Generative Modeling for Maximizing Precision and Recall in Information Visualization

Jaakko Peltonen[1] **Samuel Kaski**[1,2]
[1]Aalto University, Department of Information and Computer Science,
Helsinki Institute for Information Technology HIIT, P.O. Box 15400, FI-00076 Aalto, Finland
[2]University of Helsinki, Department of Computer Science,
Helsinki Institute for Information Technology HIIT, P.O. Box 68, 00014 University of Helsinki, Finland

Abstract

Information visualization has recently been formulated as an information retrieval problem, where the goal is to find similar data points based on the visualized nonlinear projection, and the visualization is optimized to maximize a compromise between (smoothed) precision and recall. We turn the visualization into a generative modeling task where a simple user model parameterized by the data coordinates is optimized, neighborhood relations are the observed data, and straightforward maximum likelihood estimation corresponds to Stochastic Neighbor Embedding (SNE). While SNE maximizes pure recall, adding a mixture component that "explains away" misses allows our generative model to focus on maximizing precision as well. The resulting model is a generative solution to maximizing tradeoffs between precision and recall. The model outperforms earlier models in terms of precision and recall and in external validation by unsupervised classification.

1 INTRODUCTION

The importance of information visualization as a central part of data analysis has been evident in exploratory branches of statistics, called for instance exploratory data analysis, and the importance of visualization is being emphasized in the current strong visual analytics movement. Machine learning seems to have an obvious contribution to the field through nonlinear dimensionality reduction. Many nonlinear dimensionality reduction methods developed during the past ten years have been designed for manifold learning, including Isomap (Tenenbaum et al., 2000), Locally Linear Embedding (Roweis and Saul, 2000), Stochastic Neighbor Embedding (Hinton and Roweis, 2002), Maximum Variance Unfolding (Weinberger and Saul, 2006), Laplacian Eigenmap (Belkin and Niyogi, 2002), and their more recent variants (see for example Zhang and Wang, 2007; Choi and Choi, 2007; van der Maaten and Hinton, 2008; Song et al., 2008). See van der Maaten et al. (2009) for a recent comparison of methods. At first sight it might then seem attractive to simply use manifold learning methods for visualization. However, the manifold learning methods have not been designed or optimized for visualization and hence may not work well for visualization if the inherent dimensionality of the data manifold is larger than the display dimension (Venna et al., 2010). While there now are several more or less rigorous formulations for the manifold learning problem, there are not many for the visualization problem.

Visualization has recently been formulated as a visual information retrieval task (Venna et al., 2010), with the goal being to organize points on the display such that if similar points are retrieved based on the display, the accuracy of retrieving truly similar data is maximized. As in all information retrieval, the result necessarily is a compromise between precision and recall, of minimizing false positives and misses. Stochastic Neighbor Embedding (SNE) corresponds to maximizing recall.

We also take SNE as the starting point because it works well and has a nice interpretation explicated below: The cost function is (mean) maximum likelihood of a simple user model, where the user is assumed to pick neighbors according to a kernel probability distribution on the display. The data are the actual neighbor relationships, in practice often given by specifying

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15. Copyright 2011 by the authors.

a kernel as well. Now the question we asked is: If maximizing recall is a generative modeling task, could a generative model be made to focus on precision as well, or in fact on any tradeoff between the two?

We formulate information visualization as a generative modeling task, which reduces to SNE when maximizing pure recall, and precision is maximized by a mixture model. When a mixture component is added to explain away the misses the rest of the model will focus more on minimizing false positives. This turns the whole visualization task into a generative modeling task which makes it more understandable for modelers, easier to extend and, as it turns out, makes the visualizations better. Our cost function, in contrast to Venna et al. (2010), is directly a likelihood of observed neighborhoods. This makes visualization a rigorous statistical modeling task, with all tools of generative modeling available.

2 GENERATIVE MODELING FOR VISUALIZATION

Consider visualization as a model learning task, where observed similarity relationships are the data and the coordinates of points on the display are the parameters. We construct a generative model which will generate neighbor relationships for query points, and can naturally generate a distribution over query points too although we do not consider that straightforward extension in this paper. The model can be considered as a user model, that is, a model that specifies which other points the user would inspect given the query point. When the visualization is optimized for the specific user model (neighborhood kernel), it will naturally be optimal for a user behaving according to that model.

If the data consists of observed neighborhood relationships, for instance as counts of citations in a paper or counts of social interactions of a person, we can use them directly or, assuming large sample size, normalize them into distributions. Let p_{ij} denote the "ground truth" probability that j would be chosen as a neighbor of i without any constraints coming from the visualization, and $\sum_{j \neq i} p_{ij} = 1$ for all i. In practice the analysis often starts with a kernel, or distance measure and functional form. In that case, we denote the density after appropriate normalization by p.

Let now probabilities r_{ij} denote the neighborhood relationships of the model; r_{ij} is the probability of choosing point j as a neighbor of point i, $\sum_{j \neq i} r_{ij} = 1$ for all i. The r_{ij} are interpretable as a "user model" as follows: r_{ij} is the probability with which the model believes a user will inspect point j when the query point is i, given the visualization. The user model is parameterized by the coordinates $\mathbf{y}_j$ of each point j on the visualization display. Many definitions of r_{ij} can be used depending on the needs of the analyst; we will use the simple Gaussian falloff around the query point i:

$$r_{ij} = \frac{\exp(-||\mathbf{y}_i - \mathbf{y}_j||^2/\sigma_i^2)}{\sum_{k \neq i} \exp(-||\mathbf{y}_i - \mathbf{y}_k||^2/\sigma_i^2)} \tag{1}$$

where σ_i is a neighborhood radius around point i. Another recent possibility is a t-distributed falloff (van der Maaten and Hinton, 2008) which can be easily included.

Simple generative modeling to maximize recall. Now consider simply maximizing the log-likelihood of the observed neighborhoods, that is, maximizing

$$\sum_i \sum_{j \neq i} p_{ij} \log r_{ij} \ . \tag{2}$$

This corresponds to minimizing $\sum_i D_{KL}(p_{i\cdot}, r_{i\cdot})$, the sum of Kullback-Leibler divergences from the observed neighborhoods to the user model, which is the cost function of Stochastic Neighbor Embedding (SNE; Hinton and Roweis, 2002). This is a straightforward re-interpretation of SNE.

We then consider a simple user model in order to build a connection to recall, extending the work of Venna et al. (2010). Assume that the user (or retrieval model) retrieves a set R_i of points as neighbors of query point i, and places a uniform distribution $r_{ij} = (1-\epsilon)/|R_i|$ across the retrieved points with a very small probability $\epsilon/(N-1-|R_i|)$ for others, where ϵ is a very small number and $N-1$ is the total number of points other than i. Similarly, let the set of actually relevant neighbors be P_i, with a uniform distribution $p_{ij} = (1-\epsilon)/|P_i|$ across the relevant neighbors and very small probabilities for the rest. Then the objective function for a single query point i becomes

$$\begin{aligned}\sum_{j \neq i} p_{ij} \log r_{ij} &\approx \sum_{j \in P_i \cap R_i} \frac{1-\epsilon}{|P_i|} \log\left(\frac{1-\epsilon}{|R_i|}\right) \\ &\quad + \sum_{j \in P_i \cap R_i^c} \frac{1-\epsilon}{|P_i|} \log\left(\frac{\epsilon}{N-1-|R_i|}\right) \\ &\approx \frac{N_{TP,i}}{|P_i|} \log\left(\frac{1}{|R_i|}\right) + \frac{N_{MISS,i}}{|P_i|} \log \epsilon \end{aligned} \tag{3}$$

where R_i^c and P_i^c denote complements of R_i and P_i, $N_{TP,i} = |P_i \cap R_i| = |R_i| - N_{FP,i}$ is the number of true positives (retrieved relevant points), $N_{FP,i} = |R_i \cap P_i^c|$ is the number of false positives (retrieved non-relevant points), and $N_{MISS,i} = |P_i \cap R_i^c|$ is the number of misses (relevant non-retrieved points). With small ϵ

the rightmost term in (3) dominates, and maximizing the objective function (3) with respect to the retrieval distribution defined by the r_{ij} is equivalent to minimizing the number of misses, that is, maximizing $recall = N_{TP,i}/|P_i| = 1 - N_{MISS,i}/|P_i|$. Therefore SNE, which maximizes (3), can be seen as a generative model of neighborhood relationships which maximizes recall.

2.1 Extending the generative model for flexible visualization goals

We showed above that maximizing the likelihood for the simple retrieval model corresponds to maximizing recall, and it also corresponds to the objective of SNE.

However, maximizing recall is only one possible goal of successful visualization: it corresponds to minimizing *misses* (missed true neighbors), but it ignores the other type of visualization error, *false positives*. Minimizing false positives would be equivalent to maximizing *precision* of retrieving neighbors from the visualization. Both precision and recall, or any tradeoff between them, are useful optimization goals for visualization. We next show that we can change the retrieval model to optimize a tradeoff between precision and recall, while keeping the same rigorous generative modeling approach which we introduced above.

Notice that the simple analysis above already gives a hint on how to proceed; equation (3) does involve the number of true positives $N_{TP,i}$ (or equivalently the number of false positives $N_{FP,i}$) in the first term on the right-hand side. However, this term does not have much influence on optimization because the cost function is in practice dominated by the second term involving misses. For small ϵ the second term is always much larger than the first, therefore misses are likely to dominate. If we could somehow change the model so that the cost of misses becomes less dominant, the model would be able to focus also on false positives.

More flexible generative modeling to maximize a tradeoff between precision and recall. Let us design a more flexible retrieval model q_{ij} which extends r_{ij}. We will define q_{ij} as a mixture of two retrieval mechanisms: the user model r_{ij} which depends on the visualization coordinates of points, and an additional model which need not depend on the visualization coordinates; the goal of the additional model is to *explain away* those neighbors that the user model r_{ij} misses. We give the precise definitions soon.

Intuitively, if we can create an additional retrieval mechanism which retrieves all those neighbors that the user model r_{ij} misses, then when we fit the combined model to maximize the likelihood of observed neighborhoods, the user model r_{ij} (which is part of the functional form of q_{ij}) can minimize the remaining kind of error, the number of retrieved false positives.

A simple solution is to define the retrieval distribution q_{ij} as a mixture of the plain user model r_{ij} and an "explaining away" model:

$$q_{ij} \propto r_{ij} + \gamma p_{ij} \tag{4}$$

where $\gamma \geq 0$ is a multiplier which controls the amount of explaining away. The model is again fitted to the observed neighborhoods by maximizing the log-likelihood

$$L = \sum_i \sum_{j \neq i} p_{ij} \log q_{ij} \tag{5}$$

with respect to the output coordinates $\mathbf{y}_i$ of all data points, which affect the q_{ij} through the plain user model r_{ij} .

It is easy to see that the explaining-away has no effect in the perfect retrieval case where r_{ij} already equals p_{ij} (then $q_{ij} = r_{ij}$); instead, the explaining-away affects how severely errors in r_{ij} affect the likelihood. In the log-likelihood (5) the explaining-away has the largest effect on the terms corresponding to misses, where r_{ij} is small but p_{ij} is large; for such terms q_{ij} is also large and the cost of misses thus no longer dominates the likelihood. Therefore, optimizing q_{ij} with respect to the visualization coordinates is now able to better take into account the false positives, and hence the visualization will be better arranged to avoid false positives.

An analysis of the mixture model likelihood. In the simple case that we discussed above, where the observed neighborhoods p_{ij} and plain user models r_{ij} are uniform over some subsets of points P_i and R_i respectively, and near-zero elsewhere, it can be shown that the log-likelihood of the mixture model for a single query point becomes

$$\begin{aligned}\sum_{j \neq i} p_{ij} \log q_{ij} &\approx \mathit{const.} \\ &+ \mathit{recall} \cdot \log\left(\frac{\left(\frac{\mathit{precision}}{\mathit{recall}} + \gamma\right)\left(a - \frac{\mathit{recall}}{\mathit{precision}}\right)}{\gamma\left(a - \frac{\mathit{recall}}{\mathit{precision}}\right) + \frac{\epsilon}{1-\epsilon}}\right) \\ &+ (1-\epsilon)\log\left(\frac{\gamma(1-\epsilon)}{|P_i|} + \frac{\epsilon}{N - |R_i| - 1}\right)\end{aligned} \tag{6}$$

where $a = (N-1)/|P_i|$, and the information retrieval criteria are $\mathit{recall} = N_{TP,i}/|P_i|$ and $\mathit{precision} = N_{TP,i}/|R_i|$ as usual. With no explaining-away ($\gamma = 0$), maximizing (6) reduces to maximizing equation (3), that is, maximizing the mixture model likelihood without explaining-away is the same as maximising $\mathit{recall} \cdot \mathit{const.}$ and ignoring precision. However, with a sufficient amount of explaining away such that $\gamma \gg \epsilon > 0$ the above reduces to the more appealing form

$$\sum_{j \neq i} p_{ij} \log q_{ij} \approx const.{+}recall{\cdot}\log\left(1 + \frac{1}{\gamma} \cdot \frac{precision}{recall}\right) \tag{7}$$

where we can see that, because of the explaining-away, the objective function is affected both by precision (false positives) and recall (misses). The influence of precision is strongest when γ is small but still clearly larger than ϵ.

In more detail, the log-likelihood is dominated by a sum over misses and a sum over true positives. If γ is much smaller than ϵ, the misses dominate the cost function, which reduces to maximization of recall. On the other hand, as γ grows, it begins to affect the log-likelihood of true positives: r_{ij} for true positives depends on the number of retrieved points $|R_i|$ and hence depends on precision; however, for asymptotically large γ, $q_{ij} \propto r_{ij} + \gamma p_{ij}$ for true positives becomes nearly constant with respect to the visualization, which then reduces the influence of precision on optimization. For more details, see the full derivation of equation (7) in the supplement.

To summarize, in order to make precision influence the cost function as much as possible, γ should be sufficiently larger than zero so that it can explain away the misses. Otherwise, γ should be kept small: then the log-likelihood of true positives depends on the retrieval distribution r_{ij} rather than being explained away. In our experiments we use $\gamma = 0.9$ which yielded very good results.

The objective function can be maximized with respect to the output coordinates $\mathbf{y}_i$ by gradient methods; here we use conjugate gradient. The computational complexity per iteration is $O(N^2)$ for N data points which is the same complexity as for SNE. To help avoid local minima, we first run the method with no explaining-away ($\gamma = 0$) and use the resulting coordinates $\mathbf{y}_i$ as initialization for the final run with the desired amount of explaining-away (desired γ value).

2.2 Comparison to regularization

The functional form of our retrieval distribution q_{ij} is superficially similar to regularization: the user model r_{ij} is mixed with another distribution which keeps crucial retrieval probabilities nonzero. Regularized variants of stochastic neighbor embedding have been proposed earlier: in particular, UNI-SNE (Cook et al., 2007) is a variant of SNE where the retrieval distribution is regularized by mixing it with a uniform distribution, which is equivalent to $q_{ij} \propto r_{ij} + const.$ The problem with such regularization is that it distorts the retrieval model and hence cannot achieve the optimal embedding result. Because the regularization always mixes a constant to all retrieval probabilities, the user model is forced to compensate for this regularization which distorts the embedding.

It can be shown that even if a perfect embedding (where $r_{ij} = p_{ij}$) is possible, for example when the original data lives on a low-dimensional subspace, the UNI-SNE optimum does not correspond to that perfect embedding. (This can be seen by taking the gradient of the UNI-SNE cost function with respect to r_{ij}, enforcing nonnegativity and sum-to-one constraints by reparameterization, and showing that $r_{ij} = p_{ij}$ is not a zero-point of that gradient.)

In contrast, our method mixes the user model with the "perfect retrieval" distribution p_{ij}, which is data-dependent and non-uniform. This is a true "explaining away" model which does not distort the embedding: it is easy to show that if perfect embedding (where $r_{ij} = p_{ij}$) is possible, it corresponds to the optimum of our method, as desired. To show this, simply note that if $r_{ij} = p_{ij}$ then also $q_{ij} = p_{ij}$ which yields the maximum value of the log-likelihood $\sum_{i,j \neq i} p_{ij} \log q_{ij}$, or equivalently the minimal value of $\sum_i D_{KL}(p_{i\cdot}, q_{i\cdot})$ where the D_{KL} are Kullback-Leibler divergences between the relevance probabilities p_{ij} and the q_{ij}. Therefore, if $r_{ij} = p_{ij}$ can be achieved, it corresponds to the optimum of our method.

In summary, the new method can be seen as a rigorous approach to the same problem that has been previously addressed by regularization approaches like UNI-SNE. Our new method also has a novel interpretation and an analysis in terms of precision and recall; and it corrects a problem present in UNI-SNE, so that the new method is able to find the optimal embedding.

3 EXPERIMENTS

We compare our new method to several previous methods, first in terms of retrieval performance and then in terms of unsupervised classification performance; lastly, we plot visualizations produced by our method on several data sets.

3.1 Comparison of retrieval performance

We evaluate the performance of the new method against a comprehensive set of alternatives on two data sets, in the task of visualizing the sets as scatterplots in 2D. The *Faces* data set (`http://www.cs.toronto.edu/~roweis/data.html`) contains 400 face images, from 40 people with 10 images each, with different facial expressions and lighting; each image is

64×64 pixels with 256 grey levels. The *Seawater temperature time series* (Liitiäinen and Lendasse, 2007) contains weekly measurements of seawater temperature over several years. Each data point is a 52-week window of the temperature time series, and for the next data point the window is shifted one week forward; this yields 823 data points with 52 dimensions.

We compare our method with thirteen others: Principal Component Analysis (PCA; Hotelling, 1933), Metric Multidimensional Scaling (MDS; see Borg and Groenen, 1997), Locally Linear Embedding (LLE; see Roweis and Saul, 2000), Laplacian Eigenmap (LE; Belkin and Niyogi, 2002), Hessian-based Locally Linear Embedding (HLLE; Donoho and Grimes, 2003), Isomap (Tenenbaum et al., 2000), Curvilinear Component Analysis (CCA; Demartines and Hérault, 1997), Curvilinear Distance Analysis (CDA; Lee et al., 2004), Maximum Variance Unfolding (MVU; Weinberger and Saul, 2006), Landmark Maximum Variance Unfolding (LMVU; Weinberger et al., 2005), Local MDS (LMDS; Venna and Kaski, 2006), Neighbor Retrieval Visualizer (NeRV; Venna et al., 2010), and UNI-SNE (Cook et al., 2007).

We use the same test setup as Venna et al. (2010). In brief, each method was run with several parameter values, and non-convex methods were run from five random initializations. For each method, the best result was chosen in the sense of maximizing the (unsupervised) F-measure computed as $2(P \cdot R)/(P + R)$ where P and R are rank-based smoothed precision and recall measures; see Venna et al. (2010) for details. The NeRV and LocalMDS methods which allow a tradeoff between precision and recall were run with several values of their tradeoff parameter λ; for clarity we show results for a single λ value chosen by the F-measure. We ran our method with two settings: the baseline case $\gamma = 0$ (no explaining-away; corresponds to Stochastic Neighbor Embedding) and $\gamma = 0.9$ (strong explaining-away during training). We ran the corresponding setting for UNI-SNE, with $\lambda = 0.47$ which corresponds to $\gamma = 0.9$ in our method. Note that both the explaining-away in our method and regularization in UNI-SNE are only used during training, and only the resulting visualization (data point locations) are used to evaluate the quality of the method.

The quality of the visualizations is evaluated by how well the real neighborhoods are visible in the visualization or, equivalently, how well the real neighbors (set to be the 20 nearest neighbors of points in the original space) can be retrieved based on the visualization. We use traditional precision–recall curves to measure this.

Based on Figure 1 our new method (denoted "NM" in the figures) performs very well: it attains clearly the best precision. In terms of recall it is roughly as good as NeRV or, equivalently, SNE. The simple regularization approach UNI-SNE also performs fairly well, but our more rigorous approach achieves better results.

3.2 Unsupervised classification

We additionally compare the methods using external validation, computing unsupervised 2D displays and then measuring how well known but so far unused classes are separated on the display. Class separation is measured by classification accuracy of a k-nearest neighbor classifier ($k = 5$) operating on the display coordinates; each point is classified according to a majority vote of its k nearest neighbors excluding itself.

Four data sets are used. The *Letter* recognition data set is from the UCI machine learning repository (Blake and Merz, 1998) and contains 4×4 images of capital letters, based on distorting letter shapes in different fonts; the data set has 16 dimensions and 26 classes. The *Phoneme* data set is from LVQ-PAK (Kohonen et al., 1996) and contains spoken phoneme samples; the data are 20-dimensional and there are 13 classes (different phonemes). The *Landsat* satellite data set is from the UCI machine learning repository; it contains satellite images, each of which is 3×3 and measured in four spectral bands, yielding 36 dimensions per image. Each image is classified into one of 6 classes which denote different soil types. The *TIMIT* data set is from the DARPA TIMIT speech database (TIMIT, 1998); it contains phoneme samples, each of which is 12-dimensional, and there are 41 classes.

Our new method ("NM" in Table 1) with strong explaining-away during training ($\gamma = 0.9$) yields the best results on two data sets (Landsat and TIMIT), second-best on one (Letter), and third-best on one (Phoneme). The use of explaining-away during training clearly improves results on all data sets compared to the no explaining-away case ($\gamma = 0$, corresponding to SNE). UNI-SNE performs almost as well: it is best on two data sets (Letter and Phoneme), third-best on one (TIMIT) and fourth-best on one (Landsat). Other methods that perform well are LocalMDS and NeRV.

Although for brevity we report the results of our method only with two choices of γ, the results are very good for all the nonzero gamma values that we tried (between 0.1 and 0.9). On TIMIT our method is best with any such γ value; on Letter, Phoneme and Landsat, our method is always in the top-two, top-three, and top-four respectively.

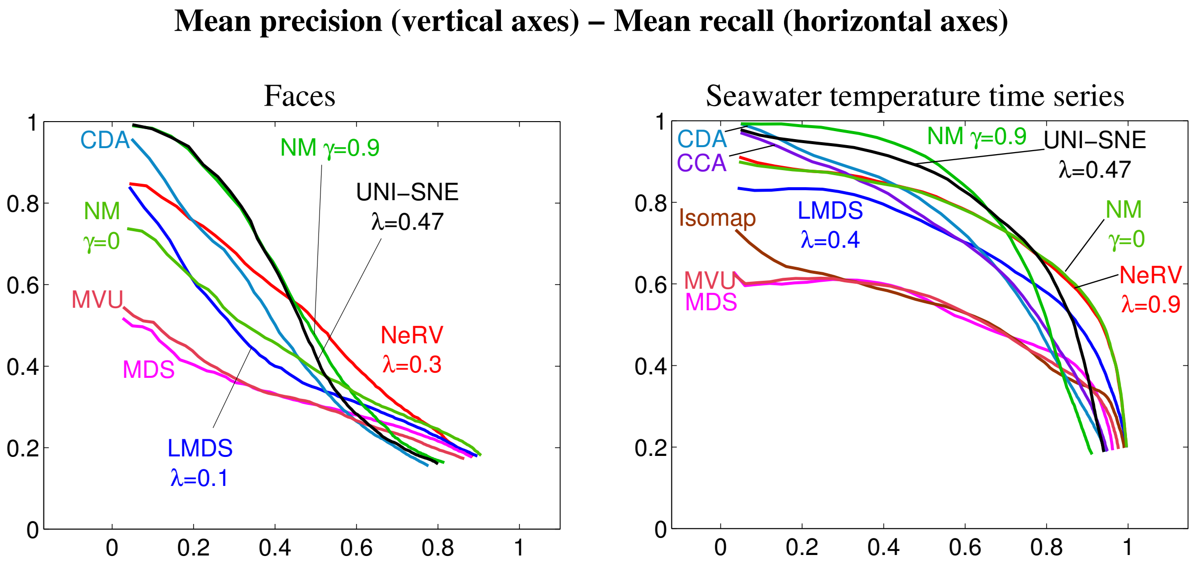

Figure 1: Retrieval quality measures for neighbor retrieval based on the visualizations, for two data sets: *Faces* and *Seawater temperature time series*. For clarity, only a few of the best-performing methods are shown for each data set. Performance is measured by standard precision-recall curves. For NeRV and LocalMDS, for clarity performance is shown for only a single λ chosen by a F-measure. For our method (denoted "NM") we report performance for $\gamma = 0$ (no explaining away; corresponds to Stochastic Neighbor Embedding) and $\gamma = 0.9$ (strong explaining away used during training). For UNI-SNE we report results for $\lambda = 0.47$ which corresponds to our setting $\gamma = 0.9$; UNI-SNE at $\lambda = 0$ is essentially equivalent to our method at $\gamma = 0$. Our new method attains the highest precision for both data sets and is comparable to NeRV/SNE in terms of recall.

Table 1: (In)separability of known classes on unsupervised diplays for four data sets. The cost measure is classification error rate, based on the visualizations, with a k-nearest neighbor classifier, $k = 5$. Our method is the best on two of the four data sets, second-best on one data set (Letter), and third-best on one data set (Phoneme). On Landsat data our method and LocalMDS yield the same accuracy. The best method in each column has been boldfaced.

	Letter	Phon.	Land.	TIMIT
Eigenmap	0.914	0.121	0.168	0.674
LLE	n/a	0.118	0.212	0.722
Isomap	0.847	0.134	0.156	0.721
MVU	0.763	0.155	0.153	0.699
LMVU	0.819	0.208	0.151	0.787
MDS	0.823	0.189	0.151	0.705
CDA	0.336	0.118	0.141	0.643
CCA	0.422	0.098	0.143	0.633
NeRV	0.532	0.079	0.139	0.626
LocalMDS	0.499	0.118	**0.128**	0.637
UNI-SNE, $\lambda = 0.47$	**0.299**	**0.072**	0.136	0.628
NM, $\gamma = 0$	0.590	0.088	0.133	0.657
NM, $\gamma = 0.9$	0.326	0.080	**0.128**	**0.594**

3.3 Demonstrations on toy data, face images, and fMRI data

We demonstrate the visualizations on three data sets. First, we replicate the simple demonstration of the precision–recall tradeoff shown in Venna et al. (2010). Data points are distributed on the surface of a three-dimensional sphere (Figure 2**A**). We create two-dimensional visualizations with our new method, with two settings: no explaining-away ($\gamma = 0$; corresponds to SNE) which concentrates on minimizing misses, and strong explaining-away during training ($\gamma = 0.9$) which concentrates on minimizing false positives. The result trained without explaining-away (Figure 2**B**) minimizes misses by squashing the sphere flat, which leads to numerous false neighbors when points originally on opposing sides of the sphere are placed near each other. With strong explaining-away (Figure 2**C**) false neighbors are minimized by opening up the sphere, at the expense of missing some neighbors across the tear. Both solutions are useful visualizations of the sphere, but for different purposes.

Secondly, we visualize the face images data set which was already used in the previous section. In the plot with $\gamma = 0.9$, Figure 2**D**, faces of the same person become mapped close to each other in the visualization.

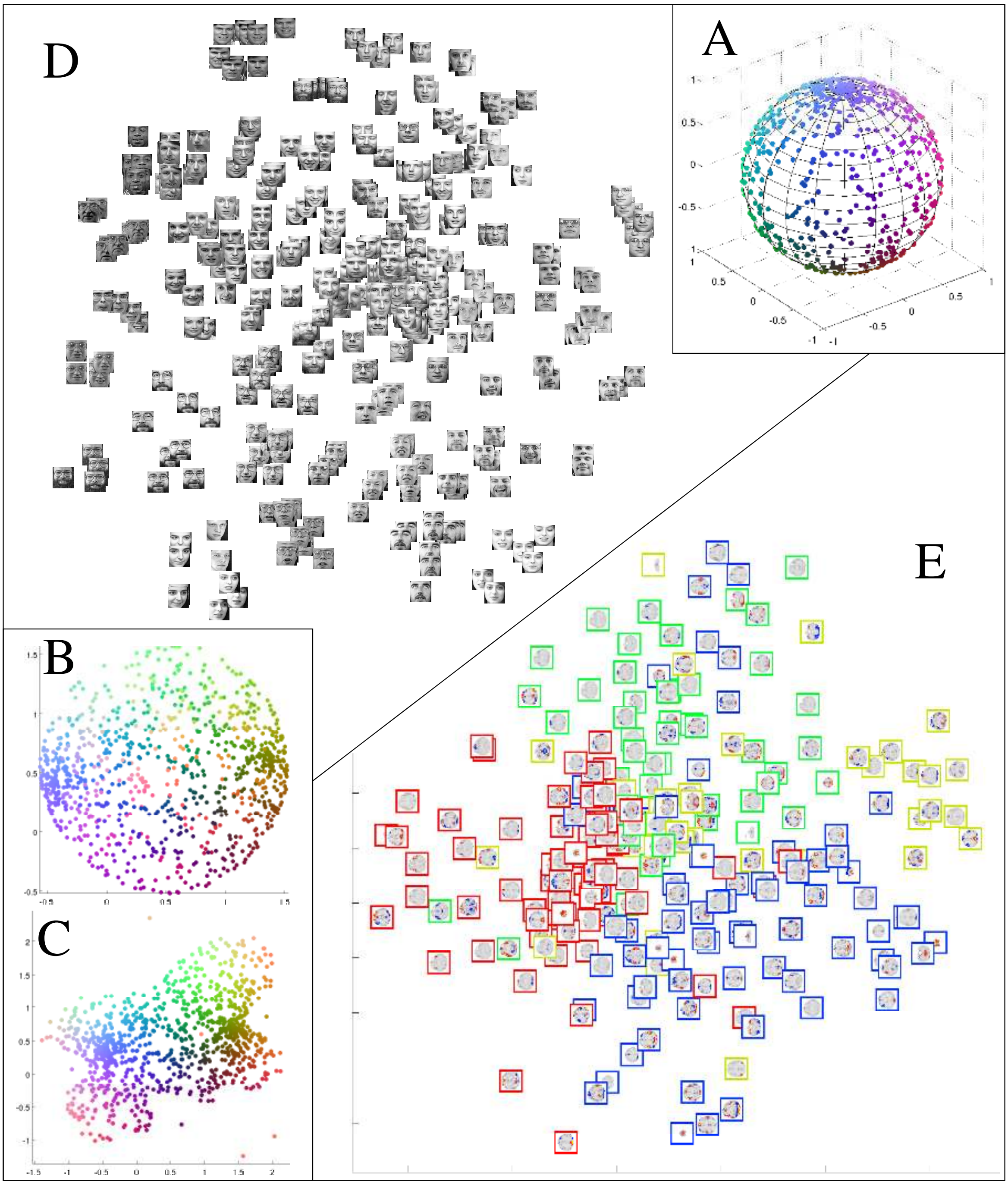

Figure 2: Demonstrations of our method. **A-C** demonstrate the tradeoff between misses and false positives. Points on a three-dimensional sphere (**A**) are mapped to a two-dimensional display by the new method. In **B**, the visualization is optimized without explaining-away ($\gamma = 0$; corresponds to Stochastic Neighbor Embedding) which minimizes misses by squashing the sphere flat. In **C**, the visualization is optimized with strong explaining-away ($\gamma = 0.9$) which minimizes false positives by opening up the sphere. **D**: Face images ($\gamma = 0.9$); faces of the same person occur close to each other. **E**: Visualization of fMRI whole-head volumes from an experiment with several people experiencing multiple stimuli ($\gamma = 0.9$). The four stimuli types (red: tactile, yellow: auditory tone, green: auditory voice, blue: visual) have become separated in the visualization; the two auditory stimuli types are arranged close-by as is intuitively reasonable. An axial slice is shown for each whole-head volume, chosen so that the shown slice contains the highest-activity voxel.

Thirdly, we visualize a set of functional magnetic resonance imaging (fMRI) measurements. The data set Malinen et al. (2007) includes measurements of six healthy young adults in two measurement sessions where they received temporally non-overlapping stimuli: auditory (binaural tones or a male voice), visual (shown video clips), and tactile (touch pulses delivered to fingers). Using an MRI scanner, 161 whole-head volumes (time points) were obtained for each person in each test. Preprocessing of the volumes included realignment, normalization, smoothing, and extraction of 40 components by independent component analysis; see Ylipaavalniemi et al. (2009) for details.

For our purposes we took every fourth time point (whole-head volume) from the first half of each session as a data item to be visualized, yielding $6\times2\times19 = 228$ data items with 40 dimensions. We visualize this data set in two dimensions using our new method with explaining-away ($\gamma = 0.9$) during training. Figure 2**E** shows the result. The different stimuli types are separated in the visualization. This kind of a display is useful for interactive analysis of the experiment, where browsing for evidence of common patterns is interleaved with interactive slicing through the 3D brain volumes to more accurately view the sets of 3D active regions.

4 CONCLUSIONS AND DISCUSSION

We have introduced a novel way to perform nonlinear dimensionality reduction by bringing in the generative modeling framework and a way of controlling the precision and recall of the visualization. The method includes Stochastic Neighbor Embedding (SNE) as a special case, and thus gives a generative interpretation for it, but where SNE minimizes only one kind of error (misses) we allow a flexible amount of *explaining-away* during training to let the model concentrate on reducing the other kind of error, false positives. Our model simply mixes the retrieval “user model” linearly with an explaining-away distribution during training; this remarkably simple model suffices to yield a flexible tradeoff between minimizing misses and minimizing false positives, and in experiments it gives visualizations that outperform alternative methods according to several measures.

Compared to the earlier regularization-based approach UNI-SNE (Cook et al., 2007), our method performs slightly better. Furthermore, it has a novel interpretation and an analysis in terms of precision and recall, and it corrects a problem present in UNI-SNE. In contrast to UNI-SNE, the new method is able to find the optimal embedding.

Compared to a previous approach (Venna et al., 2010) which also minimized a tradeoff between misses and false positives, our novelty is the rigorous generative framework; our cost function is directly a likelihood of observed neighborhoods and we control precision and recall by using a generative model. This makes it easier to analyze the performance and extend the model. In particular, it should now be possible to start to rigorously learn the user model too, on-line or off-line, to adapt to real user behavior and needs.

We have now brought information visualization into the domain of rigorous probabilistic generative modeling. The specific modeling choices were made to show that this is possible; we did not yet make any claims about optimality, in particular about maximization of precision. However, even the proof-of-concept model outperformed existing models in empirical tests, giving strong support to this line of research.

A simple extension is to use alternative distributional assumptions. Instead of the Gaussian falloffs which gave very good results here, there is evidence that t-distributed neighborhoods could work even better for visualizations (van der Maaten and Hinton, 2008).

In this paper the goal is information visualization, where it is natural to have 2-3 output dimensions. Controlling precision and recall with generative modeling may also be useful more generally in dimensionality reduction with higher output dimensionalities.

Acknowledgments

The authors belong to the Adaptive Informatics Research Centre, a national CoE of the Academy of Finland. JP was supported by the Academy of Finland, decision number 123983. This work was also supported in part by the PASCAL2 Network of Excellence, ICT 216886. We thank Jarkko Ylipaavalniemi, from the Department of Information and Computer Science, and Riitta Hari, from the Advanced Magnetic Imaging Centre (AMI Centre) and the Brain Research Unit in the Low Temperature Laboratory, for providing and helping with the fMRI data.

References

Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 585–591. MIT Press, Cambridge, MA.

Blake, C. L. and Merz, C. J. (1998). UCI repository of machine learning databases. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

Borg, I. and Groenen, P. (1997). *Modern Multidimensional Scaling.* Springer, New York.

Choi, H. and Choi, S. (2007). Robust kernel isomap. *Pattern Recognition*, 40:853–862.

Cook, J., Sutskever, I., Mnih, A., and Hinton, G. (2007). Visualizing similarity data with a mixture of maps. In Meila, M. and Shen, X., editors, *Proceedings of AISTATS*07, the 11th International Conference on Artificial Intelligence and Statistics (JMLR Workshop and Conference Proceedings Volume 2)*, pages 67–74.

Demartines, P. and Hérault, J. (1997). Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks*, 8(1):148–154.

Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100:5591–5596.

Hinton, G. and Roweis, S. T. (2002). Stochastic neighbor embedding. In Dietterich, T., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 833–840. MIT Press, Cambridge, MA.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441,498–520.

Kohonen, T., Hynninen, J., Kangas, J., Laaksonen, J., and Torkkola, K. (1996). LVQ_PAK: The learning vector quantization program package. Technical Report A30 (26 pages), Helsinki University of Technology, Laboratory of Computer and Information Science, FIN-02150 Espoo, Finland.

Lee, J. A., Lendasse, A., and Verleysen, M. (2004). Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis. *Neurocomputing*, 57:49–76.

Liitiäinen, E. and Lendasse, A. (2007). Variable scaling for time series prediction: Application to the ESTSP'07 and the NN3 forecasting competitions. In *Proceedings of IJCNN 2007, International Joint Conference on Neural Networks*, pages 2812-2816. IEEE, Piscataway, NJ.

Malinen, S., Hlushchuk, Y., and Hari, R. (2007). Towards natural stimulation in fMRI – issues of data analysis. *NeuroImage*, 35(1):131–139.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326.

Song, L., Smola, A., Borgwardt, K., and Gretton, A. (2008). Colored maximum variance unfolding. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems 20*, pages 1385–1392. MIT Press, Cambridge, MA.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323.

TIMIT (1998). TIMIT. CD-ROM prototype version of the DARPA TIMIT acoustic-phonetic speech database.

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

van der Maaten, L., Postma, E., and van der Herik, J. (2009). Dimensionality reduction: A comparative review. Technical report TiCC-TR 2009-005 (35 pages), Tilburg centre for Creative Computing, Tilburg University.

Venna, J. and Kaski, S. (2006). Local multidimensional scaling. *Neural Networks*, 19:889–99.

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

Weinberger, K., Packer, B., and Saul, L. (2005). Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. In Cowell, R. G. and Ghahramani, Z., editors, *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS 2005)*, pages 381–388. Society for Artificial Intelligence and Statistics. (Available electronically at http://www.gatsby.ucl.ac.uk/aistats/).

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

Ylipaavalniemi, J., Savia, E., Malinen, S., Hari, R., Vigário, R., and Kaski, S. (2009). Dependencies between stimuli and spatially independent fMRI sources: Towards brain correlates of natural stimuli. *Neuroimage*, 48:176–185.

Zhang, Z. and Wang, J. (2007). MLLE: Modified locally linear embedding using multiple weights. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 1593–1600. MIT Press, Cambridge, MA.