Optimization Equivalence of Divergences Improves Neighbor Embedding

Zhirong Yang[2] ZHIRONG.YANG@AALTO.FI
Jaakko Peltonen[1,4] JAAKKO.PELTONEN@AALTO.FI
Samuel Kaski[1,3] SAMUEL.KASKI@AALTO.FI
[1]Helsinki Institute for Information Technology HIIT, [2]Department of Information and Computer Science, Aalto University, Finland, [3]Department of Computer Science, University of Helsinki, and [4]University of Tampere

Abstract

Visualization methods that arrange data objects in 2D or 3D layouts have followed two main schools, methods oriented for graph layout and methods oriented for vectorial embedding. We show the two previously separate approaches are tied by an optimization equivalence, making it possible to relate methods from the two approaches and to build new methods that take the best of both worlds. In detail, we prove a theorem of optimization equivalences between β- and γ-, as well as α- and Rényi-divergences through a connection scalar. Through the equivalences we represent several nonlinear dimensionality reduction and graph drawing methods in a generalized stochastic neighbor embedding setting, where information divergences are minimized between similarities in input and output spaces, and the optimal connection scalar provides a natural choice for the tradeoff between attractive and repulsive forces. We give two examples of developing new visualization methods through the equivalences: 1) We develop weighted symmetric stochastic neighbor embedding (ws-SNE) from Elastic Embedding and analyze its benefits, good performance for both vectorial and network data; in experiments ws-SNE has good performance across data sets of different types, whereas comparison methods fail for some of the data sets; 2) we develop a γ-divergence version of a PolyLog layout method; the new method is scale invariant in the output space and makes it possible to efficiently use large-scale smoothed neighborhoods.

Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32.

1. Introduction

We address two research problems: nonlinear dimensionality reduction (NLDR) of vectorial data and graph layout. In NLDR, given a set of data points represented with high-dimensional feature vectors or a distance matrix between such vectors, low-dimensional coordinates are sought for each data point. In graph layout, given a set of nodes (vertices) and a set of edges between node pairs, the task is to place the nodes on a 2D or 3D display. Solutions to both research problems are widely used in data visualization.

The two genres have yielded corresponding schools of methods. To visualize vectorial data, many NLDR methods have been introduced, from linear methods based on eigendecomposition, such as Principal Component Analysis, to nonlinear methods such as Isomap or Locally Linear Embedding. Recent well-performing methods include Stochastic Neighbor Embedding (SNE; Hinton & Roweis, 2002), Neighbor Retrieval Visualizer (Venna et al., 2010), Elastic Embedding (EE; Carreira-Perpiñán, 2010), Semidefinite Embedding (Weinberger & Saul, 2006), and the Gaussian process latent variable model (Lawrence, 2003). However, these methods often yield poor embeddings given network data as input, especially when the graph nodes have heavily imbalanced degrees.

Force-based methods are probably the most used graph layout method family. They set attractive forces between adjacent graph nodes and repulsive forces between all nodes, and seek an equilibrium of the force system analogous to having springs attached between nodes. The methods typically modify an initial vertex placement by iteratively adjusting vertices. Many layout methods have been proposed, such as sfdp (Hu, 2005), LinLog (Noack, 2007), OpenOrd (Martin et al., 2011), and MaxEnt (Gansner et al., 2013). The methods can be used for vector-valued data as well, transformed into a neighborhood graph, but have not been designed for that task and often do not find good low-dimensional embeddings for high-dimensional neighborhoods.

Several NLDR and graph drawing methods can be ex-

pressed as optimizing a divergence between neighborhoods in the input and output spaces, collectively called Neighbor Embedding (NE; Yang et al., 2013). Two common kinds of NE objectives are 1) using a separable divergence[1] on non-normalized neighborhoods, as in e.g. EE, and 2) using a nonseparable divergence on normalized neighborhoods, as in e.g. SNE. However, it remains unknown whether the two kinds of objectives are essentially equivalent.

In this paper we address the question by introducing novel relationships between the objectives. We prove an *optimization equivalence* between a separable divergence (α or β) and its corresponding nonseparable divergence (Rényi or γ) through an optimizable connection scalar. This theorem provides a connection between common NLDR and conventional force-directed methods, allowing development of more general and robust visualization methods by using extensions and insights from either side. Separable force-directed objectives are easier to design and optimize, but the tradeoff between attraction and repulsion as a hyperparameter is hard to determine. On the other hand, objectives formulated with Rényi- or γ-divergence are scale invariant. Moreover, the optimal connection scalar yields a principled choice for the attraction-repulsion tradeoff.

We demonstrate two applications of the optimization equivalence. First, we introduce a weighted variant of symmetric SNE (ws-SNE) by integrating the "edge-repulsion" strategy from the force-directed graph layout algorithms and applying the optimization equivalence which automatically selects the best tradeoff between attraction and repulsion. Experiments show that *ws-SNE works well for both vectorial and network data*, whereas the other compared neighbor embedding or graph drawing methods achieve good results for only one of the two types of data. The superior performance of ws-SNE is explained through the optimization equivalence. Second, we develop a new variant of the PolyLog method that minimizes the γ-divergence. This new method is invariant to scale of the mapped points and allows large-scale smoothed input neighborhoods.

We review popular divergence measure families in Section 2. We introduce the optimization equivalence theorem in Section 3 and the neighbor embedding framework in Section 4. We introduce the new visualization methods and their analysis in Section 5, and show their goodness by experimental comparisons in Section 6. Section 7 concludes.

2. Divergence measures

Information divergences, denoted by $D(p||q)$, were originally defined for probabilities and later extended to measure difference between two (usually nonnegative) tensors p and q, where $D(p||q) \geq 0$ and $D(p||q) = 0$ iff $p = q$. To avoid notational clutter we only give vectorial definitions; it is straightforward to extend the formulae to matrices and higher-order tensors. We focus on four important families of divergences: α-, β-, γ- and Rényi (parameterized by r). Their definitions are (see e.g. Cichocki et al., 2011):

$$D_\alpha(p||q) = \frac{1}{\alpha(\alpha-1)} \sum_i \left[p_i^\alpha q_i^{1-\alpha} - \alpha p_i + (\alpha-1) q_i\right],$$

$$D_\beta(p||q) = \frac{1}{\beta(\beta-1)} \sum_i \left[p_i^\beta + (\beta-1) q_i^\beta - \beta p_i q_i^{\beta-1}\right],$$

$$D_\gamma(p||q) = \frac{\ln\left(\sum_i p_i^\gamma\right)}{\gamma(\gamma-1)} + \frac{\ln\left(\sum_i q_i^\gamma\right)}{\gamma} - \frac{\ln\left(\sum_i p_i q_i^{\gamma-1}\right)}{\gamma-1},$$

$$D_r(p||q) = \frac{1}{r-1} \ln\left(\tilde{p}_i^r \tilde{q}_i^{1-r}\right),$$

where p_i and q_i are the entries in p and q respectively, $\tilde{p}_i = p_i / \sum_j p_j$, and $\tilde{q}_i = q_i / \sum_j q_j$. To handle p's containing zero entries, we only consider nonnegative α, β, γ and r. These families are rich as they cover most commonly used divergences in machine learning such as

$$D_{\mathrm{KL}}(p||q) = \sum_i \tilde{p}_i \ln \frac{\tilde{p}_i}{\tilde{q}_i},$$

$$D_{\mathrm{I}}(p||q) = \sum_i \left(p_i \ln \frac{p_i}{q_i} - p_i + q_i\right),$$

$$D_{\mathrm{IS}}(p||q) = \sum_i \left(\frac{p_i}{q_i} - \ln \frac{p_i}{q_i} - 1\right),$$

where D_{KL} (obtained from D_r with $r \to 1$ or D_γ with $\gamma \to 1$), D_{I} ($\alpha \to 1$ or $\beta \to 1$), and D_{IS} ($\beta \to 0$) denote normalized Kullback-Leibler (KL) divergence, non-normalized KL-divergence and Itakura-Saito divergence, respectively. Other named special cases include Euclidean distance ($\beta = 2$), Hellinger distance ($\alpha = 0.5$), and Chi-square divergence ($\alpha = 2$). Different divergences have become widespread in different domains. For example, D_{KL} is widely used for text documents (e.g. Hofmann, 1999) and D_{IS} is popular for audio signals (e.g. Févotte et al., 2009). In general, estimation using α-divergence is more exclusive with larger α's, and more inclusive with smaller α's (e.g. Minka, 2005). For β-divergence, the estimation becomes more robust but less efficient with larger β's.[2]

3. Connections between divergence measures

Here we present the main theoretical result. Previous work on divergence measures has mainly focused on relationships within one parametrized family. Little research has been done on the inter-family relationships; it is only known that there are correspondences between D_α and D_r, as well as between D_β and D_γ (see e.g. Cichocki et al.,

[1] A separable divergence here means a divergence that is a sum of pairwise terms, where each term depends only on locations of one pair of data.

[2] A robust estimator is insensitive to small departures from the idealized assumptions. An efficient estimator is the minimum variance unbiased estimator. See e.g. (Cichocki et al., 2011).

2009). We make the more general connection precise by a new theorem of *optimization equivalence*:

Theorem 1. *For $p_i \geq 0$, $q_i \geq 0$, $i = 1, \ldots, M$, and $\tau \geq 0$,*

$$\arg\min_q D_{\gamma\to\tau}(p||q) = \arg\min_q \left[\min_{c\geq 0} D_{\beta\to\tau}(p||cq)\right]. \quad (1)$$

$$\arg\min_q D_{r\to\tau}(p||q) = \arg\min_q \left[\min_{c\geq 0} D_{\alpha\to\tau}(p||cq)\right]. \quad (2)$$

The proof is done by zeroing the derivative of the right hand side with respect to c (details in the appendix). The optimal c is given in closed form:

$$c^* = \arg\min_c D_{\beta\to\tau}(p||cq) = \frac{\sum_i p_i q_i^{\tau-1}}{\sum_i q_i^\tau},$$

$$c^* = \arg\min_c D_{\alpha\to\tau}(p||cq) = \left(\frac{\sum_i p_i^\tau q_i^{1-\tau}}{\sum_i q_i}\right)^{\frac{1}{\tau}},$$

with the special case $c^* = \exp\left(-\frac{\sum_i q_i \ln(q_i/p_i)}{\sum_i q_i}\right)$ for $\alpha \to 0$. Obviously $c^* \geq 0$ as p and q are nonnegative.

The optimization equivalence not only holds for the global minima but also all local minima. This is justified by the following proposition (proof at the end of the Appendix):

Proposition 2. *The stationary points of $D_{\gamma\to\tau}(p||q)$ and $\min_{c\geq 0} D_{\beta\to\tau}(p||cq)$, as well as of $D_{r\to\tau}(p||q)$ and $\min_{c\geq 0} D_{\alpha\to\tau}(p||cq)$, in Theorem 1 are the same.*

To understand the value of the above theorem, let us first look at pros and cons of the four divergence families. The γ- and Rényi divergences are invariant of the scaling of p and q, which is desirable in many applications. However, they are not separable over the entries which increases optimization difficulty, and it is more difficult to design variants of the divergences due to the complicated functional forms. On the other hand, α- and β-divergences are separable and yield simpler derivatives, thus stochastic or distributed implementations become straightforward. However, the separable divergences are sensitive to the scaling of p and q.

Theorem 1 lets us take the advantages from either side. To design a divergence as a cost function for an application one can start from the separable side, inserting optimization and weighting strategies as needed, for example based on analyzing the resulting gradients, and then formulate the final scale-invariant objective by a γ- or Rényi-divergence and analyze its properties (see an example in Section 5.1). In optimization, one can turn back to the separable side and use efficient algorithms. In visualization, we show below that the scalar c (denoted λ in Section 4) controls the tradeoff between two learning sub-objectives corresponding to attractive and repulsive forces. Unlike in conventional force-directed approaches, here Theorem 1 gives an optimization principle to adaptively and automatically choose the best tradeoff: in order to be equivalent to a scale-invariant objective, the objective is formulated as on the right-hand side of (1) or (2), and the optimal tradeoff c^* is found as part of the minimization.

4. Neighbor Embedding optimizes divergences

We present a framework for visualization based on information divergences. We start from multivariate data and generalize to graph visualization in the next section. Suppose there are N high-dimensional data objects $\{x_1, \ldots, x_N\}$. Their neighborhoods are encoded in a square nonnegative matrix P, where P_{ij} is proportional to the probability that x_j is a neighbor of x_i. Neighbor Embedding (NE) finds a low-dimensional mapping $x_i \mapsto y_i \in \mathbb{R}^m$ such that the neighborhoods are approximately preserved in the mapped space. Usually $m = 2$ or 3. If the neighborhood in the mapped space is encoded in $Q \in \mathbb{R}^{n\times n}$ where Q_{ij} is proportional to the probability that y_j is a neighbor of y_i, the NE task is to minimize $D(P||Q)$ over $Y = [y_1, y_2, \ldots, y_N]^T$ for a certain divergence D.

The formulation originated from Stochastic Neighbor Embedding (SNE; Hinton & Roweis, 2002). Let $p_{ij} \geq 0$ and $q_{ij} \stackrel{\text{def}}{=} q\left(\|y_i - y_j\|^2\right) > 0$. SNE minimizes $\sum_i D_{\text{KL}}(P_{i:}||Q_{i:})$ where $P_{ij} = \frac{p_{ij}}{\sum_k p_{ik}}$ and $Q_{ij} = \frac{q_{ij}}{\sum_k q_{ik}}$. Typically q_{ij} is proportional to the Gaussian distribution so that $q_{ij} = \exp\left(-\|y_i - y_j\|^2\right)$, or proportional to the Cauchy distribution so that $q_{ij} = (1 + \|y_i - y_j\|^2)^{-1}$ (i.e. the Student t-distribution with a single degree of freedom).

Different choices of P, Q, and/or D give different NE methods. For example, minimizing a convex combination of $D_{r\to 1}$ and $D_{r\to 0}$, that is, $\sum_i \kappa D_{\text{KL}}(P_{i:}||Q_{i:}) + (1-\kappa) D_{\text{KL}}(Q_{i:}||P_{i:})$ with $\kappa \in [0, 1]$ a tradeoff parameter, results in the method NeRV (Venna et al., 2010) which has an information retrieval interpretation of making a tradeoff between precision and recall; SNE is a special case ($\kappa = 1$) maximizing recall, whereas $\kappa = 0$ maximizes precision. If the normalization is matrix-wise: $P_{ij} = p_{ij}/\sum_{kl} p_{kl}$ and $Q_{ij} = q_{ij}/\sum_{kl} q_{kl}$, minimizing $D_{\text{KL}}(P||Q)$ over Y gives a method called Symmetric SNE (s-SNE; van der Maaten & Hinton, 2008). When $q_{ij} = (1 + \|y_i - y_j\|^2)^{-1}$, it is also called t-SNE (van der Maaten & Hinton, 2008).

Although NE methods can be interpreted as force-directed methods, their design is different from conventional ones: in NE a divergence is picked, and forces between data arise from minimizing it. We distinguish SNE from conventional force-directed layouts because, by Theorem 1, SNE provides an information-theoretic way to automatically select the best tradeoff between attraction and repulsion.

Next we show that two other existing visualization methods can also be unified in the framework.

Proposition 3. *Elastic Embedding (EE) is a separable-divergence variant of s-SNE; s-SNE is a non-separable divergence variant of EE.* **Proof:** Carreira-Perpiñán (2010)

proposed EE which minimizes a Laplacian Eigenmap term (Belkin & Niyogi, 2002) plus a repulsive term: $\mathcal{J}_{\text{EE}}(Y) = \sum_{ij} p_{ij}\|y_i - y_j\|^2 + \lambda \sum_{ij} \exp\left(-\|y_i - y_j\|^2\right)$, where λ controls the tradeoff between attraction and repulsion[3]. The EE objective can be rewritten as $\mathcal{J}_{\text{EE}}(Y) = D_{\text{I}}(p||\lambda q) + C(\lambda)$, where $q_{ij} = \exp\left(-\|y_i - y_j\|^2\right)$, and $C(\lambda) = \left(\sum_{ij} p_{ij}\right) \ln \lambda - \sum_{ij} \left[p_{ij} \ln p_{ij} - p_{ij}\right]$ is constant with respect to Y. Minimizing $\mathcal{J}_{\text{EE}}$ over Y is thus equivalent to minimizing the separable divergence $D_{\text{I}}(p||\lambda q)$ over Y. Notice that we do not optimize $\mathcal{J}_{\text{EE}}$ over λ. This information divergence formulation also provides an automatic way to choose λ by using Theorem 1: $\arg\min_Y \left[\min \lambda D_{\text{I}}(p||\lambda q)\right] = \arg\min_Y D_{\text{KL}}(p||q)$, with the best $\lambda = \sum_{ij} p_{ij} / \sum_{ij} q_{ij}$. The non-separable divergence on the right-hand side is the s-SNE objective. Thus EE with the best tradeoff (yielding minimum I-divergence) is essentially s-SNE with Gaussian embedding kernels.

Proposition 4. *Node-repulsive LinLog is a divergence minimization method.* **Proof:** Noack (2007) proposed the LinLog energy model which is widely used in graph drawing. The node repulsive version of LinLog minimizes $\mathcal{J}^{\lambda}_{\text{LinLog}}(Y) = \lambda \sum_{ij} p_{ij}\|y_i - y_j\| - \sum_{ij} \ln \|y_i - y_j\|$. Algebraic manipulation yields $\mathcal{J}^{\lambda}_{\text{LinLog}}(Y) = D_{\text{IS}}(p||\lambda q) +$ constant, where $q_{ij} = \|y_i - y_j\|^{-1}$.

A recent work (Bunte et al., 2012) also considers SNE for dimension reduction and visualization with various information divergences, but their formulation is restricted to stochastic matrices, and gives no optimization equivalence between normalized and non-normalized cases.

5. Developing new visualization methods

The framework and the optimization equivalence between divergences not only relate existing approaches, but also enable us to develop new visualization methods. We give two examples of such development: 1) a generalized SNE that works for both vectorial and network data, and 2) a γ-divergence formulation of PolyLog (Noack, 2007) which is scale invariant in the output space and allows large-scale smoothed input neighborhoods. In both examples, Theorem 1 plays a crucial role in the development.

5.1. Example 1: ws-SNE

We want to build a method for a vectorial embedding but incorporating the useful edge repulsion (ER) strategy from graph drawing. ER would be easy to add to the EE objective as it is pairwise separable. Borrowing the ER strategy from Noack (2007), we insert weights M in the repulsive term: $\mathcal{J}_{\text{weighted-EE}}(Y) = \sum_{ij} p_{ij}\|y_i - y_j\|^2 + \lambda \sum_{ij} M_{ij} \exp\left(-\|y_i - y_j\|^2\right)$, where $M_{ij} = d_i d_j$, and the vector d measures importance of the nodes. We use degree centrality as the measurement, i.e., d_i=degree of the i-th node. $\mathcal{J}_{\text{weighted-EE}}(Y)$ has downsides: it needs a user-set edge repulsion weight λ, and is not invariant to scaling of p. By Theorem 1 we create a corresponding improved method, ws-SNE, minimizing a nonseparable divergence.

Proposition 5. *Weighted EE is a separable divergence minimizing method and its non-separable variant is ws-SNE.* **Proof:** Writing $q_{ij} = \exp(-\|y_i - y_j\|^2)$ with $q_{ii} = 0$, we have $\mathcal{J}_{\text{weighted-EE}}(Y) = D_{\text{I}}(p||\lambda M \circ q) + F(\lambda)$, where $\circ$ denotes element-wise product and $F(\lambda) = C(\lambda) + \sum_{ij} p_{ij} \ln d_i d_j$ is a constant to Y. In the final and most important step, by a special case of Theorem 1: $\arg\min_Y D_{\text{KL}}(p||M \circ q) = \arg\min_Y \left[\min_{\lambda \geq 0} D_{\text{I}}(p||\lambda M \circ q)\right]$, we obtain a new variant of SNE which minimizes $D_{\text{KL}}(p||M \circ q)$ over Y:

$$\mathcal{J}_{\text{ws-SNE}}(Y) = -\sum_{ij} p_{ij} \ln q_{ij} + \ln \sum_{ij} M_{ij} q_{ij} + \text{constant}$$

We call the new method weighted symmetric Stochastic Neighbor Embedding (ws-SNE). As in SNE, other choices of q can be used; for example, the Gaussian q can be replaced by the Cauchy $q_{ij} = (1 + \|y_i - y_j\|^2)^{-1}$ or other heavy-tailed functions (see e.g. Yang et al., 2009).

We used degree centrality as the importance measure for simplicity. Other centrality measures include closeness, betweenness, and eigenvector centralities. Note that for vectorial data with K-Nearest-Neighbor (KNN) graphs, ws-SNE reduces to s-SNE if out-degree centrality is used.

The ws-SNE method is related to the multiple maps t-SNE method (van der Maaten & Hinton, 2011), but differently the latter aims at visualizing multiple views of a vectorial dataset by several sets of variable node weights. In addition, ws-SNE does not impose the stochasticity constraint on the node weights. It is unknown whether multiple maps t-SNE can handle imbalanced degrees in graph drawing.

Analysis: best of both worlds. As shown by Proposition 5, ws-SNE combines edge-repulsion merits from graph drawing with the scale-invariance and optimal attraction-repulsion tradeoff from SNE. We find it performs consistently well for vectorial and network data visualization.

SNE and its variants were originally designed for nonlinear dimensionality reduction, where all data points are generally assumed equally important; this assumption works well for neighborhoods of vectorial data, e.g. p coming from a symmetrized KNN graph, where degrees do not differ much. However, for graph or network data where the degree distribution can be highly imbalanced, SNE often yields poor visualizations where high degree nodes

[3] For notational clarity we only illustrate EE with $w^-_{mn} = 1$ (see Carreira-Perpiñán, 2010, Eq. 6). In his experiments Carreira-Perpiñán (2010) also used uniform w^-. It is straightforward to extend the connection to the weighted version.

are placed in the center (see Figure 1, C3 to C6 for examples). In contrast, ws-SNE uses edge repulsion to handle such cases of imbalanced degrees. When the d_i are not uniform, ws-SNE behaves differently from conventional s-SNE, which can be explained by its gradient $\frac{\partial \mathcal{J}_{\text{ws-SNE}}(Y)}{\partial y_i} = \sum_j p_{ij} q_{ij}^{\theta}(y_i - y_j) - c d_i d_j q_{ij}^{1+\theta}(y_i - y_j)$, where $c = \sum_{ij} p_{ij} / (\sum_{ij} d_i d_j q_{ij})$ is the connection scalar, and $\theta = 0$ for Gaussian q and $\theta = 1$ for Cauchy q. The first term in the summation is for attraction of nodes and the second for repulsion. Compared with the s-SNE gradient, the repulsion part is weighted by $d_i d_j$ in ws-SNE. That is, important nodes have extra repulsive force with the others and thus tend to be placed farther. This edge-repulsion strategy has been shown to be effective in graph drawing to overcome the "crowding problem", namely, many mapped points becoming crowded in the center of the display.

On the other hand, conventional graph drawing methods do not work well for multivariate data. They are usually designed by choosing various attractive and repulsive force terms, as in the "spring-electric" layout (Eades, 1984), Fruchterman-Rheingold (Fruchterman & Reingold, 1991), LinLog (Noack, 2007), and ForceAtlas2 (Jacomy et al., 2011). These force-directed models have a serious drawback: the layout is sensitive to scaling of the input graph, thus the user must carefully (and often manually) select the tradeoff between attraction and repulsion. A constant tradeoff or annealing scheme may yield mediocre visualizations.

The optimal tradeoff is automatically selected in ws-SNE by the optimization equivalence, yielding an objective invariant to the scale of p. This is more convenient and often performs better than conventional force-based methods.

Information retrieval interpretation. Venna & Kaski (2007) and Venna et al. (2010) gave information retrieval perspectives of the unweighted SNE. As a new contribution, we show in the supplement that ws-SNE optimizes visualizations for a two-stage retrieval task: retrieving initial points and then their neighbors. In brief, we prove the cost can be written as a sum of divergences as $\mathcal{J}_{\text{ws-SNE}}(Y) = D_{\text{KL}}(\{\tilde{p}_i\}||\{\tilde{q}_i\}) + \sum_i \tilde{p}_i D_{\text{KL}}(\{\tilde{p}_{j|i}\}||\{\tilde{q}_{j|i}\})$ where $\tilde{p}_i = \sum_k p_{ik} / \sum_{lm} p_{lm}$ and $\tilde{q}_i = (d_i \sum_k d_k q_{ik}) / (\sum_{kl} d_k d_l q_{kl})$ are marginal probabilities in the input and output space, and $\tilde{p}_{j|i} = p_{ij} / \sum_k p_{ik}$ and $\tilde{q}_{j|i} = d_j q_{ij} / (\sum_k d_k q_{ik})$ are conditional probabilities in the input and output space around point i. The divergence between marginal probabilities is interpreted as performance of retrieving initial points; divergences between conditional probabilities are interpreted as performance of retrieving neighbors. In both stages the retrieval performance is mainly measured by recall (cost of missing points and their neighbors), and the weighting causes optimization to distribute high- and low-importance nodes more evenly.

5.2. Example 2: γ-QuadLog

In graph layout, smoothing graph adjacencies by random walks is potentially beneficial but computationally unfeasible for many methods as smoothed adjacencies can be non-sparse. We use Theorem 1 to develop a layout method that efficiently incorporates random-walk smoothing.

A sparse input graph A can be smoothed by computing a random-walk transition probability matrix $p = (1-\rho)\left(I - \rho D^{-1/2} A D^{-1/2}\right)^{-1}$ with $\rho \in (0,1)$ and $D_{ii} = \sum_j A_{ij}$. Smoothing can help layout methods avoid poor local optima and reveal macro structure of data. However, the matrix p is dense and infeasible to use explicitly in computing layout cost functions for large graphs. We start from QuadLog, a force-based method in the r-PolyLog family (Noack, 2007): $\mathcal{J}^{\lambda}_{\text{QuadLog}}(Y) = \lambda \sum_{ij} p_{ij} \|y_i - y_j\|^2 - \sum_{ij} \ln \|y_i - y_j\|^2$. While random walk smoothing can be applied in QuadLog, it is not scale-invariant and needs a user-set tradeoff parameter λ; we now solve this.

Proposition 6. *QuadLog minimizes a separable divergence, and there exists an equivalent minimization of a non-separable divergence that still permits fast use of random walk smoothing.* **Proof:** We show the QuadLog objective is an Itakura-Saito divergence plus constants with respect to Y: simple manipulation gives $\mathcal{J}^{\lambda}_{\text{QuadLog}}(Y) = D_{\text{IS}}(p||\lambda q) + \sum_{ij} \ln(p_{ij}/\lambda) + N(N-1)$, where $q_{ij} = \|y_i - y_j\|^{-2}$. By Theorem 1, minimizing $\mathcal{J}^{\lambda}_{\text{QuadLog}}(Y)$ with respect to λ is equivalent to minimizing $D_{\gamma \to 0}(p||q)$. We call the new method γ-QuadLog as it is a counter-part of QuadLog in the γ-divergence family. Dropping the additive constants, the γ-QuadLog objective is

$$\mathcal{J}_{\gamma\text{-QuadLog}}(Y) = \ln \sum_{ij} p_{ij} \|y_i - y_j\|^2 - \frac{\sum_{ij} \ln \|y_i - y_j\|^2}{N(N-1)}.$$

The new objective has two advantages: 1) It is scale-invariant in input and output: multiplying p by a scaling factor does not change the optima; multiplying Y by a scaling factor does not even change the objective value. 2) It allows use of smoothed neighborhood graphs: the γ-QuadLog objective can be computed using the matrix product pY, and as in QuadLog, pY can be scalably computed by iterative approaches (e.g. Zhou et al., 2003, if random walk is used).

γ-QuadLog is the only method we know with both advantages 1) and 2). We could e.g. develop a scale-invariant version of node-repulsive LinLog (take the divergence form of $\mathcal{J}^{\lambda}_{\text{LinLog}}(Y)$ from Proposition 4: by Theorem 1, optimizing it w.r.t. λ is equivalent to $\min_Y D_{\gamma \to 0}(p||q)$) but it would not allow efficient use of smoothed graphs.

6. Experiments

We compare the ws-SNE method with EE, t-SNE and two widely used graph drawing programs graphviz and Lin-

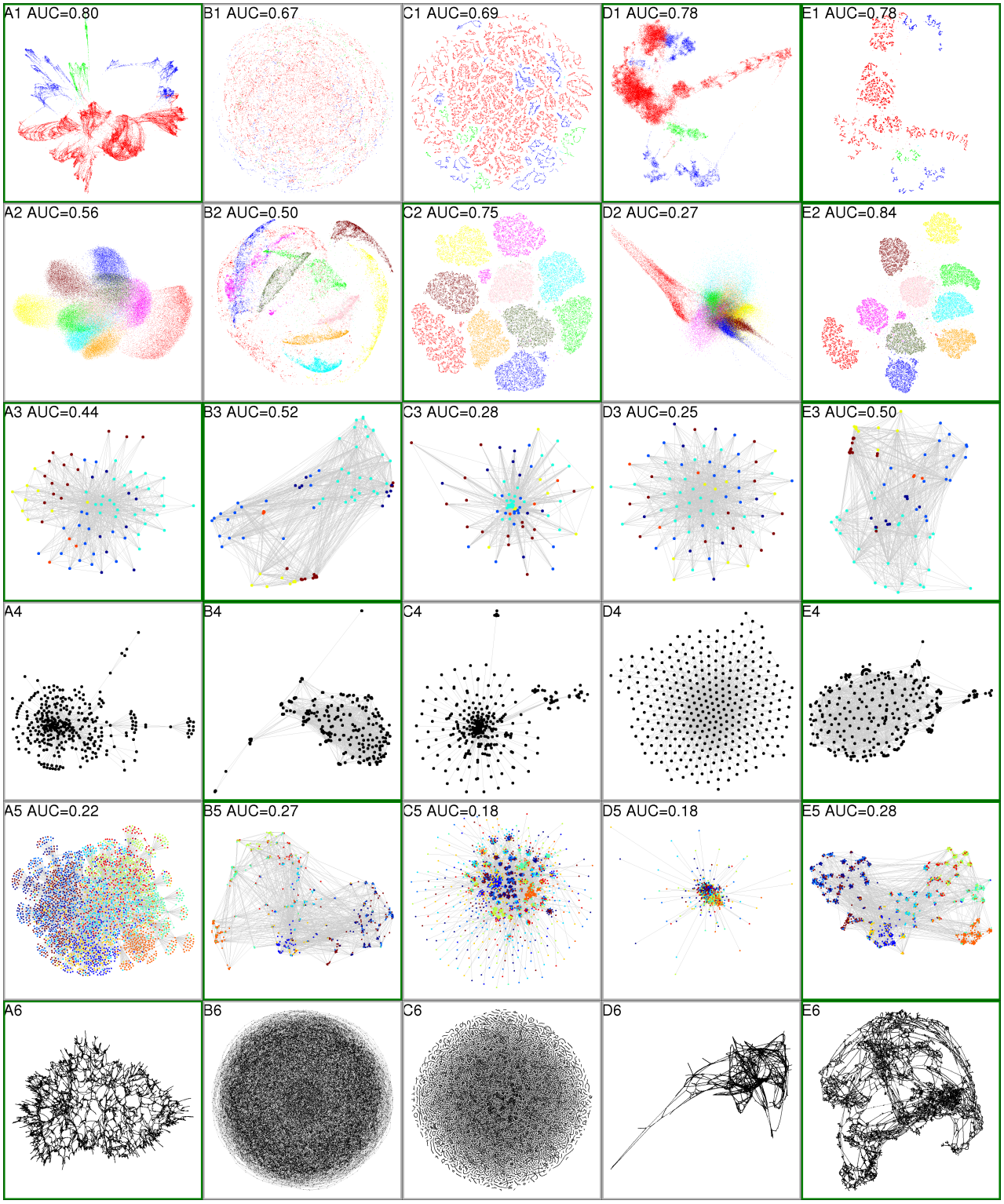

Figure 1. Visualizations of (rows 1-6) `shuttle`, `MNIST`, `worldtrade`, `usair97`, `mirex07`, and `luxembourg` using (columns A-E) graphviz, LinLog, t-SNE, EE, and ws-SNE. We manually label acceptable (see text) visualizations by a green border. Only ws-SNE yields acceptable results on all data. For data with ground truth classes, "AUC" denotes area under the retrieval precision-recall curve in KNN neighborhoods (curves in Figure 2). ws-SNE is the best or second best in all cases, yielding the most consistent good performance. Displays D1, E1, and D2 are zoomed to the densest area; full displays are in the supplemental document. The figure is best viewed in color.

Log. For EE, ws-SNE, graphviz and LinLog, we used symmetrized 10-NN graphs as input. For ws-SNE, we adopted the Cauchy kernel, spectral direction optimization (Vladymyrov & Carreira-Perpiñán, 2012) and scalable implementation with Barnes-Hut trees (van der Maaten, 2013; Yang et al., 2013). Both ws-SNE and t-SNE were run for the maximum 1000 iterations. We used default settings for the other compared methods (graphviz uses sfdp layout; Hu, 2005). The compared methods were used to visualize six data sets, two vectorial data and four network data. The descriptions of the data sets are given in the supplemental document. Figure 1 shows the resulting visualizations.

First let us look at the results for vectorial data. In a desired layout of `shuttle` and `MNIST`, data points should be grouped according to ground truth classes shown as colors. In this respect, graphviz and ws-SNE are successful for `shuttle`. In contrast, LinLog fails badly by mixing up the classes in a hairball. In the t-SNE layout, the classes were broken up into unconnected small groups, without a meaningful macro structure being visible. For `MNIST`, t-SNE and ws-SNE correctly identify most classes through clear gaps between the class point clouds; ws-SNE has even better AUC. In contrast, graphviz and LinLog perform much worse, heavily mixing and overlapping the classes.

Next we look at the visualizations of network data. For the `worldtrade` data set, a country generally has higher trading amounts with its neighboring countries. Therefore, a desired 2D visualization of the countries should correlate with their geographical layout. Here we illustrate the continents of the countries by colors. We can see that graphviz, LinLog and ws-SNE can basically group the countries by their continents. In the high-resolution images annotated with country labels (in the supplemental document), we can also identify some known regional groups such as Scandinavian and Latin-American countries in the LinLog and ws-SNE visualizations. In contrast, Figure 1 C3 shows a typical failure of t-SNE for graphs with imbalanced degrees: the high-degree nodes are crowded in the middle while those with low degrees are scattered in the periphery.

In the `usair97` data set, the network links denote whether two cities have direct flights or not. A desired visualization of the data should correlate with geographical locations of the airports. LinLog and ws-SNE are more successful in this sense. We present the names of the 50 biggest airports (by degrees) in the supplemental document, where we see that the geographical topology is almost recovered in these two visualizations except southeast airports. In contrast, graphviz and t-SNE are problematic in this task, especially for placing continental airports; they tend to squeeze big airports in the middle and dangle small ones in periphery.

For `mirex07`, a desired visualization should be able to illustrate the music genres (shown as colors) of the songs.

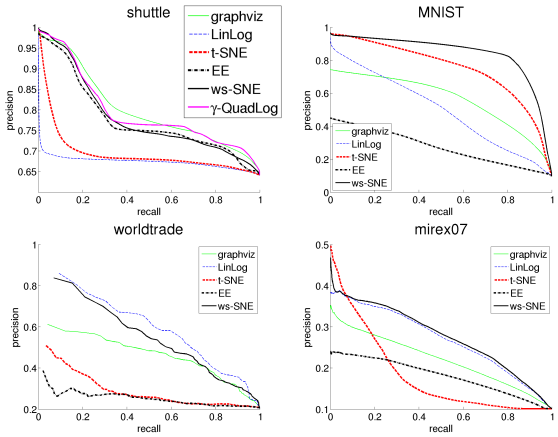

Figure 2. Recall-precision curves of the retrieval by KNN in the 2D space (with different K's) for the four data sets with ground truth classes. Corresponding areas under the curve (AUCs) are shown in Figure 1. This figure is best viewed in color.

LinLog and ws-SNE perform better for this purpose. In contrast, graphviz has a "broccoli-like" display which can only barely show the classes. The t-SNE method again suffers from the imbalanced degree problem: a lot of low-degree nodes occupy a large area and squeeze the high-degree nodes into the middle.

For `luxembourg` the ground truth is the geographical coordinates of the nodes as the edges are streets (see the supplemental document). However, visualization methods can easily fall into trivial local optima, for example, LinLog and t-SNE simply give meaningless displays like hairballs. In contrast, graphviz and ws-SNE successfully present a structure with much fewer crossing edges and higher resemblance to the ground truth, even though the geographical information was not used in their learning.

We fixed $\lambda = 1$ in EE and the results are given in D1-D6 in Figure 1. In this setting, EE only works well for `shuttle` and fails badly for all the other datasets. This indicates that EE performance heavily relies on λ. The EE layouts tend to be uniform with too large λ's (e.g. D3 and D4) or fail to unfold the structure with too small λ's (e.g. D2 and D6).

Besides qualitative results, we also quantify the visualization performance of data sets with ground truth classes. We plot the curves of mean precision vs. mean recall of retrieval in the KNN neighborhoods (in the visualization) across different K's in Figure 2. The area under the curves (AUCs) are shown in Figure 1. The quantified results show that the ws-SNE method performs the best or very close to the best for the four data sets.

In summary, ws-SNE is the only method that gives good visualizations over all the six data sets. The other four methods can only discover the data structures in either some vectorial or network data, but not over all data of both types.

We also provide a preliminary result of γ-QuadLog for

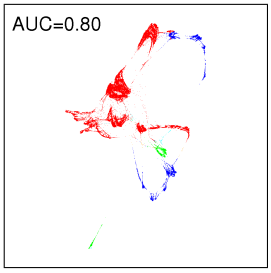

Figure 3. Visualization of `shuttle` using γ-QuadLog.

the `shuttle` dataset, where we used the random walk smoothing ($\rho = 0.99$) and gradient descent optimization, with the step size selected by backtracking to guarantee the cost function decreases. The visualization is given in Figure 3, and the ROC curve in Figure 2 (top left). γ-QuadLog ties with graphviz as the best in retrieval performance (AUC). Note that zooming the display to a dense area as in EE and ws-SNE was not needed for γ-QuadLog.

7. Conclusions

We proved an optimization equivalence theorem between families of information divergence measures. The theorem, together with the known relationships within the families, provides a powerful framework for designing approximative learning algorithms. Many nonlinear dimensionality reduction and graph drawing methods can be shown to be neighbor embedding methods with one of the divergences. As examples, we used the theorem to develop two new visualization methods. Remarkably, the ws-SNE variant works well for both vectorial and network data.

In this paper the divergence measure was selected manually. Methods exist to automatically select the best β-divergence (e.g. Simsekli et al., 2013; Lu et al., 2012). Our finding on divergence connections could extend the methods to γ-divergence and to α- and Rényi-divergences by a suitable transformation from β to α.

Both kinds of divergences have good properties: separable divergences are easy to modify, thus we started method derivations from them, whereas non-separable divergences have appealing invariances and need no connection scalars. The benefit of changing to non-separable divergences is greatest when many connection scalars are needed on the separable side; in our examples the change got rid of a single tradeoff parameter, in cases involving many such parameters the benefit would be even greater.

We gave preliminary empirical results of the two newly developed methods. In ws-SNE, one could additionally use our framework and equivalence theorem to analyze effects of other centrality measures for the edge repulsion and even other weighting schemes. The γ-QuadLog method will be tested on more datasets and with more advanced optimization methods in the future.

Acknowledgment

Academy of Finland, grants 251170, 140398, 252845, and 255725. Tekes Reknow project.

Appendix: proofs

Lemma 7 $\arg\min_z af(z) = \arg\min_z a \ln f(z)$ for $a \in \mathbb{R}$ and $f(z) > 0$. **Proof**: by the monotonicity of $\ln()$.

Proof of Theorem 1. Next we prove the first part in Theorem 1. For $\tau \notin \{0, 1\}$, $\partial D_{\beta\to\tau}(p||cq)/\partial c = 0$ gives $c^* = (\sum_i p_i q_i^{\tau-1})/(\sum_i q_i^\tau)$. Putting it back, we obtain $\min_q [\min_{c\geq 0} D_{\beta\to\tau}(p||cq)] = \min_q \frac{1}{\tau(\tau-1)} \big[\sum_i p_i^\tau - \big(\sum_i p_i q_i^{\tau-1}\big)^\tau \big(\sum_j q_j^\tau\big)^{1-\tau}\big]$. Dropping the constant $\frac{1}{\tau(\tau-1)} \sum_i p_i^\tau$, and by Lemma 7, the above is equivalent to minimizing $\frac{1}{\tau} \ln(\sum_j q_j^\tau) - \frac{1}{\tau-1} \ln(\sum_i p_i q_i^{\tau-1})$. Adding a constant $\frac{1}{\tau(\tau-1)} \ln(\sum_i p_i^\tau)$, the objective becomes $D_{\gamma\to\tau}(p||q)$.

We can apply a similar technique to the second part in Theorem 1. $\partial D_{\alpha\to\tau}(p||cq)/\partial c = 0$ gives $c^* = \big[(\sum_i q_i)^{-1} \sum_i p_i^\tau q_i^{1-\tau}\big]^{1/\tau}$ for $\tau \notin \{0, 1\}$. Putting it back, we obtain $\min_q [\min_{c\geq 0} D_{\alpha\to\tau}(p||cq)] = \min_q \frac{1}{\tau-1} \big[\sum_i p_i^\tau \tilde{q}_i^{1-\tau}\big]^{1/\tau} + \frac{1}{1-\tau} \sum_i p_i$ with $\tilde{q}_i = q_i / \sum_j q_j$. Dropping the constant $\frac{\sum_i p_i}{1-\tau}$, and by Lemma 7, the above is equivalent to minimizing $\frac{1}{\tau-1} \ln \sum_i p_i^\tau \tilde{q}_i^{1-\tau}$ for $\tau > 0$. Adding a constant $\frac{\tau}{1-\tau} \ln \sum_i p_i$ to this equation, the objective becomes $D_{r\to\tau}(p||q)$.

The proofs for the special cases are similar:

1) $\beta = \gamma \to 1$ (or $\alpha = r \to 1$): $\partial D_{\beta\to 1}(p||cq)/\partial c = 0$ gives $c^* = (\sum_i p_i)/(\sum_i q_i)$. Putting it back, we obtain $D_{\beta\to 1}(p||c^*q) = (\sum_i p_i)\, D_{\gamma\to 1}(p||q)$.

2) $\beta = \gamma \to 0$: $\partial D_{\beta\to 0}(p||cq)/\partial c = 0$ gives $c^* = \frac{1}{M} \sum_i (p_i/q_i)$, where M is the length of p. Putting it back, we obtain $D_{\beta\to 0}(p||c^*q) = M D_{\gamma\to 0}(p||q)$.

3) $\alpha = r \to 0$: $\partial D_{\alpha\to 0}(p||cq)/\partial c = 0$ gives $c^* = \exp\big(-\sum_i \tilde{q}_i \ln(q_i/p_i)\big)$, where $\tilde{q}_i = q_i / \sum_j q_j$. Putting it back, we obtain $D_{\alpha\to 0}(p||c^*q) = -\exp\big(-\sum_i \tilde{q}_i \ln \frac{\tilde{q}_i}{p_i}\big) + \sum_i p_i$. Dropping the constant $\sum_i p_i$, minimizing $D_{\alpha\to 0}(p||c^*q)$ is equivalent to minimizing $\sum_i \tilde{q}_i \ln \frac{\tilde{q}_i}{p_i}$. Adding the constant $\ln \sum_j p_j$ to the latter, the objective becomes $D_{r\to 0}(p||q)$.

Proof of Proposition 2. When proving Theorem 1, the optimization equivalence consists of three steps: 1) substituting the closed form solution of the connection scalar, 2) adding or subtracting constants, and 3) applying Lemma 7. Obviously 1) and 2) do not change the stationary points. For 3), the stationary point condition of $a \ln(z)$ is $[a \ln f(z)]' = \frac{af'(z)}{f(z)} = 0$. For $f(z) > 0$, this is equivalent to $af'(z) = 0$, the condition of stationary points of $af(z)$.

References

Belkin, M. and Niyogi, P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, pp. 585–591, 2002.

Bunte, K., Haase, S., Biehl, M., and Villmann, T. Stochastic neighbor embedding (SNE) for dimension reduction and visualization using arbitrary divergences. *Neurocomputing*, 90:23–45, 2012.

Carreira-Perpiñán, M. The elastic embedding algorithm for dimensionality reduction. In *ICML*, pp. 167–174, 2010.

Cichocki, A., Zdunek, R., Phan, A. H., and Amari, S. *Nonnegative Matrix and Tensor Factorization*. John Wiley and Sons, 2009.

Cichocki, A., Cruces, S., and Amari, S.-I. Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization. *Entropy*, 13:134–170, 2011.

Eades, P. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

Févotte, C., Bertin, N., and Durrieu, J.-L. Nonnegative matrix factorization with the Itakura-Saito divergence with application to music analysis. *Neural Computation*, 21 (3):793–830, 2009.

Fruchterman, T. and Reingold, E. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

Gansner, E., Hu, Y., and North, S. A maxent-stress model for graph layout. *IEEE Transactions Visualization and Computer Graphics*, 19(6):927–940, 2013.

Hinton, G.E. and Roweis, S.T. Stochastic neighbor embedding. In *NIPS*, pp. 833–840, 2002.

Hofmann, T. Probabilistic latent semantic indexing. In *SIGIR*, pp. 50–57, 1999.

Hu, Y. Efficient and high quality force-directed graph drawing. *The Mathematica Journal*, 10:37–71, 2005.

Jacomy, M., Heymann, S., Venturini, T., and Bastian, M. Forceatlas2, a graph layout algorithm for handy network visualization, 2011. available at http://webatlas.fr/tempshare/ForceAtlas2_Paper.pdf.

Lawrence, N. Gaussian process latent variable models for visualisation of high dimensional data. In *NIPS*, 2003.

Lu, Z., Yang, Z., and Oja, E. Selecting β-divergence for nonnegative matrix factorization by score matching. In *ICANN*, pp. 419–426, 2012.

Martin, S., Brown, W., Klavans, R., and Boyack, K. OpenOrd: an open-source toolbox for large graph layout. In *Proceedings of SPIE conference on Visualization and Data Analysis*, 2011.

Minka, T. Divergence measures and message passing. Technical report, Microsoft Research, 2005. URL http://research.microsoft.com/~minka/papers/message-passing/.

Noack, A. Energy models for graph clustering. *Journal of Graph Algorithms and Applications*, 11(2):453–480, 2007.

Simsekli, U., Yilmaz, Y., and Cemgil, A. Learning the beta-divergence in Tweedie compound poisson matrix factorization models. In *ICML*, pp. 1409–1417, 2013.

van der Maaten, L. Barnes-Hut-SNE. In *ICLR*, 2013.

van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

van der Maaten, L. and Hinton, G. Visualizing non-metric similarities in multiple maps. *Machine Learning*, 87(1): 33–55, 2011.

Venna, J. and Kaski, S. Nonlinear dimensionality reduction as information retrieval. In *AISTATS*, pp. 572–579, 2007.

Venna, J., Peltonen, J., Nybo, K., Aidos, H., and Kaski, S. Information retrieval perspective to nonlinear dimensionality reduction for data visualization. *Journal of Machine Learning Research*, 11:451–490, 2010.

Vladymyrov, M. and Carreira-Perpiñán, M. Partial-hessian strategies for fast learning of nonlinear embeddings. In *ICML*, pp. 167–174, 2012.

Weinberger, K.Q. and Saul, L.K. Unsupervised learning of image manifolds by semidefinite programming. *International Journal of Computer Vision*, 70:77–90, 2006.

Yang, Z., King, I., Xu, Z., and Oja, E. Heavy-tailed symmetric stochastic neighbor embedding. In *NIPS*, pp. 2169–2177, 2009.

Yang, Z., Peltonen, J., and Kaski, S. Scalable optimization of neighbor embedding for visualization. In *ICML*, pp. 127–135, 2013.

Zhou, D., Bousquet, O., Lal, T., Weston, J., and Schölkopf, B. Learning with local and global consistency. In *NIPS*, 2003.