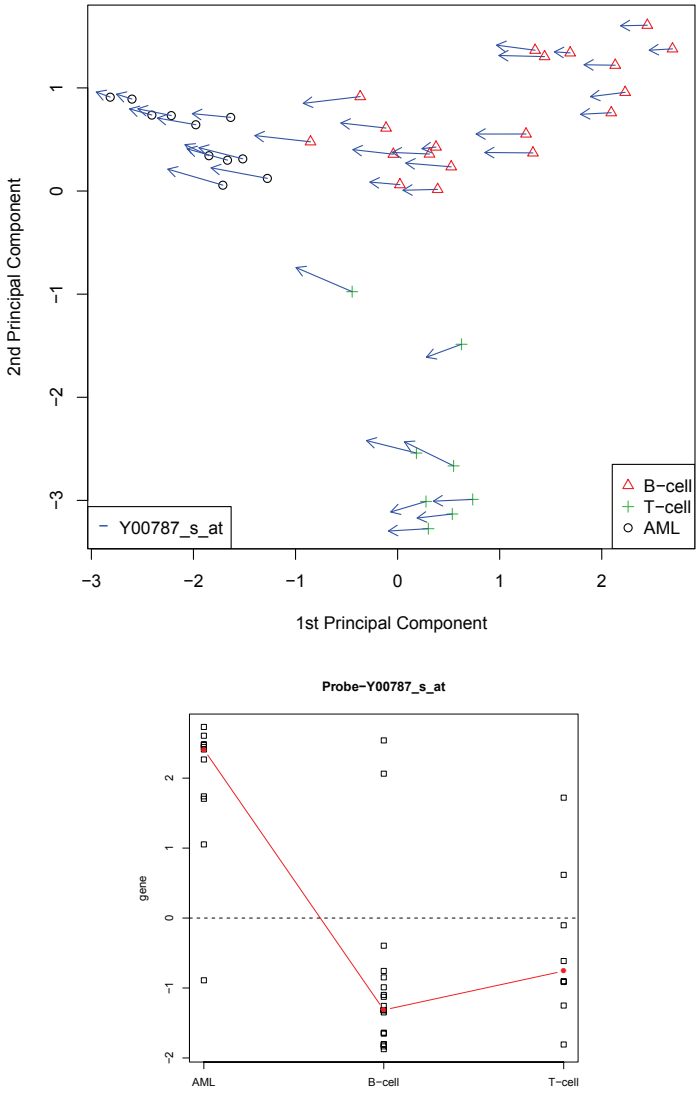



Fig. 4. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with Y00787-sat gene at each sample point. Vector field reveals upper expression towards AML cluster. (Bottom) Expression profile of Y00787-sat gene confirms KPCA plot enriched with tangent vectors representation..

Figure 5 (top) shows the tangent vectors associated with D50857_at gene attached at each sample point. This vector field shows no preferred direction to any of the three cell groups. The arrows are of short length and variable direction in comparison with other genes showed in previous Figures. Figure 5 (bottom) shows a flat expression profile of D50857_at gene. This profile is agree with our procedure because any direction of expression of the D50857_at gene is highlighted.

### 4.2 Lymphoma data sets

The lymphoma data set comes from a study of gene expression of three prevalent lymphoid malignancies: B-cell chronic lymphocytic leukemia (B-CLL), follicular lymphoma (FL) and diffuse large B-cell lymphoma (DLCL). Among 96 samples we took 62 samples 4026 genes in three classes: 11 cases of B-CLL, 9 cases of FL and 42 cases of DLCL. Gene expression levels were measured using 2-channel cDNA microarrays.

After preprocessing, all gene expression profiles were base 10 log-transformed and, in order to prevent single arrays from dominating the analysis, standardized to zero mean and unit variance. Finally, we complete the preprocessing of the gene expression data with gene centring.

In this example we perform the KPCA , as detailed in the previous section, we compute the kernel matrix with using the radial basis kernel with $c = 0.01$, this value is set heuristically. The resulting plot is given in Figure 6. It shows the projection onto the two leading kernel principal components of microarrays. In this figure we can see that KPCA detect the group structure in reduced dimension. DLCL, FL and B-CLL are fully separated by KPCA.

To validate our procedure we select a list of genes differentially expressed proposed by (Reverter et al. (2010)) and a list of genes that are not differentially expressed. In particular, in Figures 7, 8, 9 and 10 we show the results in the case of genes: 139009, 1319066, 1352822 and 1338456, respectively. The three first genes belong to the list of genes differentially expressed and the last gene is not differentially expressed.

Figure 7 (top) shows the tangent vectors associated with 139009 gene attached at each sample point. This vector field reveals upper expression towards DLCL cluster as is expected from references above mentioned. This gene is mainly represented by the first principal component. The length of the arrows indicate the influence strength of the gene on the sample position despite the dimension reduction. Figure 7 (bottom) shows the expression profile of 139009 gene. We can observe that 139009 gene is up regulated in DLCL cluster. This profile is agree with our procedure because the direction in which the expression of the 139009 gene increases points to the DLCL cluster.

Figure 8 (top) shows the tangent vectors associated with 1319066 gene attached at each sample point. This vector field reveals upper expression towards FL cluster as is expected from references above mentioned. This gene is mainly represented by the second principal component. Figure 8 (bottom) shows the expression profile of 1319066 gene. We can observe that 1319066 gene is up regulated in FL class. This profile is agree with our procedure because the direction in which the expression of the 1319066 gene points to the FL cluster.

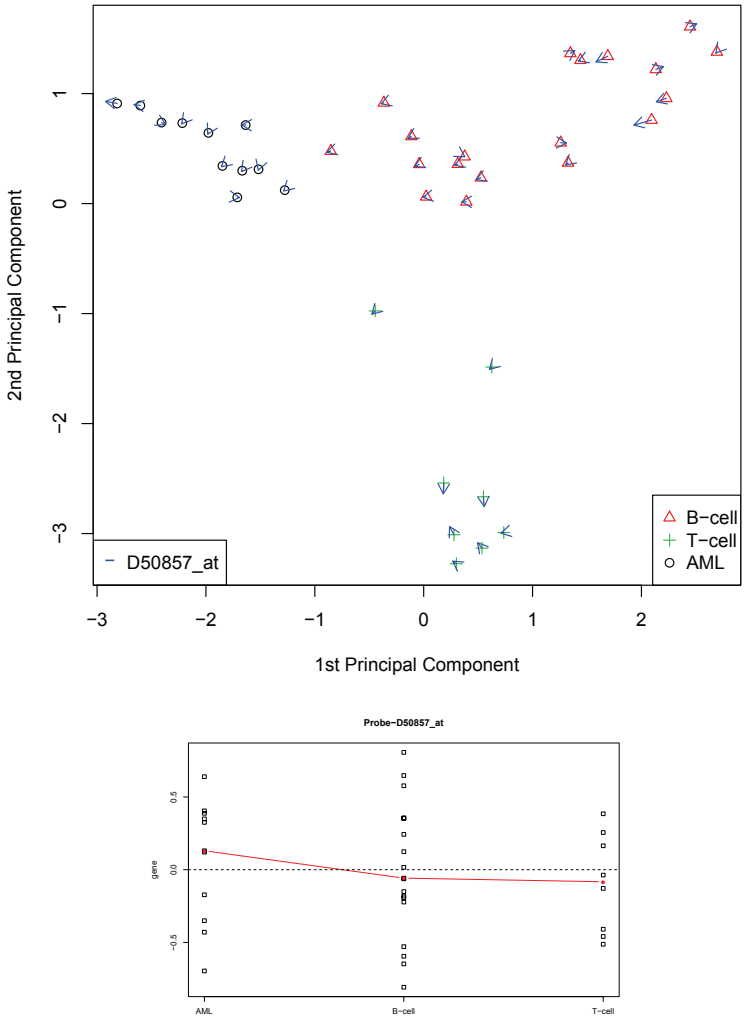



Fig. 5. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with D50857-at gene at each sample point. Vector field shows no preferred direction. (Bottom) Flat Expression profile of D50857-at gene confirms KPCA plot enriched with tangent vectors representation.

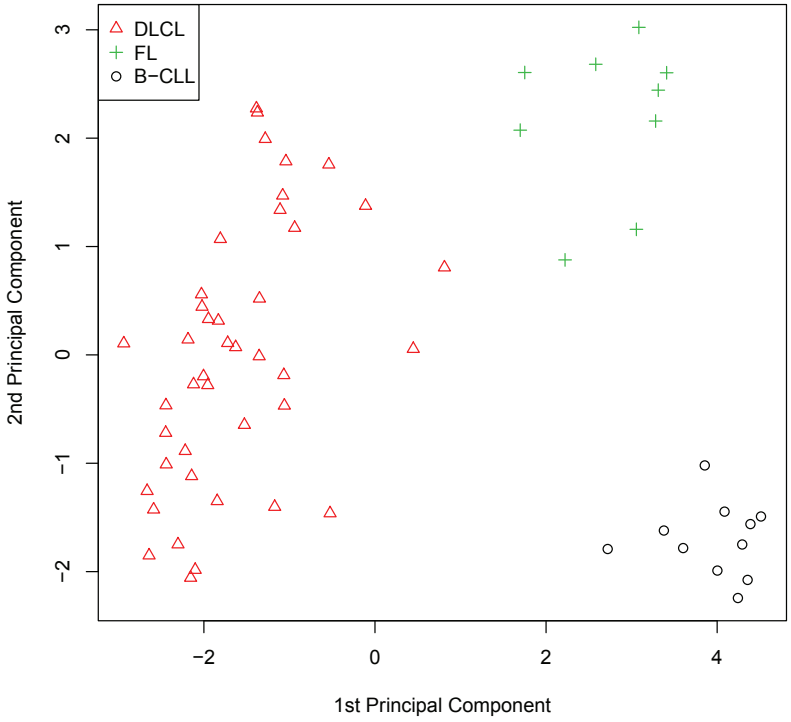



Fig. 6. Kernel PCA of Lymphoma dataset.

Figure 9 (top) shows the tangent vectors associated with `1352822` gene attached at each sample point. This vector field reveals upper expression towards B-CLL as is expected from references above mentioned. Figure 9 (bottom) shows the expression profile of `1352822` gene. We can observe that `1352822` gene is up regulated in B-CLL class. This profile is agree with our procedure because the direction in which the expression of the `1352822` gene increases points to the B-CLL cluster.

Figure 10 (top) shows the tangent vectors associated with `1338456` gene attached at each sample point. This vector field shows no preferred direction to any of the three cell groups. The arrows are of short length and variable direction in comparison with other genes showed in previous Figures. Figure 10 (bottom) shows a flat expression profile of `1338456` gene. This profile is agree with our procedure because any direction of expression of the `1338456` gene is highlighted.

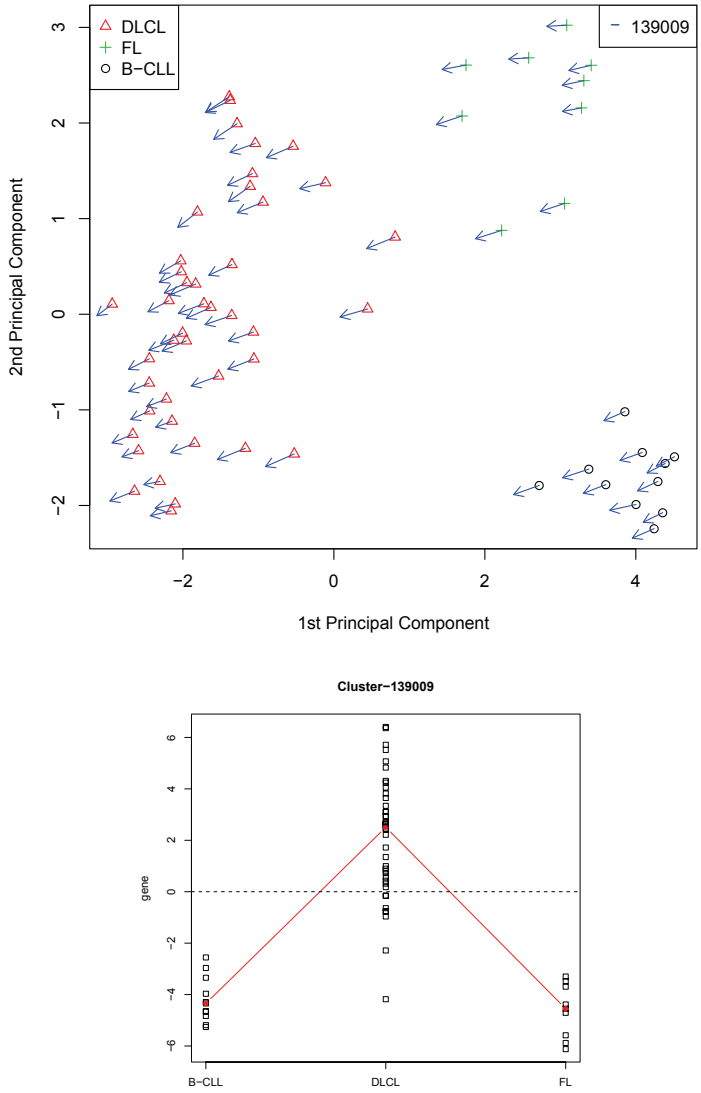



Fig. 7. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with 139009 gene at each sample point. Vector field reveals upper expression towards DLCL cluster. (Bottom) Expression profile of 139009 gene confirms KPCA plot enriched with tangent vectors representation.

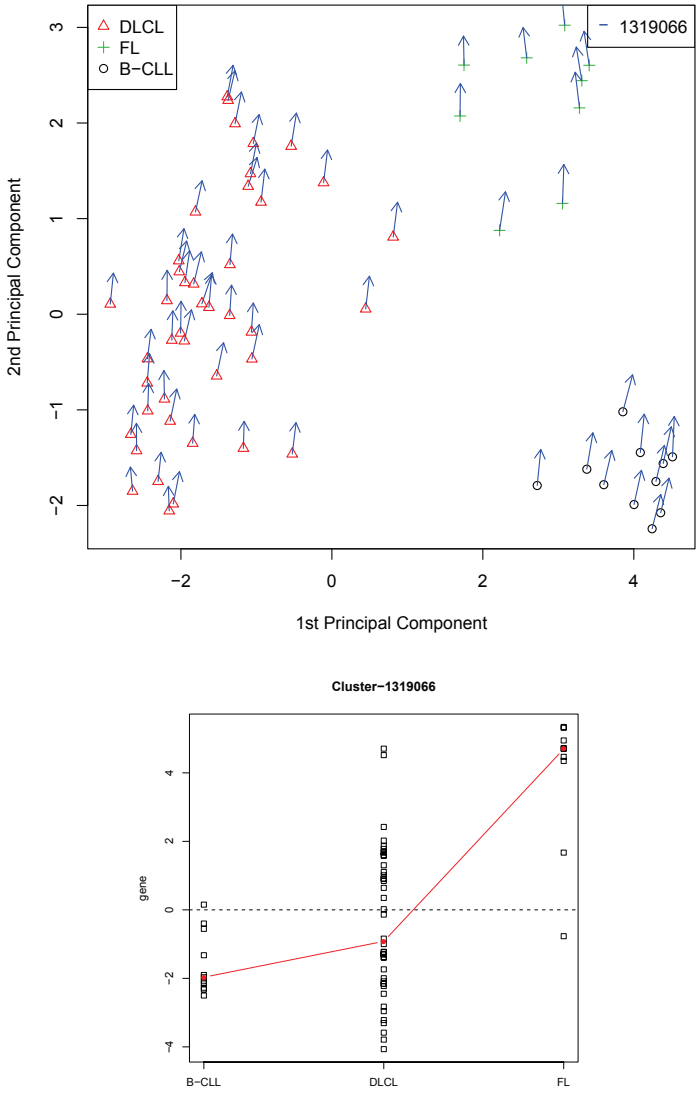



Fig. 8. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with 1319066 gene at each sample point. Vector field reveals upper expression towards FL cluster. (Bottom) Expression profile of 1319066 gene confirms KPCA plot enriched with tangent vectors representation.

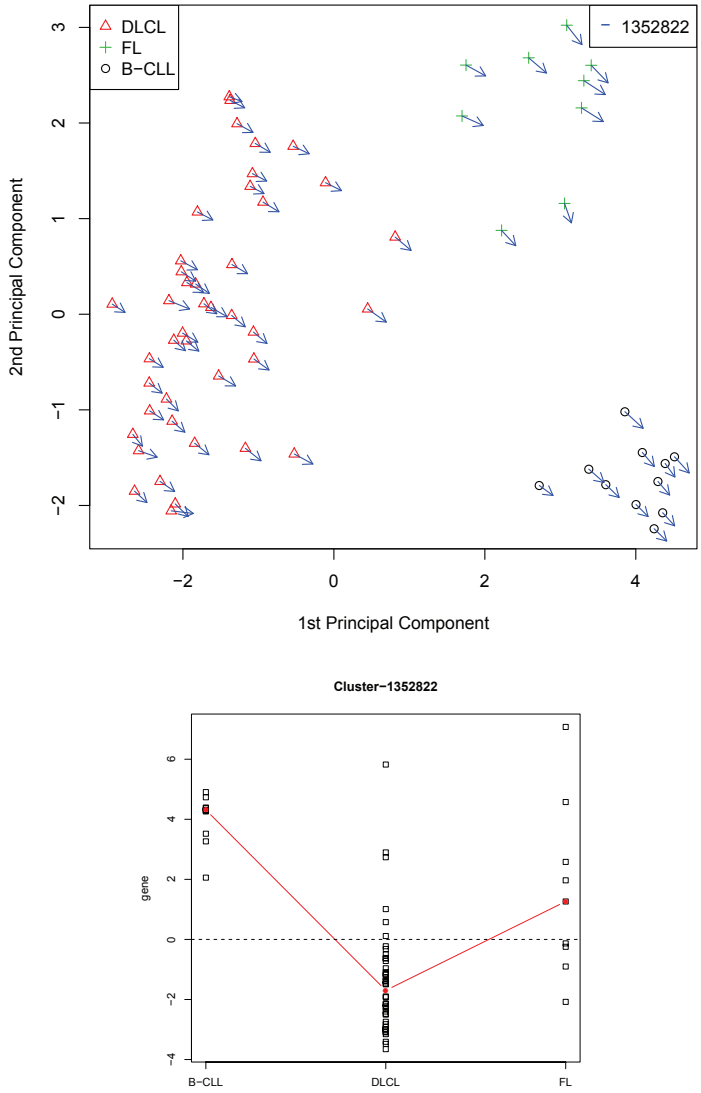



Fig. 9. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with 1352822 gene at each sample point. Vector field reveals upper expression towards B-CLL cluster. (Bottom) Expression profile of 1352822 gene confirms KPCA plot enriched with tangent vectors representation.

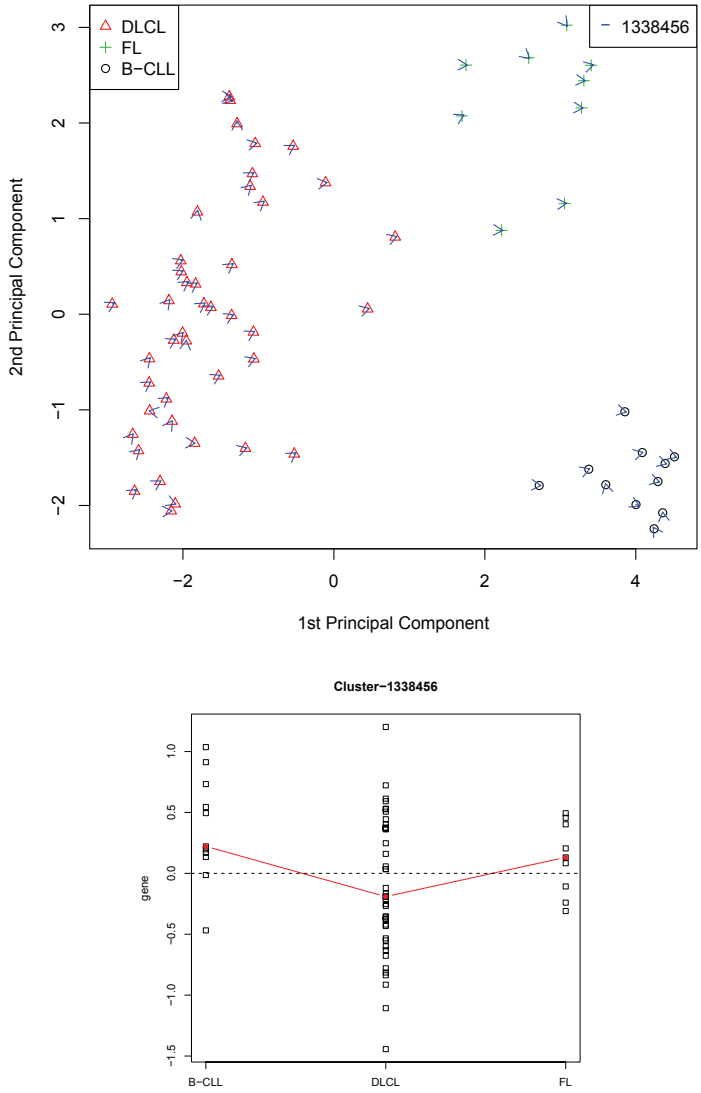



Fig. 10. (Top) Kernel PCA of Leukemia dataset and tangent vectors associated with 1338456 gene at each sample point. Vector field shows no preferred direction. (Bottom) Flat expression profile of 1338456 gene confirms KPCA plot enriched with tangent vectors representation.

## 5. Conclusion

In this paper we propose an exploratory method based on Kernel PCA for elucidating relationships between samples (microarrays) and variables (genes). Our approach show two main properties: extraction of nonlinear features together with the preservation of the input variables (genes) in the output display. The method described here is easy to implement and facilitates the identification of genes which have a similar or reversed profiles. Our results indicate that enrich the KPCA with supplementary input variable information is complementary to other tools currently used for finding gene expression profiles, with the advantage that it can capture the usual nonlinear nature of microarray data.

## 6. References

Alizadeh, A.A.; Eisen, M.B.; Davis, R,E.; Ma, C.; Lossos, I.S.; Rosenwald, A.; Boldrick, J.C.; Sabet, H.; Tran, T.; Yu, X.; Powell, J.I.; Yang, L.; Marti, G.E.; Moore, T.; Hudson, J.J.; Lu, L.; Lewis, D.B.; Tibshirani, R.; Sherlock, G.; Chan, WC.; Greiner, T.C.; Weisenburger, D.D.; Armitage, J.O.; Warnke, R.; Levy, R.; Wilson, W.; Grever, M.R.; Byrd, J.C; Bostein, D.; Brown, P,O. & Staudt, LM. (2000). Different type of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511.

Alter, O.; Brown, P,O. & Botstein, D. (2000). Singular value decomposition for genome-wide expression data processing and modeling. *Proc. Natl. Acad. Sci*. USA. 97(18), 10101-10106.

Chambers, J.M.; Cleveland, W.S.; Kleiner, B. & Tuckey, P.A. (1983) *Graphical Methods for Data Analysis statistics/probabilit*y. Wadsworth.

Chu, W.; Ghahramani, Z.; Falciani. F. & Wild, D: (2005) Biomarker discovery in microarrays gene expression data with Gaussian processes. *Bioinformatcis*. 21(16), 3385–3393.

Dettling, M. (2004). BagBoosting for tumor classificcation with gene expression data. *Bioinformatcis*. 20(18), 3583–3593.

Diaz-Uriarte, R. & Andres, S.A. (2006) Gene selection and classification of microarray data using random forest. *BMC Bioinformatcis*. 7:3, 1–13.

Dudoit, S.; Fridlyand, J. & Speed, T,P. (2002). Comparison of discrimination methods for the classification of tumours using gene expression data. *J. Am. Statist. Soc*. 97:77–87.

Eisen, M.B.; Spellman, P,T.; Brown, P.O. & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci*. USA. 95(25):14863–14868.

Fellenberg, K.; Hauser, N.C.; Brors, B.; Neutzner, A. & Hoheisel, J. (2001): Correspondence analysis applied to microarray data. *Proc. Natl. Acad. Sci*. USA. 98(19) 10781–10786.

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.; Bloomfield, C.D. & Lander, E.S. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression profiling. *Science*. 286(5439):531–537.

Park, M.; Lee, J.W.; Lee, J.B. & Song, S.H. (2008) Several biplot methods applied to gene expression data. *Journal of Statistical Planning and Inference*. 138:500–515.

Pittelkow, Y.E. & Wilson, S.R. (2003). Visualisation of Gene Expression Data - the GE-biplot, the Chip-plot and the Gene-plot. *Statistical Applications in Genetics and Molecular Biology*. Vol. 2. Issue 1. Article 6.

Reverter, F.; Vegas, E. & Sanchez, P. (2010) Mining Gene Expressions Profiles: An integrated implementation of Kernel Principal Components Analysis and Singular Value Decomposition. *Genomics, Proteomics and Bioinformatics*. 8(3):200–210.

Shawe-Taylor, J. & Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press.

Scholkopf, B.; Smola, A.J. & Muller, K.R. (1998) Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*. 10:1299-1319.

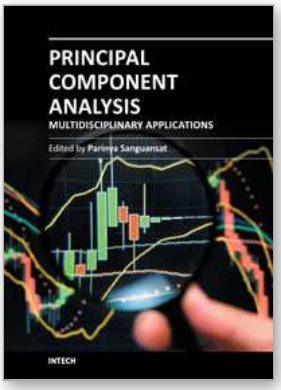
Scholkopf, B.; Smola, A.J. (2002). *Learning with Kernels - Support Vector Machines, Regularization, Optimization and Beyond*. Cambridge, MA. MIT Press.

Tamayo, P.; Solni, D.; Mesirov, J.; Zhu, Q.; Kitareewan, S.; Dmitrovsky, E.; Lander, E.S. & Golub, T.R. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proc. Natl. Acad. Sci*. USA. 96(6):2907–2912.

Xuehua, L. & Lan, S. (2009). Kernel based nonlinear dimensionality reduction for microarray gene expression data analysis. *Expert Systems with Applications* 36:7644-7650.

Zhao, X. & Cheung, L.W.K. (2007) Kernel-Imbedded Gaussian processes for disease classification using microarrays gene expression data. *BMC Bioinformatcis*. 8:67:1–26.

Zhenqiu, L., Dechang, C. & Halima B. (2005). Clustering gene expression data with kernel principal components. *Journal of Bioinformatics and Computational Biology*. 3(2):303–316.




**Principal Component Analysis - Multidisciplinary Applications**

Edited by Dr. Parinya Sanguansat

ISBN 978-953-51-0129-1
Hard cover, 212 pages
**Publisher** InTech
**Published online** 29, February, 2012
**Published in print edition** February, 2012

This book is aimed at raising awareness of researchers, scientists and engineers on the benefits of Principal Component Analysis (PCA) in data analysis. In this book, the reader will find the applications of PCA in fields such as taxonomy, biology, pharmacy,finance, agriculture, ecology, health and architecture.



**InTech Europe**
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

**InTech China**
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821