Including semi-supervision in a kernel matrix, with a view to interactive visual clustering

Pierrick Bruneau, Benoît Otjacques

Including semi-supervision in a kernel matrix, with a view to interactive visual clustering

Pierrick Bruneau *and Benoît Otjacques †

CRP - Gabriel Lippmann, Department of Informatics
41, rue du Brill, L-4422 Belvaux (Luxemburg)

Abstract

In this paper, a new kernel transformation procedure is described. It aims at incorporating a degree of supervision directly in the original pairwise similarities of a data set. The modified similarities can then be projected using a 2D kernel PCA [13], so as to reflect the compromise between genuine data and user knowledge, while being affordable for visualization and interaction. Such semi-supervised projections are evaluated with synthetic and real data, in the context of a simulated visual clustering task. Randomly selected subsets of elements are chosen to hold a label, thus reproducing actual user interactions. The results show the effectiveness of the method, with as few as one labelled element per class inducing tangible effects.

1 Introduction

Clustering is a valuable task in the context of a visual analysis, e.g. allowing to simplify the visualization of large data sets [8]. Visual objects such as clusters are strong visual cues [16], and consequently natural candidates to become entry points for a visual analysis. Yet, clusters need to be projected on a low dimensional space (preferably 2D) to become affordable as visual objects. Setting up a visual clustering system is thus not trivial, as real data sets are often high dimensional.

In this paper, we propose a new kernel construction procedure, that combines genuine pairwise similarities with prior labels. Performing a 2D kernel PCA projection with this custom kernel then allows to combine smoothly intrinsic topology with user-specified contraints. The elements can then be clustered

*bruneau@lippmann.lu
†otjacque@lippmann.lu

meaningfully in this transformed 2D space.

In real world application, unlabelled data is often plentiful, but learning examples much less, as they often rely on some handcrafted ground truth (e.g. data elements labelled by a domain expert). In this context, the semi-supervised learning task can be understood in two loosely separated ways:

- as a supervised learning task (i.e. classification) with a very small training set (see figure 1). This setting usually prevents the usage of most supervised learning algorithms. Yet, some authors proposed to exploit the density of unlabelled (i.e. easily disposable) data to overcome this limitation [5].

- as a clustering task with few labelled samples, with a view to incorporating some expert knowledge (see figure 2). This knowledge may not fully conform to the criterion used by a given unsupervised procedure, and the purpose of the semi-supervised method is to handle this conflict smoothly. This can amount to using the labelled examples for model initialization, and optionnally forcing their cluster memberships afterwards [2, 12]. In the context of probabilistic models, some authors transformed a set of prelabelled elements to contraints (i.e. must-link and must-not-link) within a classic maximum likelihood scheme, and derived an adapted optimization algorithm [9].

The existing semi-supervised clustering approaches suffer from several limitations :

- all the mentionned works rely on linear transforms, and mostly on Gaussian shaped clusters, which may be too restrictive in a variety of real-life situations (e.g. data lying on nonlinear manifolds, non-Gaussian data).

- some works tried to allow a wider range of class shapes, with the possible association of multiple Gaussian components to a single class [11]. But tuning the resulting algorithm appears to be rather tedious and data-dependent.

In this work, we do not intend to explicitly enforce contraints as in the works quoted above. We rather seek to embed a compromise between genuine similarities and user-specified labels in a nonlinear transform. In other words, we derive a 2D projection that follows the original topology as far as some prior information permits it. Any clustering algorithm, such as k-means or Gaussian-EM [3] may then operate on this low-dimensional continuous numeric data.

The visualization of high-dimensional data using 2D projections, and the distortion artifacts that may occur have been extensively studied [1], and are still an active research topic. Our contribution may be viewed as a complementary building block to this existing body of work : according to the terminology

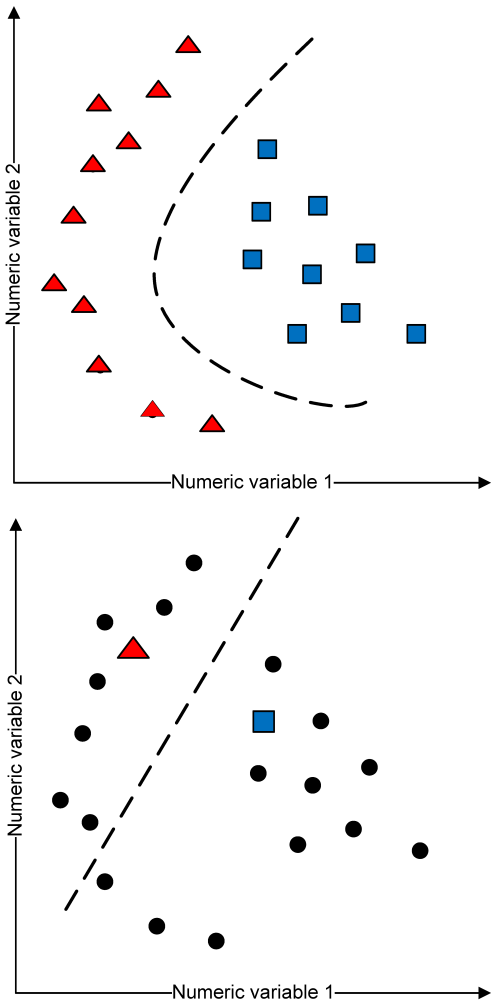

Figure 1: *Top*: decision boundary induced by a supervised learning algorithm with a labelled training set.
Bottom: with only two labelled elements, the inferred boundary is very poor. The density of unlabelled elements may then be useful to improve it.

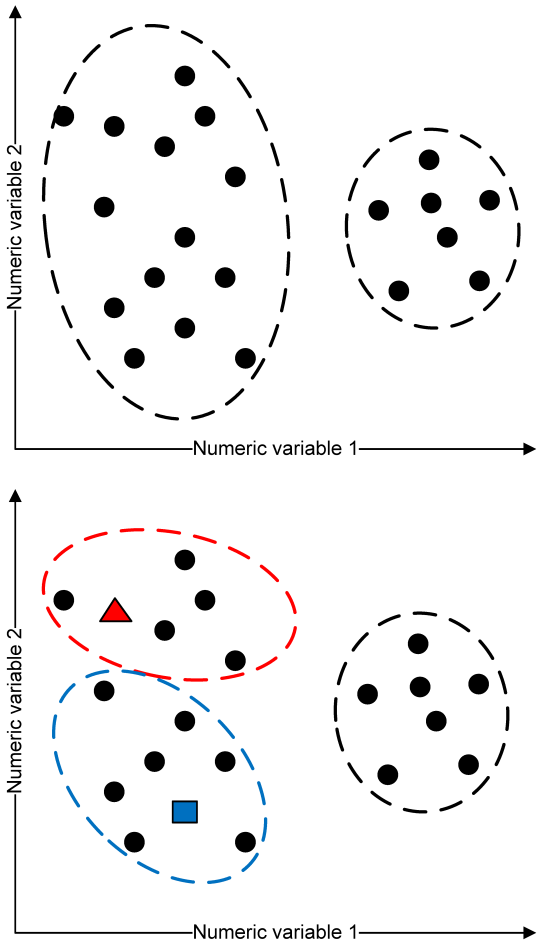

Figure 2: *Top*: potential result from a clustering algorithm.
Bottom: the adjunction of two labelled examples may suggest a different cluster structure.

stated in [1], it can be understood as a nonlinear continuous projection technique.

In section 2, for self-contency we briefly recall the theoretical foundations of the kernel PCA technique, and emphasize its possible support to data visualization with its well-behaved output nonlinear 2D projections. Then in section 3, we propose a new semi-supervised kernel transformation procedure. It outputs a compromise between genuine similarity information, and prior labelling.

This kernel may be included in the following visual clustering task:

1. perform a 2D kernel PCA projection using the semi-supervised kernel,
2. cluster the projected data.

The semi-supervision would be fed by user interactions (e.g. click-and-label actions on the 2D visualization).

In this paper, the purely interactive aspect is set aside, to focus experimentally on an objective evaluation of the behavior of the proposed kernel, when confronted to a randomly selected prior subset of labelled elements. We intend to highlight instrinsic properties of our proposition, and contrast it to an alternative approach. A baseline unsupervised kernel serves as a control setting for this comparison. After a critical discussion of our results, we conclude with the numerous perspectives that this work opens in the visual data mining domain.

2 Kernel PCA for 2D projection

Let us consider a set of elements $\mathbf{X} = \{\mathbf{x}_i\}_{i \in 1 \dots N}$, with values in some domain $\mathcal{X}$ (referred to as *original space* hereafter), and a nonlinear transformation ϕ that projects any element $\mathbf{x}_i$ onto a point $\phi(\mathbf{x}_i) \in \mathbb{R}^M$ (called *feature space* in the remainder).

Assuming $\sum_{i=1}^{N} \phi(\mathbf{x}_i) = \mathbf{0}$, the sample covariance matrix of the image of $\mathbf{X}$ in the feature space is given as:

$$\mathbf{C} = \frac{1}{N} \sum_{i=1}^{N} \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^T,$$

with the associated eigenvector equation:

$$\mathbf{C}\mathbf{v}_m = \lambda_m \mathbf{v}_m, \ m = 1 \dots M.$$

Considering the kernel function $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}')$, and following works by [13] and [3], this eigenvector problem can be transformed to:

$$\mathbf{K}\mathbf{a}_m = \lambda_m N \mathbf{a}_m, \ m = 1 \dots M, \tag{1}$$

with $\mathbf{K}$ the $N \times N$ matrix such that $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, and $\mathbf{a}_m$ a vector in $\mathbb{R}^N$. After solving (1) for its eigenvectors and eigenvalues, a set of M projection functions can be defined as follows:

$$y_m(\mathbf{x}) = \sum_{i=1}^{N} a_{mi} k(\mathbf{x}, \mathbf{x}_i). \tag{2}$$

Assuming eigenvalues in decreasing order, the 2D projection that captures the maximal variance in the feature space is then built with y_1 and y_2. The assumption $\sum_{i=1}^{N} \phi(\mathbf{x}_i) = \mathbf{0}$ can be released with the following modified kernel expression [3]:

$$\tilde{\mathbf{K}} = \mathbf{K} - \mathbf{1}_N\mathbf{K} - \mathbf{K}\mathbf{1}_N + \mathbf{1}_N\mathbf{K}\mathbf{1}_N,$$

with $\mathbf{1}_N$ the $N \times N$ matrix in which every cell has the value $\frac{1}{N}$. The mapping ϕ does generally not have to be explicitly defined: indeed, any positive semi-definite matrix $\mathbf{K}$ was proven to be the dot product in some feature space, may it be infinite dimensional [3]. Thus, practitioners preferably design kernel functions directly, only caring about the positive semi-definiteness of the induced kernel matrices.

The Gaussian kernel function is adequate to this respect, and is defined as follows:

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\sigma^2}\right).$$

Let us remark that this choice would imply the usage of the Euclidian norm, thus implicitly setting $\mathcal{X}$ to $\mathbb{R}^d$.

This kernel function has been extensively used in the literature, but was experimentally found inadequate when considering high-dimensional data ($d > 100$). This fact has already been noticed by some authors [7]. Alternatively, they propose the p-Gaussian function:

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{d_{L2}(\mathbf{x}, \mathbf{x}')^p}{\sigma^p}\right), \tag{3}$$

with $d_{L2}(.,.)$ the Euclidian distance. This function was adjoined by empirical formulas for setting p and σ, designed to ensure that the kernel values match the cumulative distribution of the distances in the original space, irrespective of its dimensionality:

$$p = \frac{\ln(\frac{\ln 0.05}{\ln 0.95})}{\ln \frac{d_{L2}^{5\%}}{d_{L2}^{95\%}}}, \quad \sigma = \frac{d_{L2}^{95\%}}{(-\ln 0.05)^{\frac{1}{p}}} = \frac{d_{L2}^{5\%}}{(-\ln 0.95)^{\frac{1}{p}}}, \tag{4}$$

with $d_{L2}^{5\%}$ (resp. $d_{L2}^{95\%}$) the 5% (resp. 95%) percentile of the cumulative distribution of d_{L2}[1]. In the remainder of this paper, the kernel expression (3) will be used as a baseline.

In figure 3, we show how a data set originating from three loosely separated 2D Gaussian components is projected using equations (4), (3), and (2). In this example, the data seems to be "inflated": the pairwise distances distribution remains similar after transformation, but the intrinsic topology (i.e. cluster structure) is now emphasized.

3 Proposed semi-supervised kernel transformation

In this section, kernel values output by k are assumed to range in $[0, 1]$. This assumption is rather conventional [7], and is respected by the p-Gaussian kernel.

A clustering task partly amounts to assigning labels (unknown *a priori*) to a collection of elements. The goal is then to achieve the best labelling with respect to (abbreviated w.r.t. in the remainder) some ground truth grouping. Recalling the data set $\mathbf{X}$ defined in the previous section, let us define a labelling function, that matches each element to one of R possible classes:

$$\begin{aligned} l : \mathbf{X} &\to \{1, \dots, R\} \\ \mathbf{x} &\to l(\mathbf{x}). \end{aligned}$$

In this paper we assume a semi-supervised context, i.e. a potentially incomplete labelling: thus we will further refer to a finite set of labelled elements $\mathbf{X}_L \in \mathbf{X}$. This allows us to access l only through its restriction $l' = l|_{\mathbf{X}_L}$.

Note that l' can define every level of supervision, from completely unsupervised (i.e. $\mathbf{X}_L = \emptyset$), to fully supervised (i.e. $\mathbf{X}_L = \mathbf{X}$), and all intermediate mappings.

Our intuition is to transform the kernel function according to the respective nearest labelled neighbors of its arguments. The following function implements part of this intuition, and gets the nearest labelled neighbor of any element in $\mathbf{X}$:

$$\begin{aligned} s : \mathbf{X} &\to \mathbf{X}_L \\ \mathbf{x} &\to s(\mathbf{x}) = \begin{cases} \emptyset & \text{if } \mathbf{X}_L = \emptyset \\ \arg\max_{\mathbf{x}' \in \mathbf{X}_L} k(\mathbf{x}, \mathbf{x}') & \text{else.} \end{cases} \end{aligned}$$

l and s are then used to transform k as follows:

$$k'(\mathbf{x}, \mathbf{x}') = \begin{cases} k(\mathbf{x}, \mathbf{x}') & \text{if } \mathbf{X}_L = \emptyset \\ k(\mathbf{x}, \mathbf{x}')^{\frac{1}{\alpha}} & \text{if } \mathbf{X}_L \neq \emptyset \wedge l'\big(s(\mathbf{x})\big) \neq l'\big(s(\mathbf{x}')\big) \\ k(\mathbf{x}, \mathbf{x}')^{\alpha} & \text{if } \mathbf{X}_L \neq \emptyset \wedge l'\big(s(\mathbf{x})\big) = l'\big(s(\mathbf{x}')\big), \end{cases} \tag{5}$$

[1] In the referenced paper, $d_{L2}^{5\%}$ and $d_{L2}^{95\%}$ have been mistakenly swaped in the expressions for σ. A corrected version is reported here.

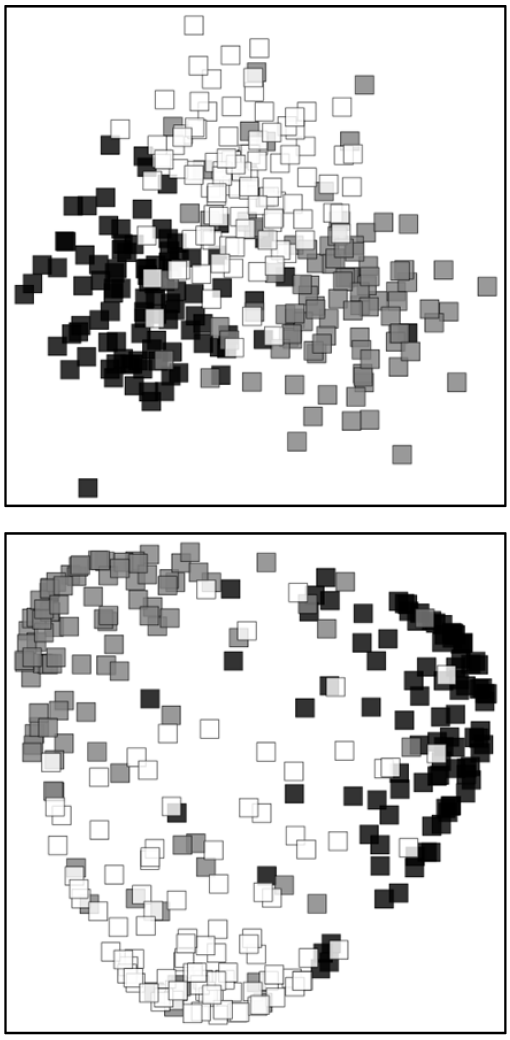

Figure 3: *Top*: Original 2D data set. The Gaussian component that originated each element is identified by a distinct shade of grey.
Bottom: 2D kernel PCA projection with the p-Gaussian function.

with $\alpha \in \mathbb{N}^*$. Intuitively, with k outputs ranging in $[0, 1]$ as required, and $\alpha > 1$, transforming k to k' amounts to increasing (resp. decreasing) the similarity of elements which have the same image value (resp. a different image value) by $l' \circ s$, while remaining in the appropriate range.

Though the illustrations and experiments conducted in this paper use the p-Gaussian kernel function (see equation (3)), the equation (5) may applied to any positive semi-definite kernel function with values ranging in $[0, 1]$ without loss of generality.

The strict inspection of kernel contruction rules (see e.g. [14] or [3]) would suggest that the function defined by eqn. (5) (and even the p-Gaussian function) is not a valid kernel (i.e. is not positive semi-definite). However, some invalid kernel functions have successfully been used in the literature [15]. Furthermore, in this work we only use the 2 first eigendimensions (i.e. 2D projection), which happen to be sufficiently well behaved with all but extremely ill-conditioned kernel matrices.

4 Experimental protocol

4.1 Task Description

The kernel transformation procedure proposed in the previous section is included in the following interactive visual clustering task:

1. an initial 2D projection (equation (2)) is computed with the p-Gaussian kernel matrix (equation (3)),
2. labels (i.e. class values) are associated to each element by a clustering algorithm,
3. a user updates the class semantics and the labels of some elements according to her preferences,
4. the user labelling is used to transform the initial kernel matrix (equation (5)),
5. this new kernel matrix is used to update the 2D kernel PCA projection,
6. go to step 2 unless the user is satisfied with the current projection and labels.

In this paper, we leave the purely interactive aspect aside, and choose to focus experimentally on a thorough evaluation of the behavior of the proposed kernel, when confronted to randomly selected subsets of labelled elements.

For a better estimation of the effects of the proposed kernel transformation, we contrast it with the two following alternatives:

- the unsupervised p-Gaussian kernel matrix (which will act as our control group),
- the translation of an existing semi-supervised clustering technique [2] to the kernel context. In brief, this approach is originally equivalent to constrain the membership of the user-designated elements, and bias the clustering algorithm using these static memberships. In the terms of the present paper, it is turned into using equation (5) without the neighborhood function, i.e. the kernel value $k(\mathbf{x}, \mathbf{x}')$ is transformed iif $\mathbf{x}$ and $\mathbf{x}'$ both belong to $\mathbf{X}_L$.

4.2 Quality Metrics

The performance of these methods are evaluated using the following metrics:

- the number of classes inferred by the clustering algorithm (identified as **nclass** in table 2),
- the **purity** of the clusters.

Also, as evoked in the introduction, the present work relates to the visualization and 2D projections literature. Thus, we also consider distortion metrics [1], that measure the compression (**compress** id in table 1) and stretching (**stretch** id) of the pairwise distances in the projection w.r.t. the respective distances in the original space.

Both these measures are normalized in $[0, 1]$, 1 indicating the highest distortion. The interested reader may consult [1] for details about their computation. Regarding this reference, we intentionally did not consider rank-based measures: the curse of dimensionality was already handled through our kernel function choice.

4.3 Chosen data sets, and their usage

One synthetic and two real UCI data sets are used for our experiments (implemented with R):

- **Gaussian**: 3000 points generated from three loosely separated 2D Gaussian components. 1000 points are sampled from each component. A subsample of this data set has already been seen in figure 3.
- **Pima**: this data set was established from medical records of Pima Indian patients. It is defined by 8 numerical variables, and a binary class variable (i.e. presence or absence of diabetes). It contains 500 negative and 268 positive examples.
- **Isolet**: this data set was created from people recorded as they spoke isolated letters. These recordings are described by 617 numerical variables.

We extracted the vowel recordings: this amounts to 5 ground truth classes, with 300 elements in each class.

Each experiment first consists in picking a random sample without replacement from one of these data sets, with 100 elements sampled from each class (with the exception of the *Pima* negative examples, where we sample 200 elements for a better balance w.r.t. the original data set). The ground truth labels are ignored for all elements in the sample, but for a given number n_{lab} of elements per class, thus simulating user interaction. An experiment is parametrized by α (see equation (5)), and n_{lab}. We choose to allow $\alpha \in \{2, 3, 5, 10\}$, and $n_{\text{lab}} \in \{1, 2, 5, 10\}$. Let us note that the associated amount of semi-supervision then ranges in $[1\%, 10\%]$.

An experiment is also parametrized by a kernel transformation method, being either:

- **unsupervised**: when using the unsupervised p-Gaussian kernel function,
- **simple**: when using the reference semi-supervised approach [2],
- **neighbors**: when using the neighbor-sensitive semi-supervised kernel function from equation (5).

Compression and stretching distortion measures are computed for each experiment. In order to produce a single measure per experiment, we retain the median of the resulting compression (respectively stretching) distribution.

The projected data is then clustered, without any supervision, with a Gaussian mixture estimated from the Bayesian EM algorithm proposed in the VBmix R package [4]. Its posterior number of components serves as an estimate for our class-related quality metric. The resulting Gaussian mixture is used to infer class labels, and the cluster purity is computed by matching these labels to the ground truth.

An experimental condition is thus characterized by a tuple (data set, method, α, n_{lab}). For each condition, we perform 20 experiments. The clustering algorithm suffers from local minima issues. To alleviate this problem, for each experiment, we perform 10 runs of the clustering algorithm, and select the best model according to a BIC-like criterion.

compress	• The experimental contrast is marginally very significant ($p < 10^{-10}$, yet $p < 0.01$ only for *Isolet*). • α very significantly induces a linear trend ($p < 10^{-10}$). • n_{lab} weakly induces a linear trend ($p \simeq 10^{-3}$). • These trends seems to interact almost exclusively with the experimental contrast.
stretch	• The experimental contrast is marginally very significant ($p < 10^{-10}$). • α very significantly induces a linear trend ($p < 10^{-10}$). • n_{lab} weakly induces a linear trend ($p \simeq 10^{-3}$), more strongly with *Gaussian* ($p < 10^{-10}$). • These trends seems to interact almost exclusively with the experimental contrast.

Table 1: ANOVA results for distortion measures, aggregated by quality metric and data set.

5 Results and analysis

We performed a three-way independent ANOVA on our experimental results. The three independent variables are ordered as follows: projection method (further abbreviated as *method*), α, and n_{lab}. For *method*, we defined a *control contrast* between *unsupervised* and the grouped semi-supervised methods, and an *experimental contrast* between *simple* and *neighbors* methods. The polynomial contrast was applied to α and n_{lab}.

Many experimental conditions were associated to non-normal sets of values, or resulted in the failure of Levene's test for the homogeneity of variance. Yet, in the proposed experimental setup, all conditions are associated to the same number of values (i.e. 20), which ensures the robustness of ANOVA [6, 10].

This analysis was run independently for each data set and quality measure. The results are summarized in tables 1 and 2. Several conclusions can subsequently be drawn:

- The influence of the *simple* method on the projected topology is generally very weak.
- The analysis of the compression and stretching distortions shows that the proposed semi-supervised kernel function leads to drastic modifications of the 2D projection. These modifications are significant with as few as one labelled element per class, with mild effects to this respect when further adding labelled elements (see figure 4). This is a desirable property, as a user expects that her actions should have tangible effects.
- The distortions are highly influenced by the variation of α. Even with low alpha values, projection artifacts inherent to the kernel function are either alleviated (i.e. for stretching), or reinforced (i.e. for compression) (see figures 4 and 5). This tendency follows an strong linear trend, which emphasizes the sensitivity of α as a parameter.
- Augmenting α tends to diminish the number of classes, sometimes incidentally harming the cluster purity (see figures 6 and 7). Indeed, cluster purity is easier to achieve when using more clusters. More generally, semi-supervision would be expected to improve cluster purity: yet, due to the random selection of labelled elements, improvements are made at best w.r.t. the inferred number of clusters. Labels provided by users in an interactive context may help overcome this limitation.
- Adding more labelled elements seems visibly influential only on the *neighbors* method. The random (yet influent, as seen on figure 4) choice of labelled elements is harmful at first, but this handicap is alleviated when increasing $|\mathbf{X}_L|$ (figure 8). This also means that with the *neighbors* method, user action are tangible, which is a rather good property.

6 Conclusion

In this paper, a new semi-supervised, neighbor-sensitive, kernel transformation method was derived and evaluated. As the experiments show, very few labelled elements are sufficient to strongly influence a subsequent kernel PCA projection, hence potentially providing a tangible feedback to a user in a visualization context. The method was also shown to emphasize clusters in the 2D projection, a useful path to an easy visual characterization of such visual objects.

The automated labelling function used in the experimental section demonstrated the influence of α, the adjustable parameter of our method. Increasing

purity	• Both control and experimental contrasts (only experimental for *Isolet*) are significant ($p \simeq 10^{-2}$). • α moderately induces a linear trend ($p < 10^{-3}$). • n_{lab} significantly induces a linear trend ($p < 10^{-5}$), more weakly for *Pima* yet ($p < 0.1$). • Depending on the data set, there may be interaction effects between a linear trend w.r.t. α and the control contrast ($p < 0.1$ for all data sets), or the experimental contrast ($p < 10^{-5}$ for *Gaussian* and *Isolet*). • Except for *Pima* (weak interaction, $p < 0.1$), there is a strong interaction between the experimental contrast and a linear trend w.r.t. n_{lab} ($p < 10^{-6}$).
nclass	• The experimental contrast is marginally very significant ($p < 10^{-10}$). The control contrast is only weakly marginally significant for *Gaussian* and *Isolet* ($p < 0.1$). • α very significantly induces a linear trend ($p < 10^{-10}$), yet only moderately for *Isolet*. • n_{lab} has no marginal influence. • There is a significant interaction between the experimental contrast and a linear trend in α ($p < 10^{-3}$).

Table 2: ANOVA results for clustering measures, aggregated by quality metric and data set.

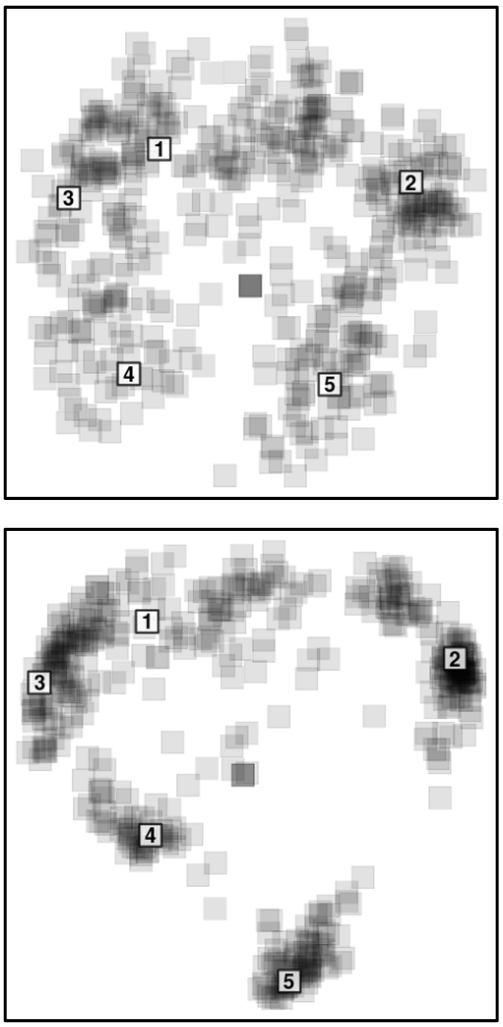

Figure 4: *Top*: projection of a subsample from *Isolet* with the *unsupervised* method. The shade of grey reveals the data density, and one element from each class is highlighted with distinctive digits.
Bottom: projection of the same subsample with the *neighbors* method, using the highlighted set as $\mathbf{X}_L$, and $\alpha = 3$ (see equation (5)). The group structure is emphasized, which results in higher compressive distortions.

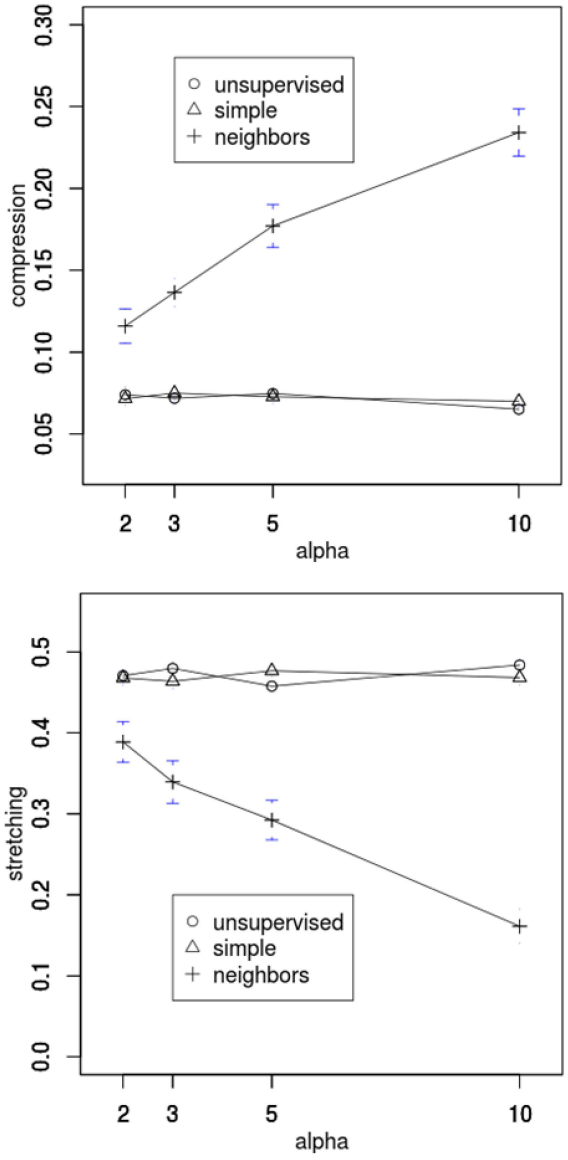

Figure 5: *Top*: influence of α on the compression measure, aggregated from our experiments on *Pima*. Confidence intervals are reported as error bars.
Bottom: influence of α on the stretching measure, for the same set of experiments.

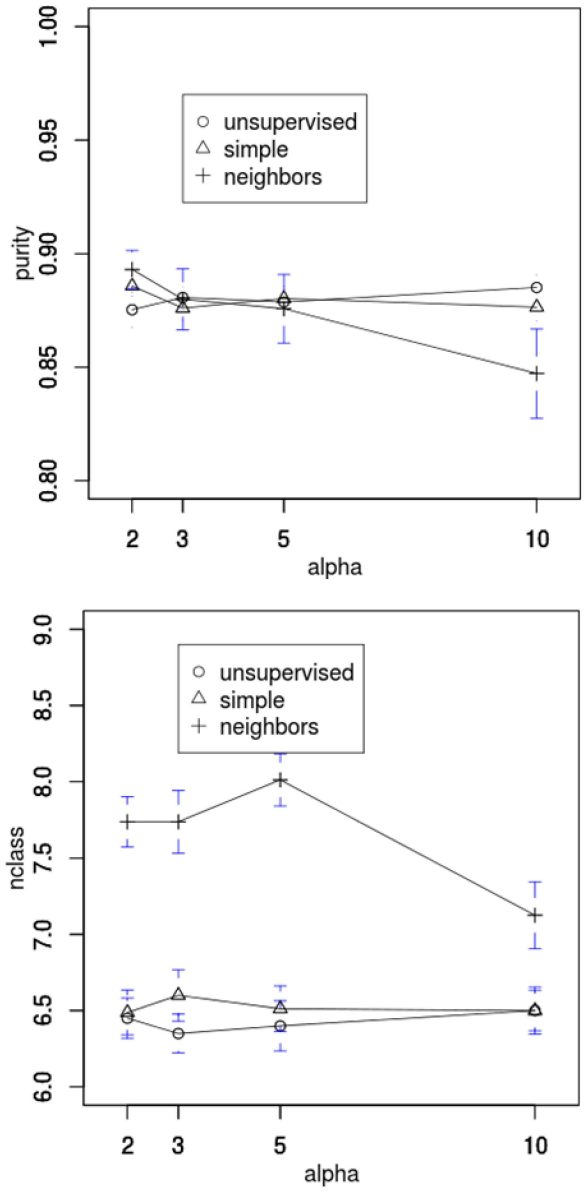

Figure 6: *Top*: influence of α on the cluster purity, aggregated from our experiments on *Isolet*.
Bottom: influence of α on the inferred number of classes, for the same set of experiments.

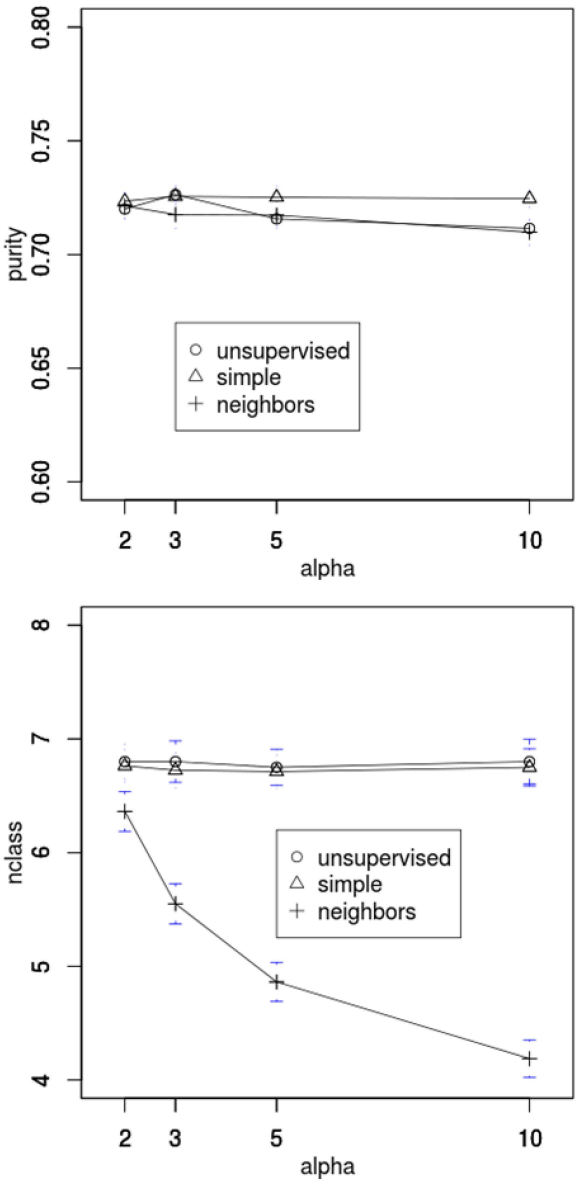

Figure 7: *Top*: influence of α on the cluster purity, aggregated from our experiments on *Pima*.
Bottom: influence of α on the inferred number of classes, for the same set of experiments.

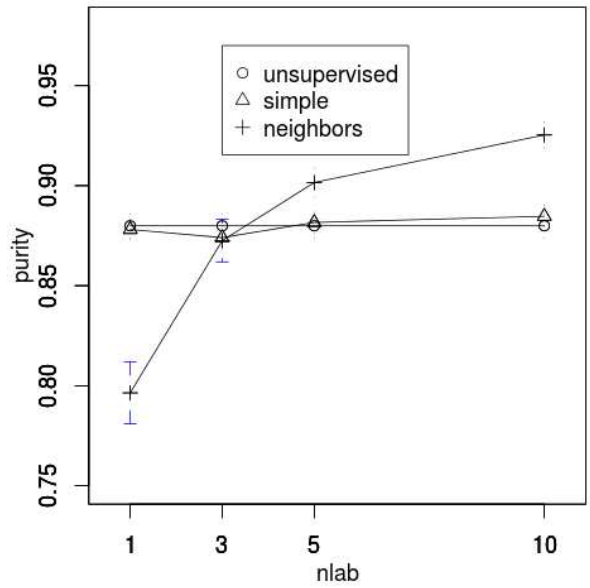

Figure 8: Influence of n_{lab} on the cluster purity, aggregated from our experiments on *Isolet*.

it led to simpler clustering models, but with more distortion artifacts, and unclear advantages from a cluster purity perspective. In an interactive context, its value may default to an intermediate value (e.g. 3), while providing the user with means to adjust it to her convenience.

Our method was shown to be highly sensitive to the labelled elements. This property suffers from a drawback: when as few as one element per ground truth class is also inappropriately selected (e.g. outlier in its class), the proposed technique may harm more than improve cluster purity. Yet, increasing the number of labelled elements quickly compensates this handicap.

This work attempted to describe and evaluate a novel semi-supervised projection method thorougly. It is intended to become a building block in an interactive visual clustering system, but we first strived to study its properties independently of human interaction-related considerations. It would of course be very instructive to include the proposed method in a fully interactive system, as sketched at the beginning of section 4.1. The ground truth would then be the user expert view of the data, and the performance of the method would be rated according to its ability to help a user achieve quickly and efficiently a clustering that conforms as much as possible to her contextual ground truth.

The general idea behind such a system would be to allow a user to label elements interactively, directly through the 2D projection, adapting the latter dynamically to these actions. Beyond defining the appropriates modes and temporal sequences of interaction, it is important to notice that the present work would not fit in such a scheme in its present state. Indeed, each interaction transforms the kernel matrix. If we consider a naive approach, computing the updated projection then requires, in practice, $O(N^3)$ operations. Some optimizations are possible, especially as we only need the two first eigenvalues and eigenvectors; but there should exist an algorithm (or at least, a heuristic) to more cleverly "pipeline" the transformations induced by equation (5) to the computation of the resulting projection using equation (2). This would allow to take interactions into account online, and update the projection with minimal computational overhead.

References

[1] M. Aupetit. Visualizing distortions and recovering topology in continuous projection techniques. *Neurocomputing*, pages 1304–1330, 2007.

[2] S. Basu, A. Banerjee, and R. Mooney. Semi-supervised clustering by seeding. *Proceedings of 19th International Conference on Machine Learning*, 2002.

[3] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[4] P. Bruneau. VBmix: a R package for Variational-Bayes mixture learning. Technical report, LINA (CNRS UMR 6241), 2012.

[5] O. Chapelle, J. Weston, and B. Schölkopf. Cluster kernels for semi-supervised learning. *Advances in Neural Information Processing Systems*, 15:585–592, 2003.

[6] T. S. Donaldson. Robustness of the F-test to errors of both kinds and the correlation between the numerator and denominator of the F-ratio. *Journal of the American Statistical Association*, pages 660–676, 1968.

[7] D. Francois, V. Wertz, and M. Verleysen. About the locality of kernels in high-dimensional spaces. *International Symposium on Applied Stochastic Models and Data Analysis*, pages 238–245, 2005.

[8] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon. Visual analytics: Definition, process, and challenges. In *Information Visualization*, pages 154–175. Springer, 2008.

[9] M. H. C. Law, A. Topchy, and A. K. Jain. Model-based clustering with probabilistic constraints. *Proceedings of SIAM Data Mining*, 2005.

[10] G. H. Lunney. Using analysis of variance with a dichotomous dependent variable: An empirical study. *Journal of Educational Measurement*, pages 263–269, 1970.

[11] D. J. Miller and D. J. Uyar. A mixture of experts classifier with learning based on both labelled and unlabelled data. *Advances in Neural Information Processing Systems*, 9:571–577, 1996.

[12] K. Nigam, A. McCallum, and T. Mitchell. Semi-supervised text classification using EM. *In Semi-Supervised Learning (Chapelle, Schölkopf and Zien eds.)*, 2006.

[13] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, pages 1299–1319, 1998.

[14] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[15] V. N. Vapnik. *The nature of statistical learning theory.* Springer, 1995.

[16] C. Ware. *Information Visualization: Perception for Design.* Elsevier, 2004.