ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

Discriminative Dimensionality Reduction in Kernel Space

Alexander Schulz and Barbara Hammer *

CITEC centre of excellence,
Bielefeld University, Germany

Abstract. Modern nonlinear dimensionality reduction (DR) techniques enable an efficient visual data inspection in the form of scatter plots, but they suffer from the fact that DR is inherently ill-posed. Discriminative dimensionality reduction (DiDi) offers one remedy, since it allows a practitioner to identify what is relevant and what should be regarded as noise by means of auxiliary information such as class labels. Powerful DiDi methods exist, but they are restricted to vectorial data only. In this contribution, we extend one particularly promising approach to non-vectorial data characterised by a kernel. This enables us to apply discriminative dimensionality reduction to complex, possibly discrete or structured data.

1 Introduction

Modern nonlinear dimensionality reduction (DR) techniques enable an intuitive and highly efficient visual inspection of dominant characteristics of given data sets, with striking applications e.g. in biomedical data analysis [1, 7, 8, 15, 18, 21]. While their nonlinearity constitutes a crucial prerequisite for their success, their high flexibility causes the risk to display spurious aspects of the data rather than relevant information especially for high-dimensional or noisy data. In general, DR constitutes an ill-posed problem whenever data dimensionality is higher than the projection space (usually two); correspondingly, the results of DR technologies severely differ depending on the used method and its parameterisation.

Discriminative dimensionality reduction (DiDi) offers a very intuitive way to regularise DR technology, such that only those aspects of the data are displayed, where the applicant is inherently interested in. The applicant specifies auxiliary information such as class labels; then DiDi methods subtract all information irrelevant to those aspects from the visual display. The result enables an answer to crucial questions such as: Do data include any information which relates to the given classes? Does the data representation include enough information to robustly separate these classes? Do there exist mis-labelings in the data? Interestingly, this idea can be used to visualise full classifiers [17].

One particularly powerful general DiDi technology is based on the Riemannian tensor induced by the local Fisher information matrix [4, 14]. Like most DiDi methods, however, it is restricted to vectorial data, and it is not applicable whenever complex, non-vectorial data structures are dealt with. In this contribution, we provide an extension of the Fisher metric to a general kernel space, this way enabling powerful DiDi technologies for general data structures which are described in terms of pairwise relations, the kernel matrix, only. We demonstrate the feasibility of the approach for several benchmarks, including complex structured data from the domains of music and java programming.

*Funding from DFG under grant number HA2719/7-1 and by the CITEC center of excellence (EXC277) is gratefully acknowledged.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

2 Fisher Metric

DR is concerned with a projection of high-dimensional data $\mathbf{x} \in X = \mathbb{R}^d$ to low-dimensional counterparts $\pi(\mathbf{x}) = \mathbf{y} \in Y = \mathbb{R}^2$ such that as much information as possible is preserved. For DiDi, auxiliary information in the form of labels $c = c(\mathbf{x})$ is available, where c is element of a finite number of class labels. The goal is to emphasise those aspects of the data $\mathbf{x}$ in the display which are relevant for c. A key observation consists in the fact that popular DR methods rely on pairwise distances of data only, i.e. auxiliary information can easily be integrated by changing the metric according to the labels c. This idea yields consistently superior results as compared to other techniques [20] and is applicable to a wide range of DR techniques [17]. Hence, we focus our investigations on it.

Locally at a given point $\mathbf{x}$, the information contained in $\mathbf{c}$ is taken into account by a linear scaling of the tangent space according to the Fisher information matrix

$$\mathbf{J}(\mathbf{x}) = E_{p(c|\mathbf{x})}\left\{\left(\frac{\partial}{\partial \mathbf{x}}\log p(c|\mathbf{x})\right)\left(\frac{\partial}{\partial \mathbf{x}}\log p(c|\mathbf{x})\right)^{\top}\right\}, \tag{1}$$

where $p(c|\mathbf{x})$ denotes the probability of the class information c conditioned on $\mathbf{x}$. This induces a Riemannian tensor and corresponding Riemannian distances

$$d_M(\mathbf{x}, \mathbf{x}') = \inf_P \int_0^1 \sqrt{P'(t)^{\top}\mathbf{J}(P(t))P'(t)}dt \tag{2}$$

where the infimum is over all differentiable paths $P : [0, 1] \to X$ with start $P(0) = \mathbf{x}$ and end $P(1) = \mathbf{x}'$. The resulting values d_M can be directly plugged into any distance-based DR method. Since the integral (2) is intractable, it is usually approximated by equidistant points $\mathbf{x}_1 = \mathbf{x}, \ldots, \mathbf{x}_{T+1} = \mathbf{x}'$ on the straight line

$$d_T(\mathbf{x}, \mathbf{x}') = \sum_{t=1}^{T} \sqrt{(\mathbf{x}_{t+1} - \mathbf{x}_t)^{\top} J(\mathbf{x}_t)(\mathbf{x}_{t+1} - \mathbf{x}_t)}. \tag{3}$$

The conditional probability is approximated by a non-parametric Parzen window estimator

$$\hat{p}(c|\mathbf{x}) = \frac{\sum_i \delta_{c=c_i}\exp(-0.5\|\mathbf{x} - \mathbf{x}_i\|^2/2\sigma^2)}{\sum_j \exp(-0.5\|\mathbf{x} - \mathbf{x}_j\|^2/2\sigma^2)} \tag{4}$$

with bandwidth σ, which yields $\mathbf{J}(\mathbf{x}) = E_{\hat{p}(c|\mathbf{x})}\left\{\mathbf{b}(\mathbf{x}, c)\mathbf{b}(\mathbf{x}, c)^{\top}\right\}/\sigma^4$ where $\mathbf{b}(\mathbf{x}, c) = E_{\xi(i|\mathbf{x},c)}\{\mathbf{x}_i\} - E_{\xi(i|\mathbf{x})}\{\mathbf{x}_i\}$ with empirical expectation E and

$$\xi(i|\mathbf{x}, c) = \frac{\delta_{c=c_i}\exp(-0.5\|\mathbf{x} - \mathbf{x}_i\|^2/2\sigma^2)}{\sum_j \delta_{c=c_j}\exp(-0.5\|\mathbf{x} - \mathbf{x}_j\|^2/2\sigma^2)} \tag{5}$$

$$\xi(i|\mathbf{x}) = \frac{\exp(-0.5\|\mathbf{x} - \mathbf{x}_i\|^2/2\sigma^2)}{\sum_j \exp(-0.5\|\mathbf{x} - \mathbf{x}_j\|^2/2\sigma^2)} \tag{6}$$

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

3 Kernelisation

We assume that data are characterised in terms of pairwise similarities only, i.e. a matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ is given with N being the number of data, entries are denoted as s_{ij}. We assume symmetry of $\mathbf{S}$, such that an implicit vectorial embedding exists [5]. Further, we require non-negativity of the values to guarantee a valid probability distribution. In particular, this covers the case of structure kernels for complex data structures [11]. However, we will see in experiments that the Fisher metric also provides reasonable results for general matrices. We denote data in kernel space as $\mathbf{x}_i$ where $s_{ij} = \mathbf{x}_i^\top \mathbf{x}_j$. Equidistant points on the line from $\mathbf{x}_i$ to $\mathbf{x}_j$ have the form $(1-\alpha)\mathbf{x}_i + \alpha\mathbf{x}_j$ where $\alpha = (t-1)/T$ for $t \in \{1, \ldots, T+1\}$, hence differences of consecutive points have the form $(\mathbf{x}_j - \mathbf{x}_i)/T$. Thus denoting $\mathbf{x}(t) := (1-\alpha)\mathbf{x}_i + \alpha\mathbf{x}_j$, distances $d_T(\mathbf{x}_i, \mathbf{x}_j) \cdot (T\sigma^2)$ consist of terms of the form

$$\sigma^4 (\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{J}(\mathbf{x}(t))(\mathbf{x}_i - \mathbf{x}_j) = \sum_c \hat{p}(c|\mathbf{x}(t)) \left(\mathbf{x}_i^\top \mathbf{b}(\mathbf{x}(t), c) - \mathbf{x}_j^\top \mathbf{b}(\mathbf{x}(t), c)\right)^2 \quad (7)$$

where

$$\mathbf{x}_i^\top \mathbf{b}(\mathbf{x}(t), c)) = \sum_l \left(\xi(l|\mathbf{x}(t), c) \cdot \underbrace{\mathbf{x}_i^\top \mathbf{x}_l}_{s_{il}} - \xi(l|\mathbf{x}(t)) \cdot \underbrace{\mathbf{x}_i^\top \mathbf{x}_l}_{s_{il}}\right) \quad (8)$$

The terms $\hat{p}(c|\mathbf{x}(t))$, $\xi(l|\mathbf{x}(t), c)$, and $\xi(l|\mathbf{x}(t))$ can be expressed in terms of Gaussians with argument $\|\mathbf{x}(t) - \mathbf{x}_l\|^2 = (1-\alpha)^2 s_{ii} + \alpha^2 s_{jj} + s_{ll} + 2(1-\alpha)\alpha s_{ij} - 2(1-\alpha)s_{il} - 2\alpha s_{jl}$, hence, the full computation can be kernelised.

4 Experiments

Our reformulation of Fisher distance computations in terms of kernels does not rely on approximations and, hence, is equivalent to the vectorial computation if the similarity matrix $\mathbf{S}$ is given by a standard scalar product. Hence, we do not present comparisons to the vectorial case, here.

Instead, we evaluate the method for six benchmark data sets that are only given as similarity matrices and are not necessarily euclidean.

Aural Sonar [16]: Data consist of 100 returns from a broadband active sonar system, their similarity is evaluated by human experts. Two classes (target of interest versus clutter) are distinguished.

Patrol [2]: 241 members of seven patrol units are characterised by (partially faulty) feedback of unit members naming five colleagues each.

Protein [6]: 226 globin proteins are compared based on their evolutionary distances, four classes of different protein families result.

Voting [2, 10]: 435 either republican or democrat candidates are characterised by 16 nominal attributes which characterise the key votes identified by the CQA, the value difference metric is used for comparison.

Java Programs [12, 13]: 64 Java programs which implement bubble sort or insertion sort, respectively, have been retrieved from the internet. They are compiled with the Oracle Java Compiler API and compared by alignment.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

Table 1: Average 1-NN classification errors in percent with standard deviations; sum of the negative EVs in relation to the summed absolute values of the EVs.

	AuralS	Patrol	Protein	Voting	Java	Sonatas
original data (clip)	17	19	10	6	11	11
t-SNE (clip)	15 (±2)	16 (±1)	8 (±1)	7 (±1)	13 (±2)	9 (±1)
Fisher t-SNE (clip)	9 (±1)	11 (±1)	3 (±1)	4 (±1)	11 (±2)	6 (±1)
original data	21	7	77	6	14	13
t-SNE	18 (±2)	87 (±1)	31 (±6)	7 (±1)	15 (±2)	10 (±1)
Fisher t-SNE	10 (±3)	15 (±1)	4 (±0)	6 (±1)	14 (±2)	6 (±1)
baseline (clip)	40	81	48	43	45	49
perc. negative Eigs	21	50	20	0	8	2

Sonatas [3]: 1068 sonatas in MIDI format from the online collection *Kunst der Fuge* are transformed to graph structures and compared with the normalised compression distance of their paths, labelling is given by one of 5 composers from the classical / baroque era.

A more detailed description of the data can be found in [2, 3].

Each data set is characterised in terms of a symmetrised similarity matrix $\mathbf{S}$. All data are projected to two dimensions based on t-Distributed Stochastic Neighbor Embedding (t-SNE) [19]. We compare the result of a projection of t-SNE, which is directly applied to the dissimilarity matrix as induced by $\mathbf{S}$, and the dissimilarity matrix computed from the Fisher metric. We denote the former step as t-SNE and the latter as Fisher t-SNE, for short. Note that some of the data matrices $\mathbf{S}$ do not relate to valid kernels, i.e. have negative Eigenvalues (EVs). Therefore, we compare the result achieved with plain data $\mathbf{S}$ and its clip-based eigenvalue correction [2, 5]. Notably, the Fisher metric does not encounter numerical difficulties when addressing the plain data, while t-SNE does.

Besides the visual impression, we compare the methods by a 1-nearest neighbour (1-NN) classification in the projection space. Thereby we also report the result which we obtain when applying Fisher t-SNE to data with randomly permuted labels, which corresponds to the quality which is merely due to statistical effects of the data. We refer to the 1-NN error in this setting as a baseline. Note that it is not reasonable to evaluate the projections by the quality framework [9] since we do not aim to preserve neighbourhoods based on euclidean distances.

For the computation of distances in the Fisher metric, the parameter σ for the Parzen window estimate has to be specified. In order to find an appropriate value, we compute bandwidths using the perplexity based idea as in [18], and average those to obtain a single bandwidth value.

Since t-SNE is not deterministic, we run the t-SNE algorithm 10 times on the respective distance matrix. The averaged leave-one-out 1-NN errors for the six data sets are displayed in Table 1, with standard deviations depicted in brackets. If clipping is applied, this is stated behind the method name. For the clipped Eigenspectrum of $\mathbf{S}$, the 1-NN errors of both t-SNE and Fisher t-SNE are comparably low (see e.g. [2]). Further, the discriminative projections have an even lower classification error, on average. The comparably high baseline error indicates that Fisher t-SNE does not neglect the intrinsic structure of the data when the task is to embed a random class distribution.

Based on the clipped Eigenspectrum, an instance of each embedding is shown

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

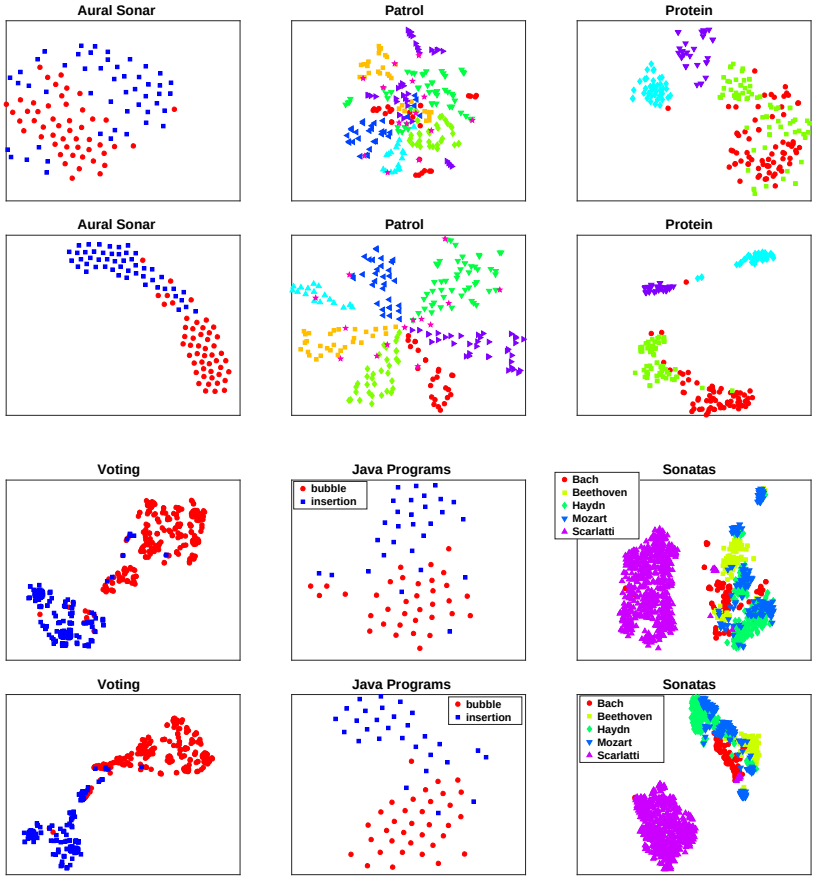

Fig. 1: Unsupervised t-SNE projections in rows one and three of the data sets Aural Sonar, Patrol, Protein, Voting, Java Programs and Sonatas. Rows two and four contain the according supervised Fisher t-SNE projection.

in Fig. 1. For each data set, a t-SNE projection is shown in rows one and three, a Fisher t-SNE mapping in rows two and four.

In addition to the numerical evaluation, these visualisations show that the Fisher Information based projections have a clearer class separability and, hence, enable the user to get a better understanding of the data. The unsupervised projection of the Protein data set, for instance, suggests that two classes are strongly overlapping. Here, the discriminative visualisation, which emphasises local directions that are relevant for class separation, shows that both classes have only a few overlapping points. Another example constitutes the Patrol data set, where the Fisher t-SNE embedding shows a clear class structure with only few noisy points coming from a specific class.

Another interesting aspect in Table 1 is the classification performance on the original data, without clipping. While t-SNE suffers from a large accuracy loss for the Patrol and Protein data sets, Fisher t-SNE obtains stable results with only a slight performance decrease. Particularly the Patrol data set has large

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

negative eigenvalues, as can be seen in Table 1.

5 Conclusion

In this contribution we have reformulated one particularly popular approach for discriminative dimensionality reduction such that it is applicable to non-vectorial data only given by (dis-)similarities. We evaluated this method with six data sets from this domain and obtained a clear improvement as compared to unsupervised projections in many cases. The robustness of Fisher t-SNE towards indefinite proximities seems interesting and requires further investigation.

References

[1] K. Bunte, M. Järvisalo, J. Berg, P. Myllymäki, J. Peltonen, and S. Kaski. Optimal neighborhood preserving visualization by maximum satisfiability. In *AAAI*, pages 1694–1700, 2014.

[2] Y. Chen, E. K. Garcia, M. R. Gupta, A. Rahimi, and L. Cazzanti. Similarity-based classification: Concepts and algorithms. *JMLR*, 10:747–776, 2009.

[3] A. Gisbrecht, B. Mokbel, and B. Hammer. Relational generative topographic mapping. *Neurocomputing*, 74(9):1359–1371, 2011.

[4] A. Gisbrecht, A. Schulz, and B. Hammer. Parametric nonlinear dimensionality reduction using kernel t-sne. *Neurocomputing*, 147:71–82, 2015.

[5] B. Hammer, D. Hofmann, F. Schleif, and X. Zhu. Learning vector quantization for (dis-)similarities. *Neurocomputing*, 131:43–51, 2014.

[6] T. Hofmann and J. M. Buhmann. Pairwise data clustering by deterministic annealing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(1):1–14, 1997.

[7] S. Kaski and J. Peltonen. Dimensionality reduction for data visualization [applications corner]. *IEEE Signal Process. Mag.*, 28(2):100–104, 2011.

[8] C. C. Laczny, N. Pinel, N. Vlassis, and P. Wilmes. Alignment-free visualization of metagenomic data by nonlinear dimension reduction. *Scientific Reports*, 4:4516 EP –, 03 2014.

[9] J. A. Lee and M. Verleysen. Scale-independent quality criteria for dimensionality reduction. *Pattern Recognition Letters*, 31(14):2248–2257, 2010.

[10] M. Lichman. UCI machine learning repository, 2013.

[11] G. D. S. Martino and A. Sperduti. Mining structured data. *IEEE Comp. Int. Mag.*, 5(1):42–49, 2010.

[12] B. Paaßen. Java Sorting Programs, doi: 10.4119/unibi/2900684, 2016.

[13] B. Paaßen, B. Mokbel, and B. Hammer. Adaptive structure metrics for automated feedback provision in Java programming. In M. Verleysen, editor, *Proceedings of the ESANN*, 2015.

[14] J. Peltonen, A. Klami, and S. Kaski. Improved learning of riemannian metrics for exploratory analysis. *Neural Networks*, 17(8-9):1087–1100, 2004.

[15] D. H. Peluffo-Ordóñez, J. A. Lee, and M. Verleysen. Recent methods for dimensionality reduction: A brief comparative analysis. In *ESANN*, 2014.

[16] S. Philips, J. Pitton, and L. Atlas. Perceptual feature identification for active sonar echoes. In *OCEANS 2006*, pages 1–6, Sept 2006.

[17] A. Schulz, A. Gisbrecht, and B. Hammer. Using discriminative dimensionality reduction to visualize classifiers. *Neural Processing Letters*, 42(1):27–54, 2015.

[18] L. van der Maaten. Accelerating t-sne using tree-based algorithms. *Journal of Machine Learning Research*, 15(1):3221–3245, 2014.

[19] L. van der Maaten and G. E. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[20] J. Venna, J. Peltonen, K. Nybo, H. Aidos, and S. Kaski. Information retrieval perspective to nonlinear dimensionality reduction for data visualization. *JMLR-10*, 11:451–490, 2010.

[21] M. Verleysen and J. A. Lee. Nonlinear dimensionality reduction for visualization. In *ICONIP*, pages 617–622, 2013.