Short Review of Dimensionality Reduction Methods Based on Stochastic Neighbour Embedding

Diego H. Peluffo-Ordóñez[1,*], John A. Lee[1,2], and Michel Verleysen[1]

[1] Machine Learning Group - ICTEAM,
Université catholique de Louvain, Machine Learning Group - ICTEAM, Place du Levant 3,
B-1348 Louvain-la-Neuve, Belgium
[2] Molecular Imaging Radiotherapy and Oncology - IREC,
Université catholique de Louvain, Avenue Hippocrate 55, B-1200 Bruxelles, Belgium
{diego.peluffo,john.lee,michel.verleysen}@uclouvain.be

Abstract. Dimensionality reduction methods aimed at preserving the data topology have shown to be suitable for reaching high-quality embedded data. In particular, those based on divergences such as stochastic neighbour embedding (SNE). The big advantage of SNE and its variants is that the neighbor preservation is done by optimizing the similarities in both high- and low-dimensional space. This work presents a brief review of SNE-based methods. Also, a comparative analysis of the considered methods is provided, which is done on important aspects such as algorithm implementation, relationship between methods, and performance. The aim of this paper is to investigate recent alternatives to SNE as well as to provide substantial results and discussion to compare them.

Keywords: Dimensionality reduction, divergences, similarity, stochastic neighbor embedding.

1 Introduction

For pattern recognition and data mining tasks involving high dimensional data sets, dimensionality reduction (DR) is a key tool. The aim of DR approaches is to extract lower dimensional, relevant information from high-dimensional input data, so that the performance of a pattern recognition system might be improved. As well, the data visualization will become more intelligible. Among the classical DR approaches, we may mention principal component analysis (PCA) and classical multidimensional scaling (CMDS), which are respectively based on variance and distance preservation criteria [1]. Nowadays, the focus of DR approaches relies on more developed criteria, which are aimed at preserving the data topology. In particular, the data topology is involved within the formulation through pairwise similarities between data points. Therefore, these approaches can be readily understood from a graph-theory point of view such that the data are represented by a non-directed and weighted graph, in which data points represent the nodes, and a non-negative similarity (also affinity) matrix holds the

[*] J.A. Lee is a Research Associate with the FRS-FNRS (Belgian National Scientific Research Fund). This work is funded by FRS-FNRS (Belgian National Scientific Research Fund) project 7.0175.13 DRedVis.

T. Villmann et al. (eds.), *Advances in Self-Organizing Maps and Learning Vector Quantization,* Advances in Intelligent Systems and Computing 295,
DOI: 10.1007/978-3-319-07695-9_6, © Springer International Publishing Switzerland 2014

pairwise edge weights. The pioneer methods incorporating similarities are Laplacian eigenmaps [2] and locally linear embedding [3], which are spectral approaches. More recently, given the fact that the rows of the normalized similarity matrix can be seen as probability distributions, methods based on divergences have emerged. Due to the probabilistic connotation, the most representative method is so named stochastic neighbour embedding (SNE) [4]. SNE and its variants have shown to be suitable for getting high-quality embedding data, since they preserve similarities in both low- and high-dimensional space during the optimization process. As alternatives to SNE, enhanced versions have been proposed. In [5, 6], a mixture of divergences is proposed. Additionally, an improved gradient to speed up the procedure is also introduced in [6]. Another approach, which consists of simplifying the SNE's formulation, is introduced in [7]. Such simpler version is founded on the same principle as elastic network [8] and it is solved by an approximate gradient following the direction of an underlying eigenvalue problem [9].

In this work, we present a short review of recent alternatives to SNE. A comparative analysis is done regarding some key aspects, namely: algorithm implementation, performance, and links between methods. For comparison purposes, we also evaluate a classic technique (CMDS), as well as a spectral approach (Laplacian eigenmaps – LE). Experiments are carried out over third conventional databases: an artificial spherical shell, the `COIL-20` image bank [10], and a subset of the `MNIST` image bank [11]. To quantify the performance of studied methods, an improved version of the average agreement rate is used, as described in [6]. Experimentally, we show the relationship between the divergence-based methods with the similarity preservation. The grounds and reasonings provided here may encourage new researches on any of the issues presented in this work, as well as the conclusions and discussions may facilitate users to select a method according to the compromise between complexity and performance.

The outline of this paper is as follows: Section 2 explains the studied methods and discusses in detail algorithm implementation issues and the links between methods. Experimental results and discussion are shown in Section 3. Finally, Section 4 draws the final remarks and conclusions.

2 Alternatives to Stochastic Neighbor Embedding

The DR problem is to embeded a high dimensional data matrix $\boldsymbol{Y} = [\boldsymbol{y}_i]_{1\leq i\leq N}$ into a low-dimensional, latent data matrix $\boldsymbol{X} = [\boldsymbol{x}_i]_{1\leq i\leq N}$, such that the relevant information is preserved. Denote $\boldsymbol{y}_i \in \mathbb{R}^D$ and $\boldsymbol{x}_i \in \mathbb{R}^d$ $(d < D)$ as the i-th data point from the high- and low-dimensional space. To cope with this problem, stochastic neighbor embedding (SNE) [4] minimizes the information divergence D between two distributions $\boldsymbol{P}_n = [p_{nm}]_{1\leq m\leq N}$ and $\boldsymbol{Q}_n = [q_{nm}]_{1\leq m\leq N}$ associated with the n-th point from observed and latent data, respectively. Then, using the Kullback-Leibler directed divergence D_{KL}, the SNE objective function is in the form:

$$E_{\mathrm{SNE}}(\boldsymbol{X}) = \sum_{n=1}^{N} \mathrm{D}_{\mathrm{KL}}(\boldsymbol{P}_n \| \boldsymbol{Q}_n) = \sum_{n,m=1}^{N} p_{nm} \log \frac{p_{nm}}{q_{nm}}. \tag{1}$$

Defining $\delta_{nm} = ||\boldsymbol{y}_n - \boldsymbol{y}_m||^2$ and $d_{nm} = ||\boldsymbol{x}_n - \boldsymbol{x}_m||^2$, distributions $\boldsymbol{P}_n$ and $\boldsymbol{Q}_n$ can be chosen as generalized, normalized nonsymmetric affinities in the form

$$p_{nm} = \frac{\exp\left(-\frac{1}{2}\delta_{nm}^2/\sigma_n^2\right)}{\sum\limits_{n\neq m'} \exp\left(-\frac{1}{2}\delta_{nm'}^2/\sigma_n^2\right)}, \quad \text{and} \quad q_{nm} = \frac{\exp\left(-\frac{1}{2}d_{nm}^2/\pi_n^2\right)}{\sum\limits_{n\neq m'} \exp\left(-\frac{1}{2}d_{nm'}^2/\pi_n^2\right)}, \tag{2}$$

with $q_{nn} = 0$ and $p_{nn} = 0$.

Symmetric SNE: A symmetric version of SNE (SSNE) can be achieved by selecting full normalized affinities which can readily be obtained by slightly expressions in (2). In this case, rather than a restricted sum, all entries must be summed on the denominator in order to enforce that all normalized entries sum to 1. This can be done by guaranteeing that $\mathbf{1}_N^\top \boldsymbol{Q} \mathbf{1}_N = \mathbf{1}_N^\top \boldsymbol{P} \mathbf{1}_N = 1$.

t-SNE: SNE-based methods suffer from reaching distorted and overlapped latent space, when d is smaller than the intrinsic dimension [7]. To cope with this issue, another variant raised, which is named t-SNE and consists of selecting the $\boldsymbol{Q}_n$ as a t-distributed sequence [5].

Jensen-Shanon embedding: In [12], it is proposed a mixture by adding a regularization parameter β to balance *precision* and *recall* so: $(1-\beta)\,\mathrm{D_{KL}}(\boldsymbol{P}_n||\boldsymbol{Q}_n) + \beta\,\mathrm{D_{KL}}(\boldsymbol{Q}_n||\boldsymbol{P}_n)$. Similarly, in [6], a novel approach is introduced which mixes the divergences as $\mathrm{D}_{\mathrm{KL}}^{\beta} = (1-\beta)\,\mathrm{D_{KL}}(\boldsymbol{P}_n||\boldsymbol{S}_n) + \beta\,\mathrm{D_{KL}}(\boldsymbol{Q}_n||\boldsymbol{S}_n)$, where $\boldsymbol{S}_n$ is a distribution following the same mixture rule so that $\boldsymbol{S}_n = (1-\beta)\boldsymbol{P}_n + \beta\boldsymbol{Q}_n$. This divergence is used in the so-called Jensen-Shannon embedding (JSE), which aims then to minimize $E_{\mathrm{JSE}} = \sum_{n=1}^{N} \mathrm{D}_{\mathrm{KL}}^{\beta}(\boldsymbol{Q}_n||\boldsymbol{S}_n)$ [6].

Elastic embedding Another alternative to SNE is introduced in [7], which is called elastic embedding (EE). EE is aimed to optimize:

$$E_{\mathrm{EE}}(\boldsymbol{X}|\lambda) = \sum_{n,m=1}^{N} w_{nm}^{+} d_{nm}^2 + \lambda \sum_{n,m=1}^{N} w_{nm}^{-} \exp(d_{nm}^2) = E_{\mathrm{EE}}^{+}(\boldsymbol{X}) + \lambda E_{\mathrm{EE}}^{-}(\boldsymbol{X}). \tag{3}$$

Briefly put, this method attempts to involve the two objectives that SNE fulfills but in a simpler way. To this end, which is the key of this method, two graphs are used. Then, we have two kind of weighting coefficients w_{nm}^{+} and w_{nm}^{-} being the entries of attractive $\boldsymbol{W}^{+}$ and repulsive $\boldsymbol{W}^{-}$ affinity matrices, respectively. Both of them are positive semi-definite matrices. For simplicity, full graphs affinities are considered: $w_{nm}^{-} = ||\boldsymbol{y}_n - \boldsymbol{y}_m||^2$ and $w_{nm}^{+} = \exp(-\frac{1}{2}\delta_n^2/\sigma^2)$. From Eq. (3), the gradient of E_{EE} can be written as:

$$\boldsymbol{G}(\boldsymbol{X}|\lambda) = 4\boldsymbol{X}(\boldsymbol{L}^{+} - \lambda\widetilde{\boldsymbol{L}}^{-}) = 4\boldsymbol{X}\boldsymbol{L}, \tag{4}$$

where $\widetilde{w}_{nm}^{-} = w_{nm}^{-}\exp(-d_{nm}^2)$, $w_{nm} = w_{nm}^{+} - \lambda\widetilde{w}_{nm}^{-}$, and their corresponding Laplacians $\widetilde{\boldsymbol{L}} = \widetilde{\boldsymbol{D}} - \widetilde{\boldsymbol{W}}$ and $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. Likewise, $\boldsymbol{L}^{+}$ is the non-normalized Laplacian and thus $\boldsymbol{L}^{+} = \boldsymbol{D}^{+} - \boldsymbol{W}^{+}$. In [7], to carry out the search for the suboptimal embedded solution $\boldsymbol{X}$, a gradient descent algorithm is used, which is powered via the spectral direction (SD) proposed in [9].

Following are discussed in detail some implementation issues in Section 2.1 as well as the links between methods in Section 2.2.

2.1 Implementation and Algorithms

In this section, we discuss about two recent implementations, here called: spectral direction and full gradient.

Implementation via spectral direction: Methods such as EE, SNE and SSNE can be implemented in a fast fashion via a SD-based gradient descent search [7]. We denote the n-th embedded data point at iteration r as $\boldsymbol{x}_n[r] = \boldsymbol{x}_n[r-1] + \alpha[r]\boldsymbol{\varrho}_n[r]$. SD is aimed at determining the optimal direction $\boldsymbol{\varrho}_n[r]$ by incorporating a partial-Hessian strategy within the gradient descent heuristic [9]. Then, by design, Hessian is heavily exploited which is advantageous for subsequent developments since it can be be computed fast and has the suitable property to be positive semi-definite. As an intuitive condition, sought direction must hold that $\boldsymbol{B}[r]\boldsymbol{\varrho}_n[r] = -\boldsymbol{g}_n$, being $\boldsymbol{g}_n$ the column n of $\boldsymbol{G}(\boldsymbol{X}|\lambda)$ and $\boldsymbol{B}[r]$ any positive semi-definite matrix. SD consists of calculating the gradient of $E_{\mathrm{EE}}(\boldsymbol{X}|\lambda)$ following the direction of an underlying convex function which arises when $\lambda = 0$. Such a function is in fact the attractive part $E^{+}_{EE}(\boldsymbol{X}) = E_{\mathrm{EE}}(\boldsymbol{X}|0)$, whose Hessian is $\nabla^2 E^{+}_{EE}(\boldsymbol{X}) = 4\boldsymbol{L}^{+}$ being evidently positive semi-definite. As a matter of fact, possible alternatives for selecting $\boldsymbol{B}[r]$ span from null perplexity to $k = N$ (full graph) which match respectively with degree $\boldsymbol{D}^{+}$ and Laplacian $\boldsymbol{L}^{+}$ [9].

Moreover, the calculation of step $\alpha[r]$ is powered by a backtracking line search [13] following the updating rule $\alpha_l[r] = \rho\alpha_{l-1}[r]$ for a user-provided constant ρ. Gathering the spectral directions in matrix $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{d\times N}$, per each iteration, output embedded data can be calculated as $\boldsymbol{X}^{*} = \boldsymbol{X} + \alpha_l[r]\boldsymbol{\mathcal{P}}$ under the convergence criterion given by $E_{\mathrm{EE}}(\boldsymbol{X} + \alpha_l[r]\boldsymbol{\mathcal{P}})|\lambda) > E_{\mathrm{EE}}(\boldsymbol{X}|\lambda) + c\alpha_l[r]\,\mathrm{tr}\,(\boldsymbol{\mathcal{P}}\boldsymbol{G}(\boldsymbol{X}|\lambda))$, where c is a small positive value. Steps for performing EE with backtracking line search are summarized in Algorithm 1. Within this framework, SNE and its variants can be alternatively implemented. To do so, the cost function of the method to be run should take place in $E(\boldsymbol{X})$. The gradient is the same for SNE-like methods, since the suboptimal solution is sough via a spectral direction.

Also, the calculation of SD is speeded up by using Cholesky decomposition. Namely, rather than calculating matrix directly with $\boldsymbol{\mathcal{P}} = -\boldsymbol{G}(\boldsymbol{X}|\lambda)(\boldsymbol{B})^{-1}$ (which is $\mathcal{O}(N^3D)$ when using conventional Gaussian-Jordan elimination), two solve triangular systems in the form $\boldsymbol{R}^{\top}\boldsymbol{R}\,\mathrm{vec}(\boldsymbol{P}) = -\,\mathrm{vec}(\boldsymbol{G})$ are solved, where $\boldsymbol{R}$ is the upper triangular matrix resulting from the Cholesky decomposition of $\boldsymbol{B} \otimes \boldsymbol{I}_d$. Latter calculation can be done in $\mathcal{O}(N^2 d)$ with standard linear algebra routines. In addition, computation of $\boldsymbol{R}$ needs to be done only once at first iteration and its complexity is $\mathcal{O}(\frac{1}{3}N)$.

Implementation via a full gradient and Hessian: In [6], the search is done by using a full gradient calculated over the whole cost function (no approximations are done). In this case, the search is done via $\boldsymbol{x}_n[r] = \boldsymbol{x}_n[r-1] + \mu_n[r]\nabla E$, where $\mu_n[r]$ is an adaptive step size dependent on the Hessian. Given the nature of divergences, doing so can increase the complexity. Even more when using a mixture of divergences ($E = E_{\mathrm{JSE}}$), calculation of gradient and Hessian may be more expensive. Nonetheless, the advantage of this implementation is that scaling is considered in both high and low dimensional space. This provides a more modulated gradient and then a better tracking of the local structure of data during the optimization process.

Algorithm 1. SNE via SD

```
Input: Affinity matrices $\boldsymbol{W}^+$ and $\boldsymbol{W}^-$, $N_{\text{iter}}$, $\epsilon$, $\lambda$, $\boldsymbol{X}$, $r = 1$
Compute the graph Laplacian $\boldsymbol{L}^+$
Compute the objective function $E(\boldsymbol{X})$ (3)
Set $\delta(\boldsymbol{X}^*, \boldsymbol{X}) \geq \epsilon$
while $\delta(\boldsymbol{X}^*, \boldsymbol{X}) \geq \epsilon$ do
    Calculate the gradient $\boldsymbol{G}(\boldsymbol{X}|\lambda)$ using Eq. (4)
    Calculate spectral direction matrix: $\boldsymbol{\mathcal{P}} = -\boldsymbol{G}(\boldsymbol{X}|\lambda)(\boldsymbol{L}^+)^{-1}$

    Backtracking line search for estimating $\boldsymbol{X}^*$
    Set $c$, $\rho$, and $\alpha_0$
    Initialize $l = 1$
    while $E(\boldsymbol{X} + \alpha\boldsymbol{\mathcal{P}}|\lambda) > E(\boldsymbol{X}|\lambda) + c\alpha_l \operatorname{tr}(\boldsymbol{\mathcal{P}}\boldsymbol{G}(\boldsymbol{X}|\lambda))$ do
        $\alpha_l = \rho\alpha_{l-1}$
        Calculate $E(\boldsymbol{X} + \alpha_l\boldsymbol{\mathcal{P}}|\lambda)$
        Increase $l$ by 1
    end while

    Estimate $\boldsymbol{X}^*$ as: $\boldsymbol{X}^* = \boldsymbol{X} + \alpha_l\boldsymbol{\mathcal{P}}$
    $\delta(\boldsymbol{X}^*, \boldsymbol{X}) = ||\boldsymbol{X}^* - \boldsymbol{X}||_F$
    Update $\boldsymbol{X} = \boldsymbol{X}^*$
end while
Output: Embedded data $\boldsymbol{X}$
```

2.2 Links between Methods

Relation between SNE and EE: Eliminating independent terms from $\boldsymbol{X}$, Equation (1) can be expanded as

$$E_{\text{SNE}}(\boldsymbol{X}) = \sum_{n,m=1}^{N} p_{nm}||\boldsymbol{x}_n - \boldsymbol{x}_m||^2 + \sum_{n=1}^{N} \log \sum_{n \neq m} \exp(||\boldsymbol{x}_n - \boldsymbol{x}_m||^2). \tag{5}$$

Hence we can appreciate that by omitting the log operator and adding a homotopy parameter λ, E_{SNE} becomes the EE's cost function. Furthermore, EE is a variant of the elastic network applied to solve the traveling salesman problem as explained in [8].

Relation between SNE and LE: Laplacian Eigenmaps (LE) introduced in [2] is a popular approach for DR. This approach is spectral and is aimed at minimizing local distances. The LE's cost function can be written as $\sum_{n,m=1}^{N} w_{nm}||\boldsymbol{x}_n - \boldsymbol{x}_m||$, where $\boldsymbol{W} = [w_{nm}]_{1 \leq n \leq N}$ is the similarity matrix and $|| \cdot ||$ stands for Euclidean distance. Alternatively, we can express LE's formulation as

$$E_{\text{LE}}(\boldsymbol{X}) = \operatorname{tr}(\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^\top) \text{ s. t. } \quad \boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^\top = \boldsymbol{I}_d, \quad \boldsymbol{X}\boldsymbol{D}\boldsymbol{1}_N = \boldsymbol{0}_d, \tag{6}$$

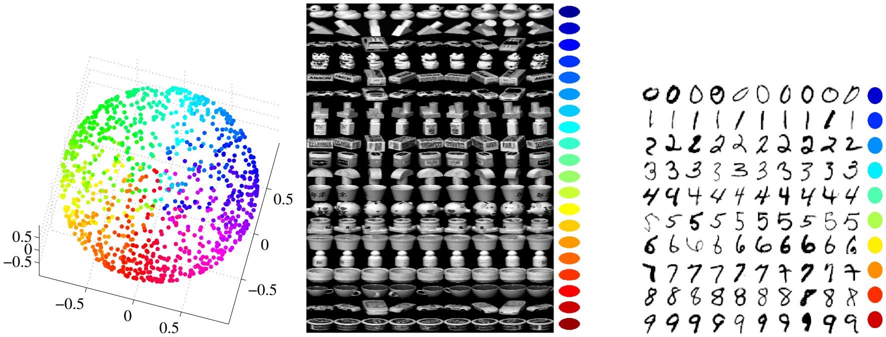
where $\boldsymbol{D} = \operatorname{Diag}(\boldsymbol{W}\boldsymbol{1}_N)$ is the degree matrix and $\boldsymbol{L}$ is the graph Laplacian matrix given by $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. LE's constraints facilitates the solution leading to a generalized eigenvalue problem. Along this line, the embedded data is then the d smallest vector eigenvectors of normalized Laplacian $\boldsymbol{D}^{-1/2}\boldsymbol{L}\boldsymbol{D}^{-1/2}$. This formulation is also useful to determine underline data clusters within input data [14]. Recalling Equation (5), it is noticeable that, doing as in diffusion maps [15] which means using the normalized affinities so that $p_{nm} = w_{nm}$, the right hand side of the Equation is the same as the LE objective function.

Relation between EE and LE: This relationship is quite similar to that when comparing SNE with EE. However, it is worth mentioning that by setting $\lambda = 0$, EE does not reach the same embedding as LE, since the optimization is different. EE's embedding is determined through a search and that of LE comes from a spectral decomposition under orthonormality assumptions.

3 Experiments and Results

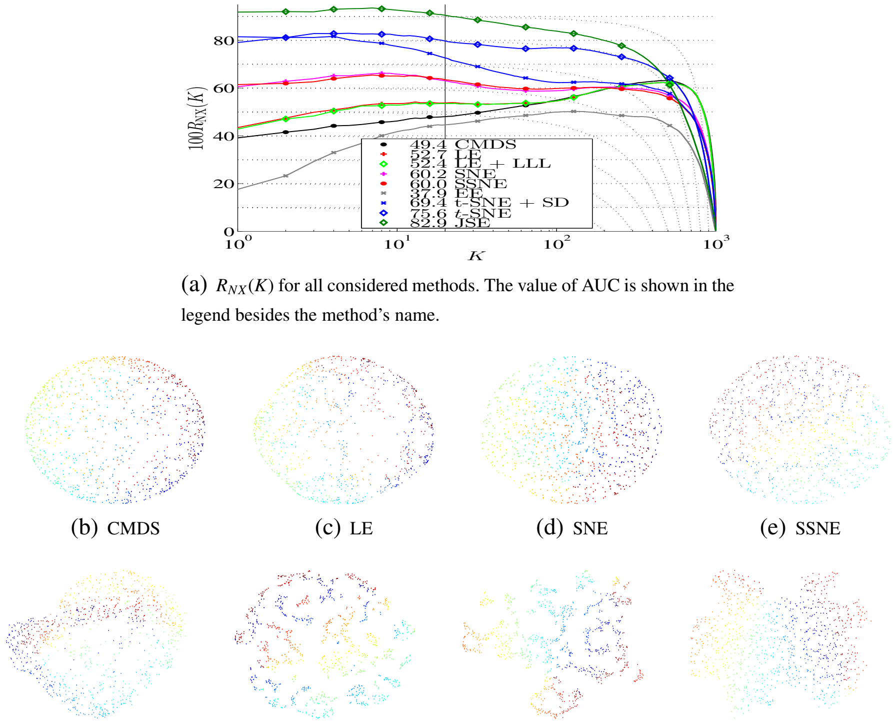
Following the experiments to compare the DR methods are described. First, the considered data sets and the methods to be compared are mentioned. Also, the parameter settings to carry out the DR procedure as well as the performance measure are described. Finally, obtained results and discussion are drawn.

Data sets and methods: Experiments are carried out over three conventional data sets. The first data set is an artificial spherical shell ($N = 1500$ data points and $D = 3$). The second data set is the `COIL-20` image bank [10], which contains 72 gray-level images representing 20 different objects ($N = 1440$ data points –20 objects in 72 poses/angles– with $D = 128^2$). The third data set is a randomly selected subset of the `MNIST` image bank [11], which is formed by 6000 gray-level images of each of the 10 digits ($N = 1500$ data points –150 instances for all 10 digits– and $D = 24^2$). Figure 1 depicts examples of the considered data sets.



(a) `Spherical shell` (b) `COIL-20` (c) `MNIST`

Fig. 1. The three considered data sets. To carry out the DR procedure, images from `COIL-20` and `MNIST` data sets are vectorized.



(a) $R_{NX}(K)$ for all considered methods. The value of AUC is shown in the legend besides the method's name.

(b) CMDS (c) LE (d) SNE (e) SSNE

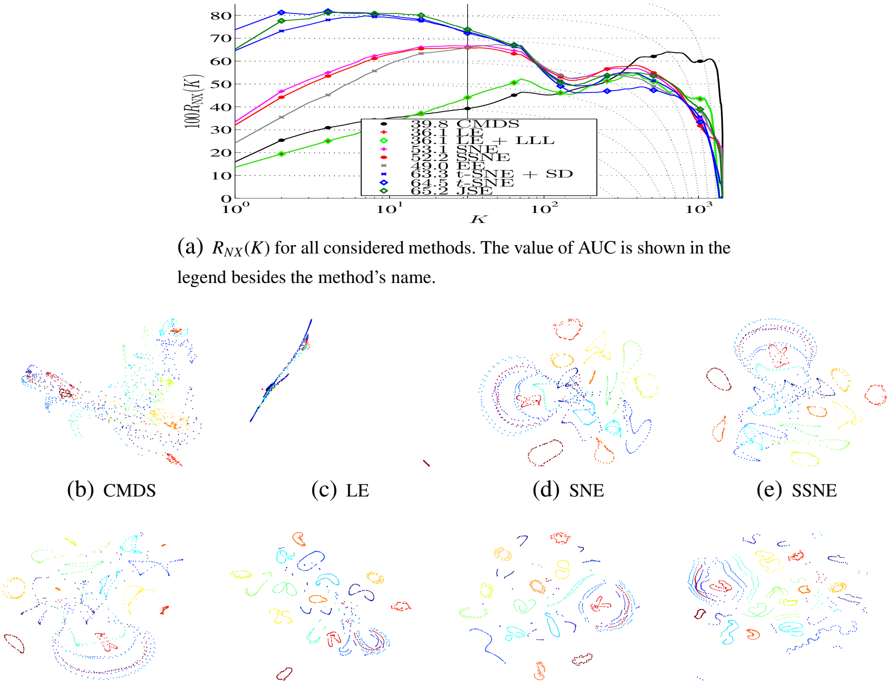
(f) EE (g) t-SNE + SD (h) t-SNE (i) JSE

Fig. 2. Results for `Spherical shell`. Results are shown regarding the quality measure $R_{NX}(K)$. The curves and their AUC (a) for all considered methods are depicted, as well as the embedding data (b)-(j).

Methods to be compared: We consider the SNE-like methods, namely: classical SNE, SSNE, t-SNE, EE, t-SNE via spectral direction (t-SNE + SD), and JSE. Also, we evaluate a representative classical technique, which is a CMDS; and a spectral technique being LE.

Performance measure and parameter settings: To quantify the performance of studied methods, the scaled version of the average agreement rate $R_{NX}(K)$ introduced in [6] is used, which is ranged within the interval $[0, 1]$. Since $R_{NX}(K)$ is calculated at each perplexity value from 2 to $N - 1$, a numerical indicator of the overall performance can be obtained by calculating its area under the curve (AUC). EE, t-SNE+SD and SSNE are implemented via a spectral direction procedure. Meanwhile, SNE, t-SNE and JSE are implemented via a full gradient scheme. Both SD and full gradient implementations involve a backtracking line search.

To form the similarity matrices, given a perplexity parameter K, the relative bandwidth parameter σ_n is estimated regarding its distribution $\boldsymbol{P}_n$ so that the entropy over neighbors of such distribution is approximately $\log K$. This is done by a binary search



(a) $R_{NX}(K)$ for all considered methods. The value of AUC is shown in the legend besides the method's name.

(b) CMDS (c) LE (d) SNE (e) SSNE

(f) EE (g) t-SNE + SD (h) t-SNE (i) JSE

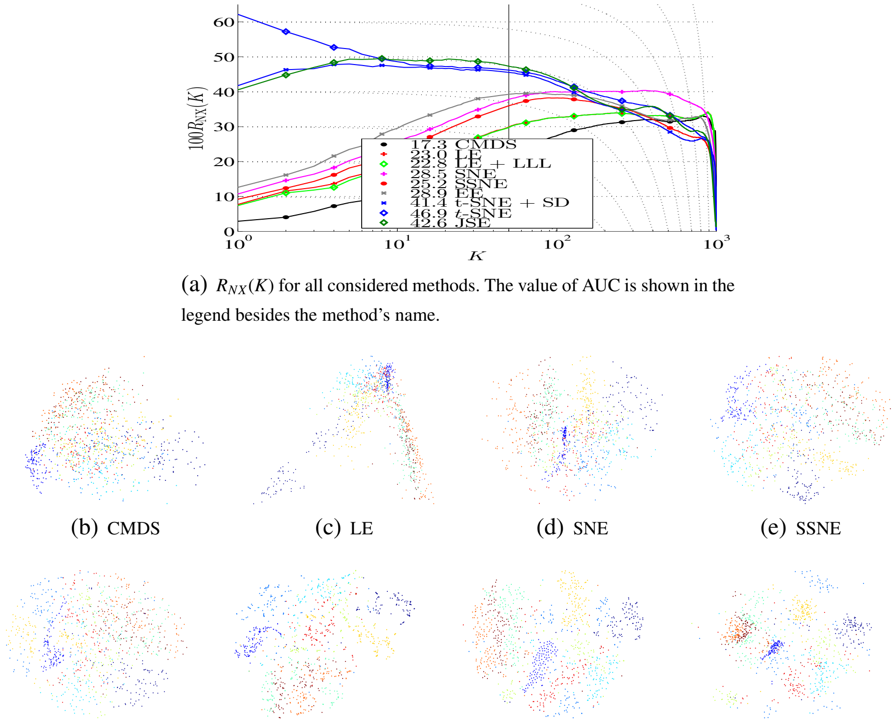
Fig. 3. Results and obtained embedding data for `COIL-20`

as explained in [7]. The homotopy parameter for EE is set $\lambda = 100$. Regularization parameter β for JSE is set to be $1/2$. For all methods, input data is embedded into a 2-dimensional space, then $d = 2$. The number of neighbors is established as $K = 30$. The rest of free parameter are $\epsilon = 10^{-3}$, $c = 0.1$, $\rho = 0.8$, and $\alpha_0 = 1$.

Results and discussion: Overall results for `Sphere`, `COIL` and `MNIST` regarding AUC $R_{NX}(K)$ are respectively shown in Figures 2, 3 and 4. As well, the resultant embedded spaces reached by each method are depicted.

For all considered databases, SNE-like methods perform a better embedding preserving smaller neighbours (local structure) in comparison the other methods. We can notice that SNE, SSNE and EE have a similar performance. In this case, SD makes that SNE and EE behave as a symmetrized version due to the strong assumption done over the gradient calculation. On the contrary, t-SNE + SD performs a better embedding since t-distributed probabilities may improve the separation of underline clusters despite that the gradient is biased to be that of the related, quadratic and symmetric form. Indeed, t-SNE + SD accomplishes a similar $R_{NX}(K)$ shape and AUC in comparison with t-SNE. JSE outperforms the remaining considered methods due to both the divergence type, and the identical similarity definition in the high-dimensional and low-dimensional space.

As another important observation from this work, we notice that the spectral methods (LE and CMDS), in general, attempt to preserve the global structure (larger neighbors).



(a) $R_{NX}(K)$ for all considered methods. The value of AUC is shown in the legend besides the method's name.

(b) CMDS (c) LE (d) SNE (e) SSNE

(f) EE (g) t-SNE + SD (h) t-SNE (i) JSE

Fig. 4. Results and obtained embedding data for `MNIST`

Particularly, CMDS exhibiting a pronounced peak on large neighbors. Then, we can claim that SNE based methods are better at preserving local structure, meanwhile those based on spectral analysis preserve the global structure.

4 Conclusions

This work reviews recent dimensionality reduction methods based on divergences. In particular, stochastic neighbor embedding and its improved variants. We provide a short comparative analysis involving key aspects such as relations between methods, algorithm implementation, and performance. Empirically, we demonstrate that methods using normalized similarities as probabilities and optimizing divergences reach better embedding by preserving the local structure of data. This is the case of SNE and its variants, in which the similarities are optimized in both high- and -low dimensional spaces. Meanwhile, spectral methods like multidimensional scaling and Laplacian eigenmaps are better at preserving global structure.

Discussion and results given here may facilitate users to choose a method seeking a good trade-off between performance and complexity.

References

1. Borg, I.: Modern multidimensional scaling: Theory and applications. Springer (2005)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15(6), 1373–1396 (2003)
3. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500), 2323–2326 (2000)
4. Hinton, G.E., Roweis, S.T.: Stochastic neighbor embedding. In: Advances in Neural Information Processing Systems, pp. 833–840 (2002)
5. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research 9(2579-2605), 85 (2008)
6. Lee, J.A., Renard, E., Bernard, G., Dupont, P., Verleysen, M.: Type 1 and 2 mixtures of kullback-leibler divergences as cost functions in dimensionality reduction based on similarity preservation. Neurocomputing (2013)
7. Carreira-Perpinán, M.A.: The elastic embedding algorithm for dimensionality reduction. In: ICML, vol. 10, pp. 167–174 (2010)
8. Durbin, R., Szeliski, R., Yuille, A.: An analysis of the elastic net approach to the traveling salesman problem. Neural Computation 1(3), 348–358 (1989)
9. Vladymyrov, M., Carreira-Perpiñán, M.Á.: Partial-hessian strategies for fast learning of nonlinear embeddings. CoRR, abs/1206.4646 (2012)
10. Nene, S.A., Nayar, S.K., Murase, H.: Columbia object image library (coil-20). Dept. Comput. Sci., Columbia Univ., New York, 62 (1996), http://www.cs.columbia.edu/CAVE/coil-20.html
11. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11), 2278–2324 (1998)
12. Venna, J., Peltonen, J., Nybo, K., Aidos, H., Kaski, S.: Information retrieval perspective to nonlinear dimensionality reduction for data visualization. The Journal of Machine Learning Research 11, 451–490 (2010)
13. Nocedal, J., Wright, S.: Numerical optimization. Series in operations research and financial engineering. Springer, New York (2006)
14. Yu, S.X., Shi, J.: Multiclass spectral clustering. In: Proceedings of the Ninth IEEE International Conference on Computer Vision, pp. 313–319. IEEE (2003)
15. Singer, A., Wu, H.-T.: Vector diffusion maps and the connection Laplacian. Communications on Pure and Applied Mathematics 65(8), 1067–1144 (2012)