Appendix: Colored Maximum Variance Unfolding

Le Song
NICTA and University of Sydney
Canberra, Australia
lesong@it.usyd.edu.au

Alex Smola
National ICT Australia
Canberra, Australia
alex.smola@nicta.com.au

Karsten Borgwardt
University of Cambridge
Cambridge, United Kingdom
kmb51@eng.cam.ac.uk

Arthur Gretton
MPI for Biological Cybernetics
Tübingen, Germany
arthur.gretton@tuebingen.mpg.de

A Influence of the Adjacency Matrices

An observation of our experiments is that MUHSIC considerably improves the embedding of the newsgroups and NIPS papers datasets, but its improvement over MVU on USPS digits dataset seems to be minor. Except for clearer separation between classes and an embedding with lower dimension, the overall visualization remains very similar to that by MVU and PCA. To investigate this, we plotted the adjacency matrix of the corresponding nearest neighbor graph in Figure 1.

We find that the nearest neighbor graphs for newsgroups and NIPS papers datasets are noisier with no clear block-diagonal structure, whereas the USPS digits dataset has an almost block-diagonal form. Also note that for newsgroups and NIPS papers dataset, several data points almost have all other data points as nearest neighbors. Second, while we have ordered data points from the same class in contiguous places for both USPS digits and newsgroups datasets, only the adjacency matrix of USPS digits dataset show clear correspondence with the class labels, that is, only the USPS digits dataset exhibits a clear block structure. In this case clearly the additional labels do not convey much additional information over the similarity matrix between observations and it is not too surprising that in this case MUSIC generates results not much different from MVU. This suggests that MUHSIC may provide improvement in cases where:

1. The nearest neighbor information is inexact.
2. The side information provides complementary information.

It also indicates that wherever the dominant features of the data are present in the nearest neighbor graph MVU will be able to recover them. However, in general, it is not clear that the desired properties are necessarily those which are dominant. For instance a linguist might care more about the date, length and vocabulary diversity of the documents rather than their topics (or vice versa).

B Influence of the Local Refinement Step

As pointed out [1] it is preferable to perform gradient descent on the embedding after solving the low-dimensional approximation of the overall optimization problem. The latter allows for visually more appealing low-dimensional representations. In this sense, our implementation (which is based on that by [1]) shares the same properties. Figures 2 and 3 visualize this fact quite clearly.

The experiments show that while both MVU and MUHSIC strive to generate a low-dimensional embedding which preserves local distance information, explicit side-information used for MUHSIC ensures that after the initial guess of the subspace which is used to keep optimization tractable, the algorithm finds the more representative subspace suitable for visualization. Note that while

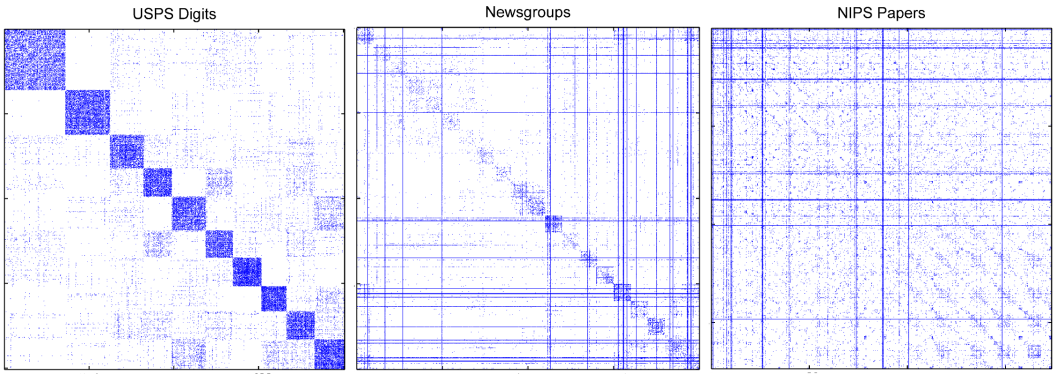

Figure 1: Adjacency matrices of the nearest neighbor graphs for the three datasets. We used the Euclidean distance between the vector space representations of the USPS digits, and the TF.IDF representations of the Newsgroups and NIPS papers datasets. 1% of the data were chosen as nearest neighbors and the graphs were symmetrized subsequently.

the full semidefinite program and the low-dimensional semidefinite program are both convex, a *dimensionality-constrained* version of the full problem is clearly not, hence the need for a local refinement step. (We will refer to the initial solutions of the unfolding algorithms before the gradient refinement as MUHSIC$^-$ and MVU$^-$ respectively.)

C Comparison to Other Methods

[2] propose Neighborhood Component Analysis (NCA) for extracting low-dimensional representations of data which is optimized for classification. More specifically they minimize a smoothed-out variant of the k-nearest neighbor classification error (the exact k-nearest neighbor error is a piecewise constant function and hence intractable for optimization). That is, the optimization is carried out *directly* on the features of the data. This is advantageous insofar as it generates a direct projection of the data onto a lower-dimensional space which is thought to be representative for the problem. Such a representation can be very convenient at test time.

At the same time, this setting has several drawbacks: firstly, it is restricted to Euclidean spaces underpinning the space of observations, which makes nontrivial Banach space distances, such as [3] inapplicable. Secondly, it being a nonconvex method, optimization may become stuck in local optima, thereby rendering the generation of a specific representation somewhat irreproducible. The main drawback, however, is that computational cost increases with the dimensionality of the data. In practice this means that we were unable to apply NCA to datasets other than the USPS digits dataset, since the dimensionality of the features and the sample size were just too high.[1]

Two other algorithms used for comparison purposes were Relevant Component Analysis (RCA) by [4] and Linear Discriminant Analysis (LDA) by [5]. Examples of their performance on the USPS dataset are given in Figure 4. Arguably NCA performs best on this dataset.

To obtain a more quantifiable measure of the performance of low-dimensional representations of the data we computed the nearest neighbor scores produced by various embedding algorithms (the nearest neighbor score computes the percentage of the data points in the embedding that has data point from the same class as the nearest neighbor). The performance is given in Table 1 below. We can see that (not very surprisingly) MUHSIC outperforms MVU. More surprising is that it also outperforms RCA and NCA in most problems.

We visualize the embeddings generated by the different methods on two datasets (DNA and SVMguide2) in Figures 5 and 6. These are further examples where MUHSIC manages to separate different classes particularly well.

[1] The algorithm kindly provided by [2] did not run successfully on the newsgroups dataset.

Table 1: Nearest neighbor scores in % for various multiclass datasets produced by various methods. The sizes of the datasets are listed as triples: (size of dataset, number of dimensions, number of classes). We typically used k =1% of the data points as nearest neighbor for MVU and MUHSIC. In the case that the resulting nearest neighbor graph is not connected, we increase the neighbor size to 2%. Furthermore, we typically used the top n =10 eigenvectors of the graph Laplacian as the bases for optimizing MVU and MUHSIC. In the case that the dimension of the data is small (≤ 100), we decrease the number of bases used to 5.

Dataset	Size	k (%)	n	PCA	LDA	RCA	NCA	MVU	MVU^-	MUHSIC	MUHSIC^-
USPS	(2007, 256, 10)	1	10	43.9	50.2	50.0	66.2	49.8	59.4	59.4	**71.2**
Wine	(178, 13, 3)	2	5	96.6	**97.2**	**97.2**	**97.2**	95.5	93.8	93.8	94.4
Satimage	(1331, 36, 6)	1	5	75.7	77.3	77.1	77.1	78.4	79.0	**79.1**	78.4
Segment	(2310, 19, 7)	2	5	77.9	82.6	83.3	83.3	82.6	**87.8**	84.5	87.1
Vehicle	(846, 10, 11)	1	5	51.9	45.7	46.2	46.2	42.7	50.1	**57.7**	54.0
DNA	(2000, 180, 3)	1	10	70.5	88.9	88.9	92.4	54.4	60.8	**95.6**	63.9
Vowel	(528, 10, 11)	2	5	52.8	67.1	65.3	65.3	**72.5**	66.5	70.1	44.9
SVMguide2	(391, 20, 3)	1	5	56.0	70.1	67.5	67.5	61.4	62.9	**79.0**	60.1

D Embedding this Paper among other NIPS Papers

We also embedded the current paper into the visualization of the NIPS papers in the main text. Basically, we represent the current paper as a TF.IDF vector and then place it in the location of its nearest neighbor among the NIPS papers.

To do this, we first represent this paper using the TF.IDF vector with the model we obtained from the NIPS papers dataset (We produced the TF vector for the main text, and then computed the TF.IDF presentation using the vocabulary and IDF obtained from the NIPS paper dataset). Since the nearest neighbor graph of the NIPS paper dataset is noisy (some data points have almost all other papers as neighbors), we excluded those papers which have more than 3% of the papers as neighbors (Note when we build the nearest neighbor graph, we only require 1% of the data points).

E A General Feature Selection Framework

Below we give a short list of some more feature selection and generation algorithms which can be viewed as special instances of the dependence maximization framework.

Principal Component Analysis Assume that we want to find a d-dimensional Euclidean embedding of the data by means of a projection matrix. That is, we want to find an idempotent ($\mathbf{KK} = \mathbf{K}$) positive semidefinite matrix $\mathbf{K}$ with $\mathrm{rank}\mathbf{K} = d$. In this case the optimization problem

$$\underset{\mathbf{K}}{\text{maximize}} \ \operatorname{tr}\mathbf{HKHL} \text{ subject to } \mathbf{KK} = \mathbf{K} \text{ and } \operatorname{rank}\mathbf{K} = d \tag{1}$$

is solved by choosing the subspace of the leading d eigenvectors of $\mathbf{HLH}$. Since PCA requires centering of the data first, $\mathbf{L}$ is naturally centered, which means that PCA and feature extraction subject to rank constraints are equivalent.

Kernel Principal Component Analysis A simple modification allows us to recover kernel-PCA: simply allow arbitrary kernels for $\mathbf{L}$ in the above optimization problem, since $\mathbf{K}$ will always choose the leading d principal components of the corresponding matrix.

Clustering If we restrict $\mathbf{K}$ to be of the form $\Pi^\top \mathbf{D}\Pi$, where $\mathbf{D} \in \mathbb{R}^{k\times k}$ is a diagonal matrix with $\mathbf{D} \succeq 0$ and $\Pi \in \{0,1\}^{k\times m}$ is an assignment matrix, i.e. $\Pi^\top \mathbf{1} = \mathbf{1}$, we have a clustering problem. In fact, this problem is well studied in theoretical computer science [6, 7] as the so-called multicut problem. A simplified version of this is known in machine learning as the min-cut and normalized min-cut problem [8, 9]. What HSIC does is provide an information-theoretical footing for these problems.

References

[1] K. Weinberger, F. Sha, Q. Zhu, and L. Saul. Graph laplacian regularization for large-scale semidefinte programming. In *Neural Information Processing Systems*, 2006.

[2] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood component analysis. In *Advances in Neural Information Processing Systems 17*, 2004.
[3] Y. Rubner, C. Tomasi, and L.J. Guibas. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vision*, 40(2):99–121, 2000.
[4] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relation. In *Proc. Intl. Conf. Machine Learning*, 2003.
[5] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.
[6] S. Chawla, R. Krauthgammer, R. Kumar, Y. Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. *Computational Complexity*, 15(2):94–114, 2006.
[7] D. Emanuel and A. Fiat. Correlation clustering - minimizing disagreements on arbitrary weighted graphs. In G. Di Battista and U. Zwick, editors, *Algorithms - ESA 2003, 11th Annual European Symposium*, volume 2832 of *Lecture Notes in Computer Science*, pages 208–220. Springer, 2003.
[8] Y. Weiss. Segmentation using eigenvectors: A unifying view. In *International Conference on Computer Vision ICCV*, pages 975–982, 1999.
[9] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

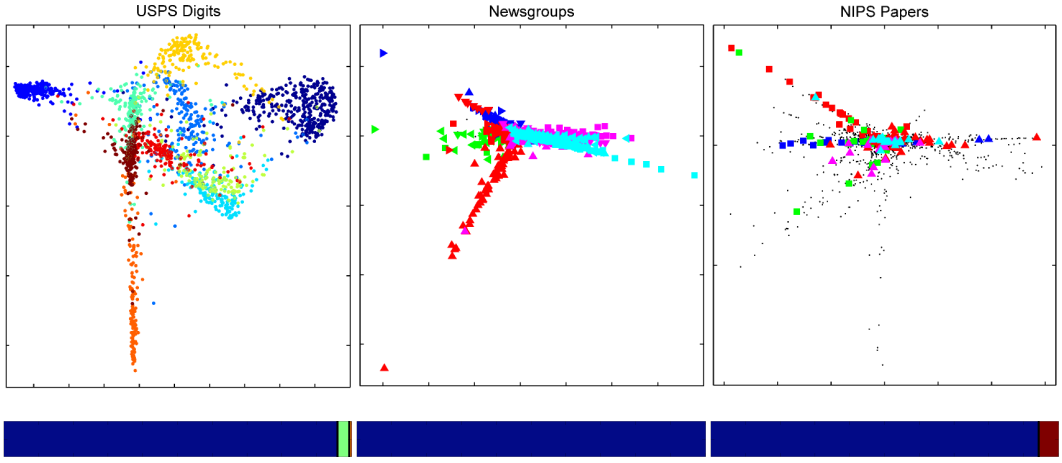

Figure 2: Embedding of the three datasets produced by MUHSIC *without* the refinement via gradient descent (MUHSIC^{-}). Colors of the dots are used to denote digits from different classes. The color bar below each figure shows the eigenspectrum of the learned kernel matrix $\mathbf{K}$.

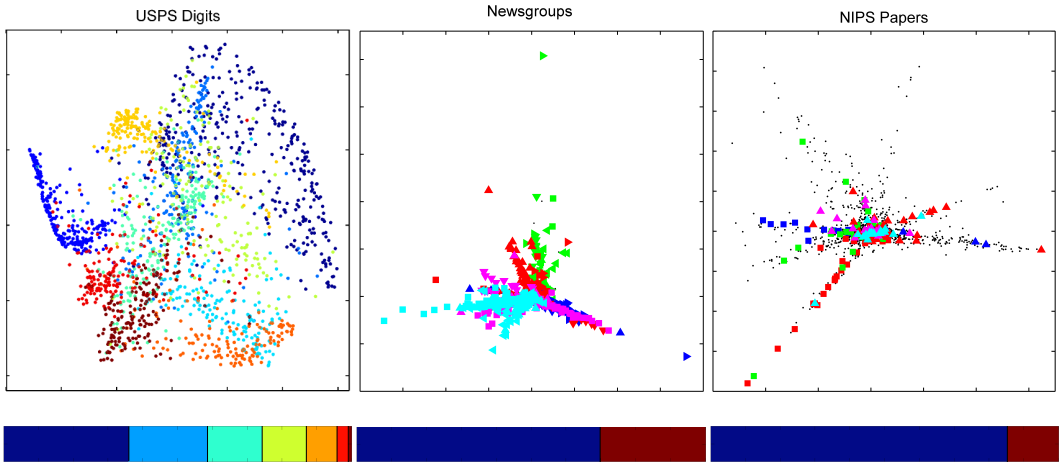

Figure 3: Embedding of the three datasets produced by MVU *without* the refinement via gradient descent (MVU^{-}). Colors of the dots are used to denote digits from different classes. The color bar below each figure shows the eigenspectrum of the learned kernel matrix $\mathbf{K}$.

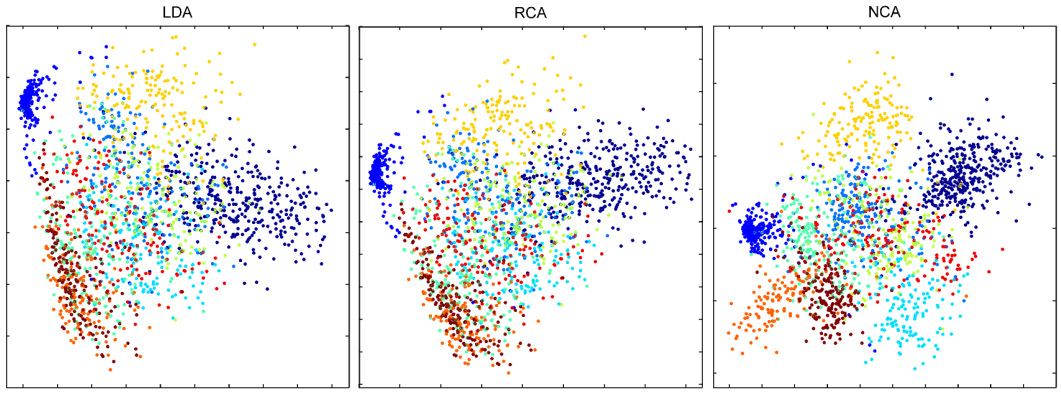

Figure 4: Embedding of 2007 USPS digits produced by LDA, RCA and NCA methods. The same color scheme is used as that for MUHSIC. These methods directly learn a 2 dimensional projection, so no eigenspectrum of $\mathbf{K}$ is shown.

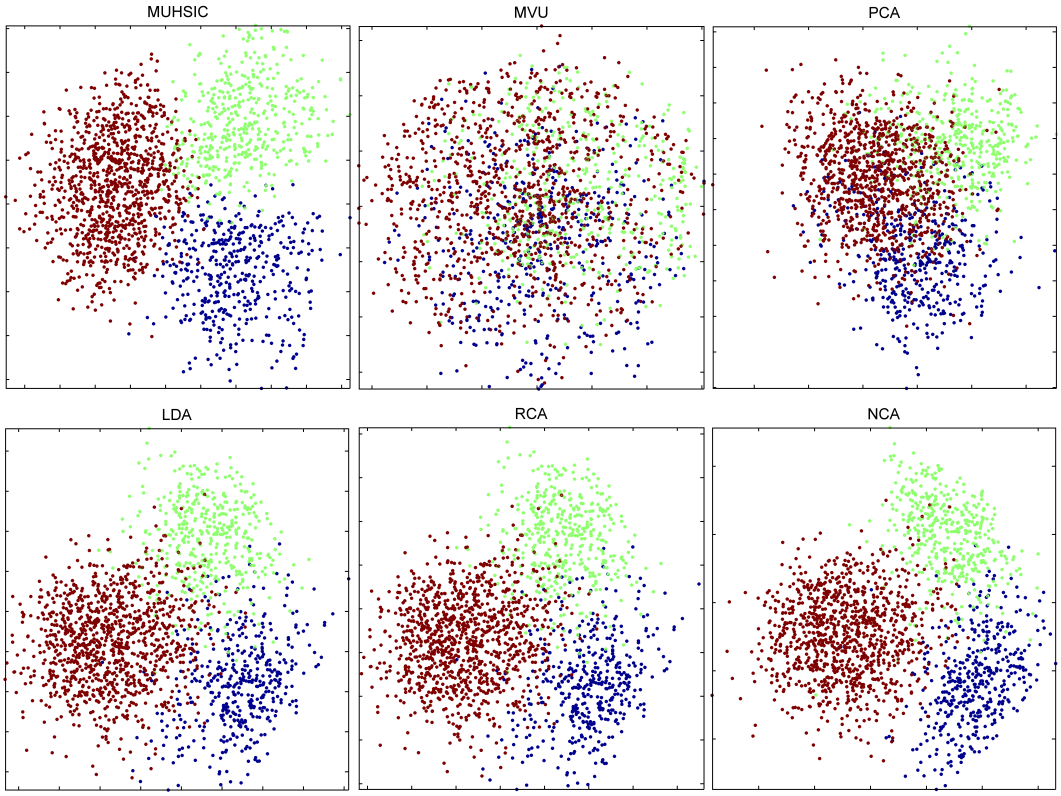

Figure 5: The embeddings of the DNA dataset produced by various methods.

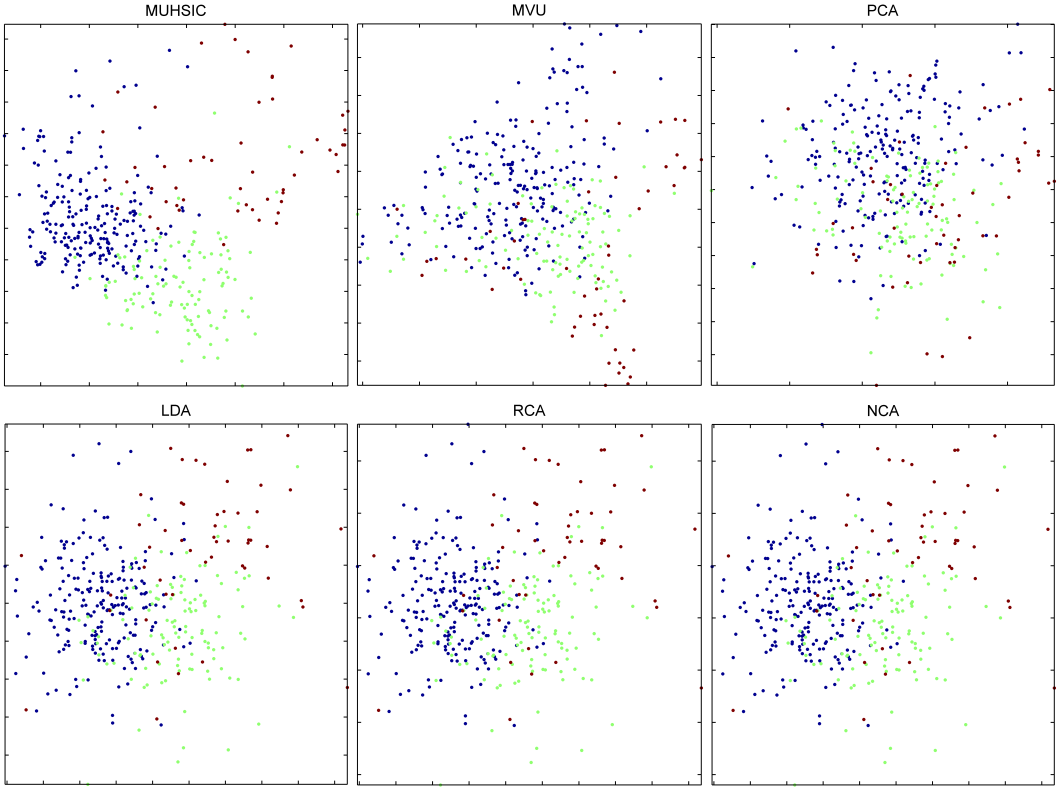

Figure 6: The embeddings of the SVMguide2 dataset produced by various methods.