Deep kernel dimensionality reduction for scalable data integration

Nataliya Sokolovska, Karine Clément, Jean-Daniel Zucker

Deep Kernel Dimensionality Reduction for Scalable Data Integration

Nataliya Sokolovska[a,b,c], Karine Clément[a,b,c], Jean-Daniel Zucker[a,b,d]

[a] *Institute of Cardiometabolism and Nutrition, ICAN, Assistance Publique Hôpitaux de Paris, Pitié-Salpêtrière Hospital, Paris, France*
[b] *Sorbonne Universités, UPMC University Paris 6, UMR_S 1166, ICAN, NutriOmics Team, Paris, France*
[c] *INSERM, UMR S U1166, NutriOmics Team, Paris, France*
[d] *Research Institute for Development, UMI 209, UMMISCO, Bondy, France*

Abstract

Dimensionality reduction is used to preserve significant properties of data in a low-dimensional space. In particular, data representation in a lower dimension is needed in applications, where information comes from multiple high dimensional sources. Data integration, however, is a challenge in itself.

In this contribution, we consider a general framework to perform dimensionality reduction taking into account that data are heterogeneous. We propose a novel approach, called Deep Kernel Dimensionality Reduction which is designed for learning layers of new compact data representations simultaneously. The method can be also used to learn shared representations between modalities. We show by experiments on standard and on real large-scale biomedical data sets that the proposed method embeds data in a new compact meaningful representation, and leads to a lower classification error compared to the state-of-the-art methods.

Keywords: Dimensionality reduction, heterogeneous data integration

1. Introduction

Data integration is a challenging task with an ambitious goal to increase performance of supervised learning, since various sources of data tend to contain different parts of information about the problem.

Structure learning and data integration allow to better understand the properties and content of biological data in general and of "omics" data

in particular. Combining complementary pieces issued from different data sources is likely to provide more knowledge, since distinct types of data provide distinct views of the molecular machinery of cells. Medical and biological knowledge can be naturally organized into hierarchies: symptoms of diseases are observed and pathological states on all levels of omics data are hidden. Hierarchical structures and data integration methods reveal dependencies that exist between cellular components and help to understand the biological network structure.

Graphical models follow a natural organization and representation of data, and are a promising method of simultaneous heterogeneous data processing. Hidden variables in a graphical hierarchical model can efficiently agglomerate information of observed instances via dimensionality reduction, since fewer latent variables are able to summarize multiple features. However, integration of latent variables is a crucial step of modeling.

Multi-modal learning, heterogeneous data fusion, or data integration, involves relating information of different nature. In biological and medical applications, data coming from one source are already high-dimensional. Hence, data integration increases the dimensionality of a problem even more, and some feature selection or dimensionality reduction procedure is absolutely needed both to make the computations tractable and to obtain a model which is compact and easily interpretable.

Our goal is to develop an efficient dimensionality reduction approach which will design a compact model. The method needs to be scalable, to fusion heterogeneous data, and be able to reach a better generalizing performance compared to a full model and to state-of-the-art methods. Another important question is whether introducing data of different nature have a positive effect, and provides additional knowledge.

In this contribution, we propose a deep dimensionality reduction approach which agglomerates original features from a high-dimensional space and creates a hierarchy of new representations. To construct the hidden layers of the proposed deep learning framework, we introduce a deep kernel dimensionality reduction method, and we compare its performance to some standard clustering and dimensionality reduction methods.

The biomedical problem of our interest is a real problem which is a binary classification of obese patients. The aim is to stratify patients in order to choose an efficient appropriate personalized medical treatment. The task is motivated by a recent French study [1] of gene-environment interactions carried out to understand the development of obesity. It was re-

ported that the gut microbial gene richness can influence the outcome of a dietary intervention. A quantitative metagenomic analysis stratified patients into two groups: group with low gene gut flora count (LGC) and high gene gut flora count (HGC) group. The LGC individuals have a higher insulin-resistance and low-grade inflammation, and therefore the gene richness is strongly associated with obesity-driven diseases. The individuals from a low gene count group seemed to have an increased risk to develop obesity-related cardiometabolic risk compared to the patients from the high gene count group. It was shown [1] that a particular diet is able to increase the gene richness: an increase of genes was observed with the LGC patients after a 6-weeks energy-restricted diet. A similar study with Dutch individuals was conducted by [2], and made a similar conclusion: there is a hope that a diet can be used to induce a permanent change of gut flora, and that treatment should be phenotype-specific. There is therefore a need to go deeper into these biomedical results and to identify candidate biomarkers associated with cardiometabolic disease (CMD) risk factors and with different stages of CMD evolution.

Our contribution is multi-fold:

- we introduce a novel kernel-based deep dimensionality reduction method which constructs layers of a deep structure simultaneously,
- we illustrate that the proposed framework is efficient on standard data sets and on a real original rich heterogeneous MicrObese data set [1], which contains meta-data, i.e., clinical parameters and alimentary patterns of patients, gene expressions of adipose tissue, and gene abundance of gut flora. We efficiently learn new data representations structured into a multi-level hierarchy. We evaluate the prediction power of the models with the reduced dimensionality showing that the proposed approach outperforms the state-of-the-art dimensionality reduction methods.

The paper is organized as follows. Section 2 considers the related work and the state-of-the-art data integration and dimensionality reduction methods. We introduce our approach in Section 3. We show the results of our experiments in Sections 4 and 5. Concluding remarks and perspectives close the paper.

2. Related Work

We tackle a complex problem which consists of a data integration task and a dimensionality reduction procedure. In this section, we consider some state-of-the-art data fusion methods, dimensionality reduction approaches, and some recent attempts to combine both within a hierarchical model. The literature on clustering and dimensionality reduction is quite rich; publications on heterogeneous data integration, on the contrary, are not so numerous.

The state-of-the-art data integration methods are traditionally divided into four categories: functional linkage networks, vector subspace integration, kernel fusion methods, and ensemble methods. Graphical models (functional linkage networks) are based on graphical representation of nodes and relations between variables of interest, e.g., Bayesian networks. Vector space integration is a method where data from various sources are concatenated in a vector. Kernel methods for data integration are motivated by the fact that variables with similar functions share expression patterns. Kernel functions are used to define similarities between the variables of interest. Recently [3] reported that ensemble methods, that have been ignored for a long time, are a competitive data integration approach. Ensemble methods combine outputs produced by different classifiers trained on various data sets, or data views; they are known to be scalable, and data of different formats can be easily integrated, since the data integration is done at the decision level. Our framework, introduced in the next section, is a graphical framework and incorporates a vector concatenation of heterogeneous data, a similarity matrix base on a kernel function, and can also embed an ensemble method.

The idea to use a hierarchy for biomedical data is not new. So, Bayesian networks are still often used in systems biology. They model a joint probability distribution, parameterized by a parameter θ over all nodes. More specifically, the Bayesian networks define a joint probability distribution $P(x;\theta) = \prod_{i=1}^{n} P(x_i|x_{PA_i})$, where PA stands for "parent". E.g., [4] considers a linear model, where observed and hidden variables follow $x_i = \sum_{j \in PA_i} \alpha_{ij} h_j + \epsilon_i$, where x are observed and h are hidden. To estimate the vector of parameters α of the model, the hidden variables are integrated out. The problem that is typical for Bayesian networks with hidden variables, is the identifiability problem. It has shown by [4] that the model is identifiable under certain conditions, and that the discovery of latent layers, i.e. the structure estimation does not result in many models which describe the data. Note that a

high complexity issue is still an open problem for probabilistic models with hidden variables.

The problem of hierarchical protein function annotation, where the simplest method is to annotate each instance independently, is considered by [5]. However, the output can be composed of terms which are inconsistent with one another. An SVM was used for individual predictions. A proposition to combine the SVM predictions in a naive Bayes network, and to perform a hierarchical correction of the SVM outputs, comes from [6]. Unfortunately, the kernel (SVM) data fusion approaches have in general poor scalability properties, since the method operates with matrices whose size equals to a squared number of observations which can be very big.

Another recent hierarchical graphical model based on intuition that latent variables can synthesize the information and can lead to easily interpretable models, was proposed by [7]. The optimization in the hierarchical model is done using the Expectation-Maximization (EM) algorithm [8, 9]. The parameters estimated by an EM are means and variances of the hidden variables. A family of hierarchical latent class models, where optimization is also done with the EM, was introduced in [10]. In such a graphical model, the leaves are variables of interest, and the latent variables are generalizing or agglomerating nodes. The number of layers of latent variables can be unlimited but it is reported that the higher the generalization level, the less information the nodes contain. E.g., for genetic association studies, the optimal number of levels of latent parameters equals two or three. Latent variables are also used to reduce the dimensionality of a problem. A similar idea is considered in [11], where edges in a graphical model stand for mutations, and generalizing hidden layers can be interpreted as ancestral haplotypes.

Another idea is to exploit clustering for dimensionality reduction. A scheme for clustering simultaneously of rows and columns of contingency tables was proposed by [12]. The major idea is that if parameters are tied into clusters of "high quality", then a better prediction can be obtained. An intuition behind is that clustering variables can reduce noise. The approach takes pairwise interactions between variables into consideration, and an objective function is optimized locally. The clusters are constructed using the mutual information between variables. A discrete problem was considered by [12], where clustering is based on mutual information. The mutual information can be computed directly $I(x,y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$. To choose a cluster for a given x, it is proposed to maximize $\sum_{e_{ij}} w_{ij} I(x_i, x_j)$, where w

are weights, and e are edges in the considered graph.

Clustering data using kernels and general measures have been discussed by [13]. Clustering can be seen as a mixture model $p = \sum_{k=1}^{K} \pi_k P_k$, where π_k is a cluster weight and P_k is a component distribution. Clusters can be separated based on the distance between the clusters means, or clusters variances which are also distance functions between distributions. Data separation can be also based on higher-order criteria, and not only on point locations. The maximum mean discrepancy

$$MMD(P_1, P_2) = MMD(\hat{P}_1, \hat{P}_2) = \left\| \frac{1}{n} \sum_{i=1}^{n} K(x_i, \cdot) - \frac{1}{m} \sum_{j=1}^{m} K(x_j, \cdot) \right\|_{\mathcal{H}} \quad (1)$$

was introduced by [14] as a more general distance function. It was reported [13, 14] that maximisation of a criterion based on the regularized MMD term (eq. 1) provides with a very reasonable clustering. A large MMD corresponds to a low Bayes risk, in other words, to the situation where the clusters are well-separated. The disadvantage of the approach is that it deals with two-sample problems, and a generalisation to more than two clusters is not obvious.

Dimensionality reduction is crucial not only for the computational issues but also for data visualization in a two- or three-dimensional space. So, [15] compares a number of unsupervised dimensionality reduction methods applied to visualization of microarrays. Below we mention briefly some efficient standard approaches which we use further in our experiments.

Principal Component Analysis (PCA) [16, 17] is a linear approach to map high-dimensional data into its low-dimensional representation. PCA chooses the coordinates which maximize the variance in the data, and, therefore, the principal components explain most of the variance. Kernel PCA [18] was developed to suite for nonlinear data, and, being a kernel method, it maps the data into a higher dimensional space before applying PCA.

A number of extensions of PCA appeared recently. Here we mention some of them. An assumption that an input signal can be represented by a sparse linear model is made by [19]. Dimensionality reduction in the case of the sparse linear model where $x \in \mathbb{R}^n$, $x = Da + \epsilon$, D is a dictionary, takes the following form:

$$\min_a \frac{1}{\sigma^2} \|x - Da\|_2{}^2 + \frac{1}{\tau} \|a\|_1, \quad (2)$$

what is equivalent to the kernel PCA with the kernel DD^T. If we denote $y = Lx$, where $y \in \mathbb{R}^m$, $m < n$, then for two samples, the model estimation is minimisation of the expectation

$$\min_{L_{m\times n}} E_{x_1,x_2,a_1,a_2}(y_1^T y_2 - a_1^T a_2)^2, \tag{3}$$

what can be very computationally expensive. As already mentioned, the problem burns down to a kernel PCA with a linear kernel. We compare our approach to the standard PCA and also to a KPCA in the experimental section.

An efficient extension of PCA which incorporates both sparsity and structure was introduced by [20]. The approach is based on structured regularization. The structured sparsity integrates higher-order prior information of data structure, compared to classical L_1-based sparse priors which perform feature selection without taking any structure into consideration.

Isomap [21, 22] is a non-linear method which constructs a neighborhood graph weighted by shortest distances between nearest neighbors. The low-dimensional space is constructed by minimization of pairwise distances between all nodes of the graph. Laplacian Eigenmaps [23, 24] is a local approach which builds a graph where the edges are weighted by values from the Gaussian kernel function, and the weighted distances between the nodes are minimized. The Laplacian eigenmaps incorporate cluster assumption, and enforce natural clusters in the data. Although a number of linear and non-linear dimensionality reduction methods have been recently proposed, it is still not clear how these approaches take the underlying data structure into consideration.

Finally, approaches that are very close in some sense to our contribution, are introduced in [25] and [26]. Multimodal deep learning [26] was proposed to learn features over multiple modalities, where sparse restricted Boltzmann machines are used to model the probability distribution over observed and hidden variables. In our work, we also consider shared representation learning of heterogeneous data, what corresponds to the "mid-level" in [26]. Hinton in [25] proposes to carry out dimensionality reduction with neural networks. Both approaches build structures that are similar to ours, however, both [26] and [25] make use of parametric models, where a distribution over all variables have to be estimated. In our approach, on the contrary, no effort is wasted for modeling distributions.

Multiple kernels are of great interest if learning problems involve multiple,

or heterogeneous data. A multiple kernel learning paradigm, proposed by [27], states that we are given m matrices $K_j \in \mathbb{R}^{n \times n}$. The matrices are symmetric, positive, and semidefinite. The goal is to find the best linear combination $\sum_{j=1}^{m} \eta_j K_j$, with $\eta_j \geq 0$ and a constraint $\sum_{j=1}^{m} \eta_j \mathrm{tr} K_j = c$, $c > 0$. The learning is done with sequential optimisation techniques.

In [28] an idea of hierarchical multiple kernels is explored. A kernel can be decomposed into a sum of individual basis kernels which can be represented as a directed acyclic graph. The framework is of particular interest for non linear variable selection. Both state-of-the-art methods [27] and [28] are quite efficient, however, the resulting models are hardly interpretable. It is also important to choose an optimal kernel for a good functioning, and, as we have seen on our data, in a case where $n \ll p$, such complex models tend to overfit.

3. Deep Dimensionality Reduction

Our goal is to reduce the dimension of the problem, in other words, to reduce the level of details without degrading predictive performance. In this section, we introduce a deep data integration framework which performs dimensionality reduction by constructing a multi-level hierarchy of new, more compact, data representations.

To learn the hierarchical model, a training algorithm has access to n i.i.d. labeled pairs $(X_i, Y_i)_{1 \leq i \leq n}$. The input variable or covariate is $X \in \mathcal{X}$, and the class variable is $Y \in \mathcal{Y}$. The covariate variables are high-dimensional, and $X_i = (X_{i,1}, \ldots, X_{i,d})$, where d is the dimensionality of the problem. We are interested, in particular, to perform a dimensionality reduction so that the dimensionality of our problem becomes $r \ll d$, and so that we can carry out a classification task on a much more compact, and probably less noisy, feature space.

3.1. Framework

The framework we consider here is a multi-level hierarchical structure. It is a tree, where the leaves are initial features extracted from a corpus. Nodes of all other layers (latent layers organized in a hierarchy) are new data representations, where an upper level is obtained from a lower one. The dimensionality reduction is done as follows: each hidden layer encodes a new representation of a layer underneath, and, the higher the level, the higher the abstraction level.

As we mentioned above, the graph is a tree which is constructed by a bottom-up technique. The number of hidden layers can be completely arbitrary; intuitively, the bigger dimensionality of an initial problem, the deeper the structure. However, note that the amount of information can potentially decrease with each level [10]. We suppose that the number of layers is task-dependent. We provide some discussion and intuition on it in Sections 4 and 5, where we describe our experimental results. In general, to perform dimensionality reduction according to the framework proposed, we can apply any state-of-the-art clustering or any dimensionality reduction approach which seems to suit well to the data being processed.

We can imagine two scenarios to reduce the dimensionality of the problem while using variables from all data sources available. We can carry out the dimensionality reduction separately for each type of data, and then combine the new compact representations in a new data set which we will use further for a prediction task. Another possibility is a multimodal fusion, where the hidden variables of all levels are constructed from instances coming from various data sources. In our experiments in Section 5 we test both scenarios, and it turns out that data integration based on the multimodal fusion is more efficient than learning new representations on separate data sets. However, we cannot claim that the multimodal fusion is always better, since optimal data integration can be data dependent. Another point is interpretability of a hierarchy. The results of the multimodal approach can be more difficult to interpret, but at the same time, they can provide more insights into new hypotheses and relations between data sources.

Below, we propose a novel supervised kernel dimensionality reduction method which constructs the hierarchy and performs dimensionality reduction on different layers of the hierarchical structure simultaneously.

3.2. Supervised Deep Kernel Dimensionality Reduction

In this section, we introduce our approach which is based on a kernel dimensionality reduction technique, and which constructs the layers of the deep framework simultaneously. We start with the method of Fukumizu, Bach, and Jordan [29, 30] and consider it in details since our approach is strongly based on it.

The semiparametric method known as Kernel Dimensionality Reduction (KDR) [29, 30], is based on the estimation and optimization of a particular class of operators on reproducing kernel Hilbert spaces (RKHS). The idea

is to relate dimensionality reduction to the problem of conditional independence, and to construct an objective function for optimization.

The KDR method assumes that it is possible to find a projection of initial covariate variables into a lower dimension space. The approach is based on an assumption that there is a r-dimensional subspace ($r \ll d$) which is referred to as the effective subspace. The dimensionality reduction can be viewed as a procedure testing conditional independence of variables such that

$$p(y|x) = \hat{p}(y|\theta^T x), \tag{4}$$

where θ is an orthonormal projection. The covariance operator on RKHS is responsible for capturing conditional independence between variables. The new representation in a more compact feature space is linear combinations of the components of observations.

The KDR method aims to minimize the following objective function

$$\det \hat{\Sigma}_{YY|U} = \frac{\det \hat{\Sigma}_{[YU][YU]}}{\det \hat{\Sigma}_{YY} \det \hat{\Sigma}_{UU}}, \tag{5}$$

where

$$U = \theta^T X, \tag{6}$$

and

$$\hat{\Sigma}_{[YU][YU]} = \begin{pmatrix} \hat{\Sigma}_{YY} & \hat{\Sigma}_{YU} \\ \hat{\Sigma}_{UY} & \hat{\Sigma}_{UU} \end{pmatrix} = \tag{7}$$

$$\begin{pmatrix} (\hat{K}_Y + \epsilon I_n)^2 & \hat{K}_Y \hat{K}_U \\ \hat{K}_U \hat{K}_Y & (\hat{K}_U + \epsilon I_n)^2 \end{pmatrix}, \tag{8}$$

where $\epsilon > 0$ is a regularization parameter. $\hat{K}_Y$ and $\hat{K}_U$ are the centralized Gram matrices defined as follows:

$$\hat{K}_Y = \Big(I_n - \frac{1}{n} 1_n 1_n^T\Big) G_Y \Big(I_n - \frac{1}{n} 1_n 1_n^T\Big), \tag{9}$$

$$(G_Y)_{ij} = k(Y_i, Y_j), \tag{10}$$

$$\hat{K}_U = \Big(I_n - \frac{1}{n} 1_n 1_n^T\Big) G_U \Big(I_n - \frac{1}{n} 1_n 1_n^T\Big), \tag{11}$$

$$(G_U)_{ij} = k(U_i, U_j). \tag{12}$$

The Gaussian kernel

$$k(a,b) = \exp\Big(\frac{-\|a-b\|^2}{\sigma^2}\Big) \tag{13}$$

is used throughout the paper and in our experiments.

To optimize the criterion, a gradient descent with line search can be used. The matrix of parameters is updated on iteration t according to

$$\theta^{t+1} = \theta^t - \gamma \frac{\partial \log\det \hat{\Sigma}_{YY|U}}{\partial \theta} = \tag{14}$$

$$\theta^t - \gamma 2\epsilon Tr[\hat{\Sigma}^{-1}_{YY|U}\hat{K}_Y(\hat{K}_U + \epsilon I_n)^{-1}\frac{\partial \hat{K}_U}{\partial \theta}(\hat{K}_U + \epsilon I_n)^{-2}\hat{K}_U\hat{K}_Y], \tag{15}$$

where

$$\hat{\Sigma}_{YY|U} = (\hat{K}_Y + \epsilon I_n)^2 - \hat{K}_Y\hat{K}_U(\hat{K}_U + \epsilon I_n)^{-2}\hat{K}_U\hat{K}_Y. \tag{16}$$

Therefore, the KDR approach produces a new reduced representation of the data X which is $\theta^T X$.

It was reported that the KDR is an efficient state-of-the art method of dimensionality reduction on real data [29, 30]. In general, if we want to combine the advantages of the KDR and a hierarchical "smoothing" structure, we could construct a cascade of KDRs, where an output of one run of the KDR would be an input for another run. However, in this situation we would obtain a solution which is approximated, and not exact.

The proposed deep dimensionality reduction technique is as follows. Each layer of the hierarchical structure is a new data representation $X'' = \theta_i^T X'$ of a layer underneath X', and where X', in its turn, is a reduced representation of some previous layer. The iterative process such as a convex optimization algorithm which updates parameters of a model, makes an update for parameters of all levels of the hierarchy on each iteration, i.e. simultaneously.

We introduce a deep semiparametric model with D layers

$$p(y|x) = \hat{p}\Big(y|\theta_D^T(\theta_{D-1}^T \ldots (\theta_1^T(\underbrace{\theta_0^T x}_{x'}))\Big), \tag{17}$$

$$\underbrace{\qquad\qquad}_{x''}$$

$$\underbrace{\qquad\qquad\qquad\qquad}_{\ldots}$$

where x', x'', ..., are the new representations of the deep structure that are learned simultaneously in one optimization procedure. We clearly see that

θ_{j+1} depends on θ_j for all $j \in \{1, \dots D\}$, and optimization can not be done separately for each layer. By the implicit function theorem, applying the chain rule, for each θ_j, except for θ_0, we have

$$\frac{\partial \ell(\theta)}{\partial \theta_j} = \frac{\partial \ell(\theta)}{\partial \theta_{j'}} \Big(\frac{\partial \ell^2(\theta)}{\partial \theta_{j'}^2} \Big)^{-1} \frac{\partial \ell^2(\theta)}{\partial \theta_j \partial \theta_{j'}}, \tag{18}$$

where $\ell = \det \hat{\Sigma}_{YY|U}$. In other words, to optimize parameters associated with a layer, to compute the first derivative with respect to parameters of this layer, we also need the second derivative of a layer underneath. Using eq. (18) we update simultaneously all θ in one iteration of a gradient descent, and new compact representations associated with different levels of generalization are estimated simultaneously according to

$$\theta_j^{t+1} = \theta_j^t - \gamma \frac{\partial \ell(\theta)}{\partial \theta_j}. \tag{19}$$

3.3. KDR versus DKDR: Discussion

A natural question which arises is why the Deep KDR is better than the KDR. Although it is currently impossible to provide a theoretical foundation for it, there is an intuition why the deep method is expected to and actually performs better in practice. Note that real data are always noisy, and a "good" clustering or dimensionality reduction can significantly reduce the noise. The major idea is that if features are tied into clusters of "high quality", then it is easier to detect a signal from data, and the generalizing classification performance is higher. The hierarchical dimensionality reduction plays here the role of a filter, and a filter with multiple layers seems to perform better than a one-layer filter.

Also note that the DKDR criterion is convex, and we can apply any gradient-based method to optimize the model parameters.

3.4. Unsupervised Case

The framework discussed above, can also be learned in an unsupervised manner, without any information about classes. To do it, we can apply standard dimensionality reduction methods or clustering methods. If we choose a clustering method, however, we have to decide what new variables will represent and how to define new variables based on clustering results (centers

of clusters, medoids, or anything else). In Section 5 where we share the results of our experiments, we compare performance of a number of standard clustering and dimensionality reduction methods.

In this section, we consider a heuristic method, a covariance-based clustering, which is both intuitively clear and robust. The idea to exploit the covariance operators to capture dependence between variables of interest and conditional independence between them, is the same as in the previous section and the one exploited by [30, 29]; [31] proposed to use the Hilbert-Schmidt norm of the normalized conditional cross-covariance operator to reveal the underlying structure between variables.

We propose to use the covariance operator as a distance measure for clustering, since, as mentioned above, the RKHS can provide information on independence of variables. Let the RKHS be (H_X, K_X) and (H_Y, K_Y), and random variables defined on them are $\Phi_X(X) = K_X(\cdot, X)$ and $\Phi_Y(Y) = K_Y(\cdot, Y)$, where K is positive definite.

The covariance operator on RKHS is defined as:

$$\Sigma_{YX} = E[\Phi_Y(Y)\langle\Phi_X(X), \cdot\rangle] - E[\Phi_Y(Y)]E[\langle\Phi_X(X), \cdot\rangle], \tag{20}$$

where Σ_{YX} is an operator from H_X to H_Y such that

$$\langle g, \Sigma_{YX} f\rangle = E[g(Y)f(X)] - E[g(Y)]E[f(X)] = \tag{21}$$

$$\operatorname{cov}(f(X), g(Y)), \text{ for all } f \in H_X, g \in H_Y. \tag{22}$$

In the Euclidean case, we have the covariance matrix

$$V_{YX} = E[YX^T] - E[Y]E[X]^T, \text{ and} \tag{23}$$

$$(b, V_{YX}a) = \operatorname{cov}\big((b, Y), (a, X)\big). \tag{24}$$

In the experimental section, the approach is referred to as covariance-based clustering. The covariance operator can be also applied in an unsupervised dimensionality reduction based on an objective function optimization, as it is done e.g., in [32].

4. Experiments on Standard Data Sets

In this section we apply the proposed deep approach to two standard biological data sets, the Golub et al. (1999) data and Alon et al. (1999) set.

We compare the performance of the Deep Kernel Dimensionality Reduction method to some standard unsupervised dimensionality reduction approaches such as

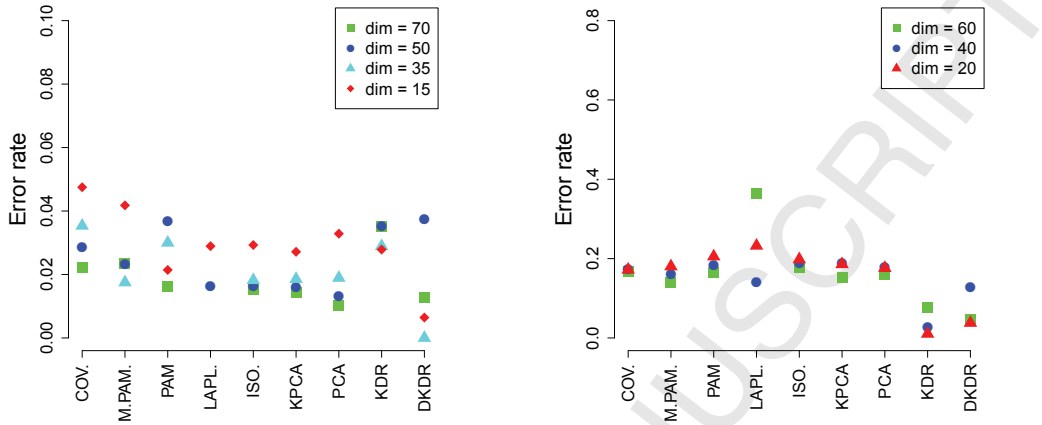

Figure 1: Experiments on Golub Data Set (on the left) and Alon Data (on the right). Error rate as a function of dimensionality reduction method and dimension in reduced models.

- Principal Component Analysis (PCA),
- Kernel Principal Component Analysis (KPCA),
- Isomap (ISO),
- Laplacian Eigenmaps (LAPL),
- robust clustering methods, such as the Partitioning Around Medoids (PAM) which is a robust version of the k-means; the medoids are representatives of clusters,
- PAM clustering, where the representatives of clusters are median values of instances in clusters (M.PAM),
- the heuristic unsupervised covariance-based clustering (COV) described in Section 3.4.

We also compare our results to the following supervised dimensionality reduction approaches:

- the full model, i.e. the model with the original high-dimensional feature space,

- a supervised dimensionality reduction learning procedure KDR.

We use an SVM with an Radial Basis Function kernel to learn models. We use the cross validation method to adjust parameters of all approaches being tested.

In Golub [33] data we dispose of 72 patients and about 7000 gene expressions (Affymetrix probes). Among these patients, 47 subjects have acute lymphoblastic leukemia, and 25 are diagnosed with acute myeloid leukemia, therefore, we have a classification problem with 2 classes. Figure 1 on the left illustrates the results in terms of 5-folds cross validation error rate. Since the number of observations is 72, we consider the performance of models with reduced dimensionality. We consider reduced models with 70, 50, 35, and 15 parameters. The choice of the reduced dimensions is due to the number of observations: for several dimensionality reduction methods (PCA, KPCA, Isomap) the reduced dimension of parameters has not be bigger than one of observations. So, in the DKDR case we have a hierarchy of 4 layers (with dimensions 70, 50, 35, 15). We see quite clearly that the proposed DKDR approach outperforms all other methods, and the best accuracy is reached for models with the least number of parameters, i.e., 15 and 35 features.

The Alon data set [34] contains 62 patients and 2000 gene expressions (Affymerix origonucleaotide array) of colon tissues. The patients are coming from two classes: 40 patients are diagnosed with a tumor, and 22 patients have normal colon tissues. The results on the dimensionality reduction experiments are shown on Figure 1 on the right (5-folds cross validation). Taking into consideration that the number of patients is 62, we reduce the dimensionality to 60, 40, and 20. The results are similar to ones we obtained on the Golub data, except for the fact that KDR here slightly outperforms the DKDR (3 layers). However, the best models are the most compact among tested.

Figure 2 shows the performance of the state-of the art methods on Alon and Golub data. HKL stands for Hierarchical Kernel Learning [28], SKMsmo stands for Support Kernel Machine solved by Sequential Minimal optimization [27], and SSPCA for Structured Sparsity PCA [20]. We observed that the HKL and SKMsmo methods have a tendency to overfit. We have run experiments with Hermite kernel, Polynomial kernel, hermite expansion of Gaussian kernel, and Anova kernel. The best results were achieved with the Hermite kernel. The SSPCA method leads to accurate results on the data, and the error rates are comparable to ones we obtained with the DKDR. The

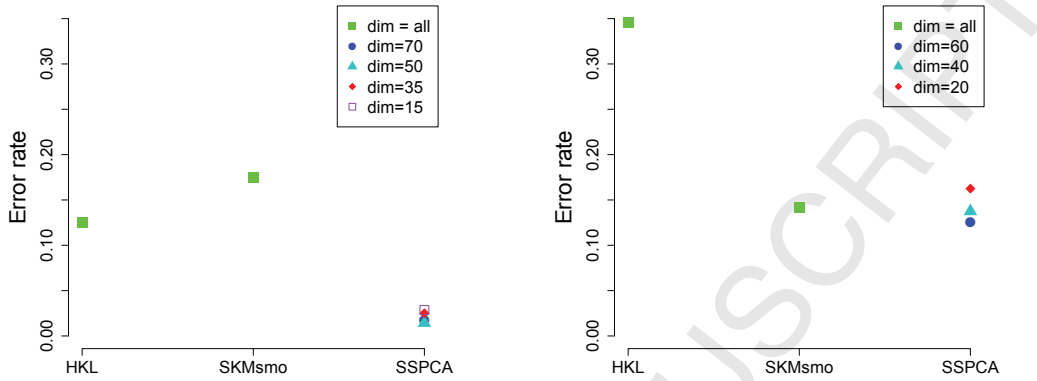

Figure 2: Experiments on Golub Data Set (on the left) and Alon Data (on the right). Error rate as a function of state-of-the art dimensionality reduction approaches.

dimensions chosen for the SSPCA are the same as above, on Figure 1.

5. Experiments on Real MicrObese Data

In this section, we show that the framework introduced above in Section 3, can be efficiently applied to a real high-dimensional heterogeneous data integration problem.

We describe our results on the MicrObese data [1], and we compare the performance of the deep kernel dimensionality reduction (DKDR) to state-of-the-art dimensionality reduction methods.

The MicroObese cohort consists of data coming from different sources, including clinical data of patients, abundance of gut flora genes, and gene expressions of adipose tissue. In our experiments, we consider models which integrate these heterogeneous sources pairwise and altogether. Our primary goal is to illustrate that the DKDR is an efficient dimensionality reduction method. Another question is which data source or a combination of data sources is more informative for the patients classification.

5.1. Brief MicrObese Data Description

The MicrObese corpus contains meta-data, genes of adipose tissue, and gut flora metagenomic data. For each patient, we have the information to

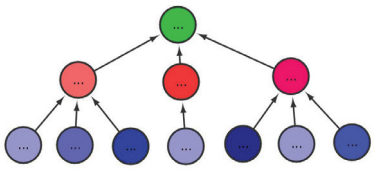

Figure 3: Scheme of deep data representation learning, where latent variables are "mixed". The blueish nodes are variables coming from different data sources. The upper layers are therefore also mixed.

which class he or she belongs. There are two classes, high gene count (HGC) and low gene count (LGC) classes. Therefore, our problem is a binary prediction task from heterogeneous data.

In general, 49 patients have been hired and examined at the Pitié-Salpêtrière hospital, Paris, France [1], but as to the genes of the adipose tissue, we dispose data for less patients, and not for all patients their class, LGC or HGC is provided. Therefore, in our experiments we have access to 35 observations (patients). To get rid of important noise, we run a significance test (Kruskal-Wallis), and we keep those variables for which the raw (not adjusted for the multiple hypothesis testing) p-values < 0.05.

Initially, we have 135 meta-parameters which can be divided into clinical parameters and alimentary patterns reflecting nourishing habits of the patients. The data set contains more than 42,000 genes of the adipose tissue, and the gut flora data contains counts for more than 3 million genes. The metagenomic matrix is quite sparse, and not all of the genes are significant. We have pre-selected about 24,000 genes of gut flora and about 350 genes of the adipose tissue for our further experiments. As to the clinical parameters, only 7 of them are significant enough (with respect to the LGC and HGC classes) to be considered in our experiments. Although we reduced some important noise in data with significance tests, and, therefore, reduced the dimensionality of the task, the problem is still a perfect illustration of $n \ll p$ problem, i.e., where the number of observations is much smaller than the number of parameters.

5.2. Deep Dimensionality Reduction on MicrObese Data

We compare the results of DKDR on the MicrObese data set to the state-of-the-art dimensionality reduction methods mentioned above, in Section 4. Note that "ALL" method stands for the result without any feature selection

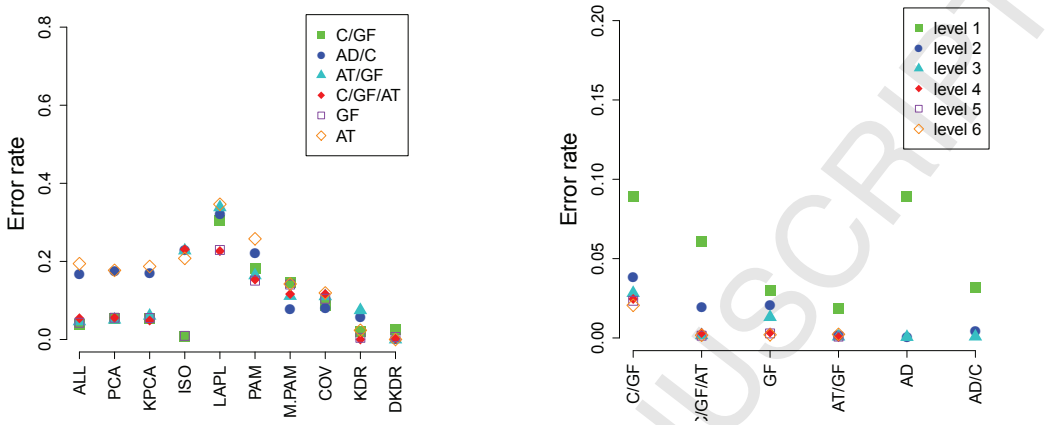

Figure 4: MicrObese Cohort. On the left: error rate as a function of dimensionality reduction method and data integrated into the model. On the right: error rate as a function of data integrated and level in the hierarchy.

or dimensionality reduction. To train models with and without dimensionality reduction, we use an SVM with an RBF kernel [35], since a non-linear separator is more efficient on our data. We show the results in terms of the 5-folds cross validation error rate. As mentioned before, we have three sources of data, and we test various combinations of them to explore the data. We test the following combinations of data sources

- Gut Flora metagenomics (GF abbreviation on Figure 4)
- gene expressions of Adipose Tissue (AT)
- Clinical parameters, Gut Flora abundance, and gene expressions of Adipose Tissue (C/GF/AT)
- Clinical parameters and Gut Flora metagenomics (C/GF)
- Clinical parameters and gene expressions of Adipose Tissue (C/AT)
- Gut Flora metagenomics and gene expressions of Adipose Tissue (GF/AT)

We decided not to run tests with clinical parameters only, since there are too few of them. Optimal parameters for all tested methods are chosen by

cross validation. Figure 3 provides some intuition on data integration in the proposed hierarchical model.

We construct the deep framework DKDR as follows. Although the choice of the number of layers in the hierarchy is a delicate matter, here, without loss of generality, we fix the dimension of each level to be 2 times smaller than the dimensionality of its lower level. The number of genes of gut flora is about 24,000, and all models including this data source contain more than 24,000 parameters. So, for all the models with gut flora, i.e., Gut Flora (GF), Clinical parameters, metagenomics of Gut Flora, and gene expressions of Adipose Tissue (C/GF/AT), Clinical parameters and Gut Flora metagenomics (C/GF), and Gut Flora metagenomics, gene expressions of Adipose Tissue (GF/AT), we construct a hierarchy with 6 levels. The models with gene expressions of Adipose Tissue (AT) and Clinical parameters and gene expressions of Adipose Tissue (C/AT) have 3 levels only, since we consider about 350 genes of adipose tissue.

Figure 4 on the left shows the error rate as a function of a dimensionality reduction method and the data being used for the classification task. We have observed that data integration has a positive effect: integrating all data sources leads to a lower error on a test data set (5-folds cross validation error rate). It is also easy to see that the proposed DKDR approach reaches a higher accuracy than other state-of-the-art methods.

We would also like to understand the impact of the number of layers in a hierarchy. Figure 4 on the right illustrates our observations on the MicrObese data. The highest layer (the most compact models) is 6 for the models including the Gut Flora genes (GF, C/GF, GF/AT, C/GF/AT), and level 3 for all other models (C/AT, AT). Dimensionality of level 1 of models with gut flora is about 12,000 (initial dimension 24,000 is divided by 2 for level 1), and the dimensionality of level 6 is about 30, since the number of patients is limited to 35. The number of features in models without gut flora genes is 150 for level 1, and about 30 for level 3 which is the highest for these models. We notice that the worst performance is obtained by models with most features. It is also possible that performance of several levels is the same, and that further dimensionality reduction does not ameliorate the accuracy anymore. Note, however, that models with 6 levels (for ones with gut flora) and 3 levels (for all the rest) are the most compact ones, and also the most efficient in terms of prediction.

Figure 5 demonstrates the performance we get with the state-of-the art Hierarchical Kernel Learning, Multiple Kernel Learning, and Sparse Struc-

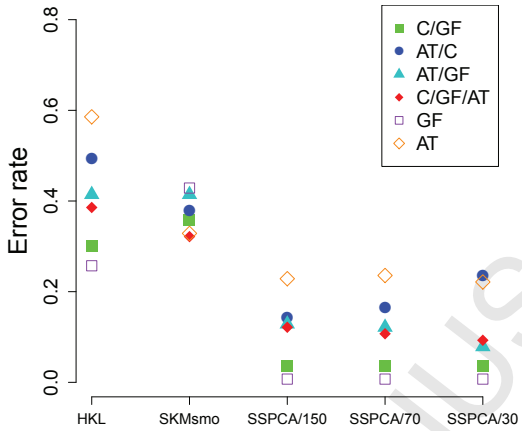

Figure 5: Results on MicroObese Data. Error rate as a function of data sources and of state-of-the-art methods.

tured PCA methods. The accuracies are consistent with our previous results, showing that SSPCA is a very competitive method, and that genes of gut flora and clinical parameters are the best predictors of the gene richness.

Figure 6 illustrates a hierarchy of clinical parameters and alimentary patterns of MicrObese data set, and Table 1 provides a brief description of the parameters. In the deep structure on Figure 6, each level is a generalization of its lower level. E.g., if we look at the leftmost branch of the tree, we will see that for a reasonable patients classification it is sufficient to measure walking index and particular bacteria (the yellow node), and spending efforts on measuring total cholesterol, ratio of total cholesterol to HDL cholesterol, non-HDL cholesterol, and triglycerides does not bring any additional information. Note that the predictive power of the upper level (the yellow one, with a quite small number of parameters) is not worse than of the lowest level of the tree.

A particular interest is to consider mixed signatures, i.e. feature selection where parameters come from more than one data source. Figure 7 shows a signature (the highest layer of a hierarchy) based on both clinical parameters and genes of adipose tissue. It is quite interesting that some bacteria are strongly associated with specific genes and probably share the same biological

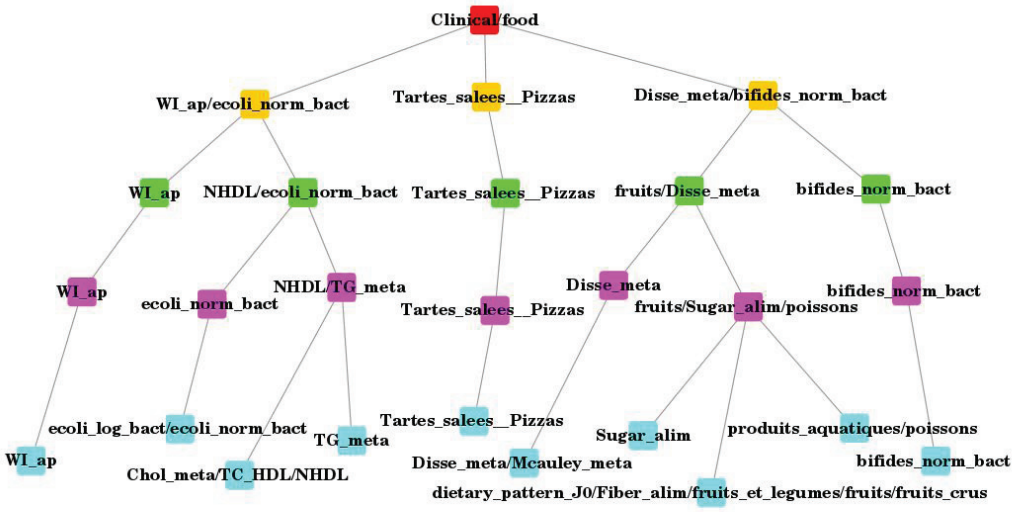

Figure 6: A hierarchy of clinical parameters of MicrObese data constructed by a discrete approach.

WI_ap	Walking index based on physical activity
ecoli_log (norm)_bact	Escherichia coli in log scale and normalized
Chol_meta	Total cholesterol
TC_HDL, NHDL	Ratio of cholesterol, non-HDL cholesterol
TG_meta	Triglycerides
Tartes_salees_Pizzas	Savory pies and pizza
Disse_meta, Mccauley_meta	Insuline sensitivity
Sugar_alim	Sugar intake
produits_aquatiques_poissons	Fish and fish products
dietary_pattern, Fiber_alim	Diatary quality clusters
fruits_et_legumes, fruits, fruits_crus	Fruit and vegetables intake

Table 1: Description of clinical parameters of MicrObese data.

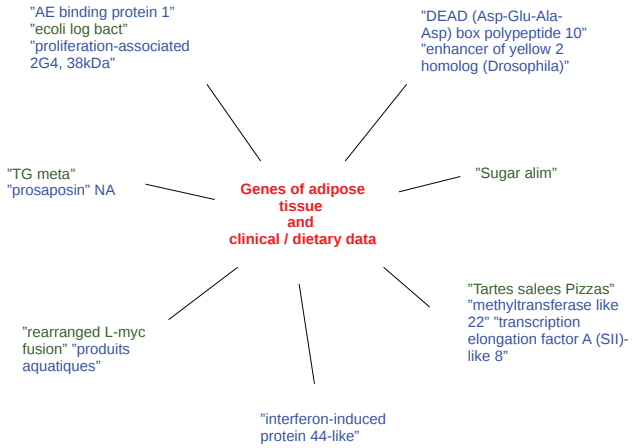

Figure 7: A signature based on clinical parameters and genes of adipose tissue.

function.

6. Conclusion

Data integration is a delicate problem, especially in applications where data are high-dimensional (metagenomics) and the number of observations is small. We have proposed to reduce dimensionality by a deep kernel-based approach which learns new representations of data simultaneously in a hierarchical way, and which do not waste any effort on modeling data distributions, as the state-of-the-art methods do. We have considered supervised and unsupervised dimensionality reduction, as well as we considered a real data integration challenge. We show that the novel deep kernel dimensionality reduction is efficient on standard data sets, and on a real medical complex data set, and significantly outperforms modern state-of-the-art approaches. Moreover, the multi-level hierarchy can provide new scientific hypotheses for biologists doing pre-clinical research and help to develop methods of personalized medicine.

Acknowledgments

The clinical work was supported by Agence Nationale de la Recherche (ANR MICRO-Obes), KOT-Ceprodi and the association Fondation Coeur et Arteres. All ethical agreement are obtained. This work is also part of the European Union's Seventh Framework Program under grant agreement HEALTH-F4-2012-305312 (MetaCardis project).

[1] A. Cotillard, al., Dietary intervention impact on gut microbial gene richness, Nature 500 (2013) 585–588.
URL doi:10.1038/nature12480

[2] E. Le Chatelier, al., Richness of human gut microbiome correlates with metabolic markers, Nature.
URL http://dx.doi.org/10.1038/nature12506

[3] M. Ré, G. Valentini, Simple ensemble methods are competitive with state-of-the-art data integration methods for gene function prediction, in: JMLR: Machine Learning in Systems Biology, 2010.

[4] A. Anandkumar, D. Hsu, A. Javanmard, S. M. Kakade, Learning linear Bayesian networks with latent variables, in: International Conference on Machine Learning, Vol. 28, 2013, pp. 249–257.

[5] G. Obozinski, G. Lanckriet, C. Grant, M. I. Jordan, W. S. Noble, Consistent probabilistic outputs for protein function prediction, Genome Biology S6 (9).

[6] Y. Guan, C. L. Myers, D. C. Hess, Z. Barutcuoglu, A. A. Caudy, O. G. Troyanskaya, Predicting gene function in a hierarchical context with an ensemble of classifiers, Genome Biology S3 (9).

[7] Y.-Y. Yu, T. Fletcher, S. P. Awate, Hierarchical graphical models for multigroup shape analysis using expectation-maximization with sampling in kendall's shape space, in: arXiv:1212.5720, 2013.

[8] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2007.

[9] C. B. Do, S. Batzoglou, What is the expectation maximization algorithm?, Nature Biotechnology 26 (8).

[10] R. Mourard, C. Sinoquet, P. Leray, A hierarchical Bayesian network approach for linkage disequilibrium modeling and data-dimensionality reduction prior to genome-wide association study, BMC Bioinformatics 16 (11).

[11] P. Scheet, M. Stephens, A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypesand haplotypic phase, Am J Human Genet 4 (78) (2006) 629–644.

[12] R. Bekkerman, R. El-Yaniv, A. McCallum, Multi-way distributional clustering via pairwise interactions, in: ICML, 2005, pp. 41–48.

[13] S. Jegelka, A. Gretton, B. Schölkopf, B. Sriperumbudur, U. von Luxburg, KI 2009: Advances in Artificial Intelligence: 32nd Annual German Conference on AI. Proceedings, Springer Berlin Heidelberg, 2009, Ch. Generalized Clustering via Kernel Embeddings.

[14] A. Gretton, K. Borgwardt, M. Rasch, B. Schoelkopf, A. Smola, A kernel method for the two-sample-problem, in: NIPS, 2006.

[15] C. Bartenhagen, H.-U. Klein, C. Ruckert, X. Jiang, M. Dugas, Comparative study of unsupervised dimension reduction techniques for the visualization of microarray gene expression data, BMC Bioinformatics (11).

[16] H. Hotelling, Analysis of a complex of statistical variables into principal components, Journal of Educational Psychology 24 (1933) 417–520.

[17] I. Jolliffe, Principal Component Analysis, Springer, 2002.

[18] B. Schölkopf, A. Smola, K. Müller, Kernel principal component analysis, in: Advances in kernel methods: support vector learning, MIT Press, 1999, pp. 327–352.

[19] I. A. Gkioulekas, T. Zickler, Dimensionality reduction using the sparse linear model, in: NIPS, 2011.

[20] R. Jenatton, G. Obozinski, F. Bach, Structured sparse principal component analysis, in: Proceedings of the International Conference on Artificial Intelligence and Statistics, 2010.

[21] V. Silva, J. Tenenbaum, Global versus local methods in non-linear dimensionality reduction, in: NIPS, 2003.

[22] J. Tenenbaum, V. de Silva, J. Langford, A global geometric framework for nonlinear dimensionality reduction, Science (290) (2000) 2319–2323.

[23] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral clustering for embedding and clustering, in: T. G. Dietterich, S. Becker, Z. Ghahramani (Eds.), NIPS, 2001, pp. 585–591.

[24] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (6) (2003) 1373–1396.

[25] G. Hinton, R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (2006) 504–507.

[26] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, A. Y. Ng, Multimodal deep learning, in: ICML, 2011.

[27] F. Bach, G. R. G. Lanckriet, M. I. Jordan, Multiple kernel learning, conic duality, and the smo algorithm, in: ICML, 2004.

[28] F. Bach, Large feature spaces with hierarchical multiple kernel learning, in: NIPS, 2008.

[29] K. Fukumizu, F. Bach, M. I. Jordan, Kernel dimensionality reduction for supervised learning, in: NIPS, 2003.

[30] K. Fukumizu, F. Bach, M. Jordan, Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces, Journal of Machine Learning Research 5 (2004) 73–99.

[31] K. Fukumizu, A. Gretton, X. Sun, B. Schölkopf, Kernel measures of conditional dependence, in: NIPS, 2007.

[32] M. Wang, F. Sha, M. I. Jordan, Unsupervised kernel dimension reduction, in: NIPS, 2010.

[33] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, E. S. Lander, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring., Science 286 (5439) (1999) 531–537.

[34] U. Alon, N. Barkai, D. Notterman, K. Gish, S. Ybarra, D. Mack, A. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proceedings of the National Academy of Sciences 96 (12) (1999) 6745–6750.

[35] A. Karatzoglou, A. Smola, K. Hornik, A. Zeileis, kernlab – an S4 package for kernel methods in R, Journal of Statistical Software 11 (9) (2004) 1–20.