附件1

Visualizing Data using t-SNE

Laurens van der Maaten L.VANDERMAATEN@MICC.UNIMAAS.NL MICC-IKAT

Maastricht University

P.O. Box 616, 6200 MD Maastricht, The Netherlands

Geoffrey Hinton HINTON@CS.TORONTO.EDU Department of Computer Science

University of Toronto

6 King’s College Road, M5S 3G4 Toronto, ON, Canada

Abstract

We present a new technique called “t-SNE” that visualizes high-dimensional data by giving each datapoint a location in a two or three-dimensional map. The technique is a variation of Stochastic Neighbor Embedding (Hinton and Roweis, 2002) that is much easier to optimize, and produces signifificantly better visualizations by reducing the tendency to crowd points together in the center of the map. t-SNE is better than existing techniques at creating a single map that reveals structure at many different scales. This is particularly important for high-dimensional data that lie on several different, but related, low-dimensional manifolds, such as images of objects from multiple classes seen from multiple viewpoints. For visualizing the structure of very large datasets, we show how t-SNE can use random walks on neighborhood graphs to allow the implicit structure of all of the data to inflfluence the way in which a subset of the data is displayed. We illustrate the performance of t-SNE on a wide variety of datasets and compare it with many other non-parametric visualization techniques, including Sammon mapping, Isomap, and Locally Linear Embedding. The visualizations produced by t-SNE are signifificantly better than those produced by the other techniques on almost all of the datasets.

1. Introduction

Visualization of high-dimensional data is an important problem in many different domains, and deals with data of widely varying dimensionality. Cell nuclei that are relevant to breast cancer, for example, are described by approximately 30 variables (Street et al., 1993), whereas the pixel intensity vectors used to represent images or the word-count vectors used to represent documents typically have thousands of dimensions. Over the last few decades, a variety of techniques for the visualization of such high-dimensional data have been proposed, many of which are reviewed by Ferreira de Oliveira and Levkowitz (2003). Important techniques include iconographic displays such as Chernoff faces (Chernoff, 1973), pixel-based techniques (Keim, 2000), and techniques that represent the dimensions in the data as vertices in a graph (Battista et al., 1994). Most of these techniques simply provide tools to display more than two data dimensions, and leave the interpretation of the data to the human observer. This severely limits the applicability of these techniques to real-world datasets that contain thousands of high-dimensional datapoints.

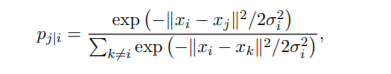
In contrast to the visualization techniques discussed above, dimensionality reduction methods con*{*y1, y2, ..., yn} that can be displayed in a scatterplot. In the paper, we refer to the low-dimensional data representation Y as a map, and to the low-dimensional representations yi of individual datapoints as map points. The aim of dimensionality reduction is to preserve as much of the signififi- cant structure of the high-dimensional data as possible in the low-dimensional map. Various techniques for this problem have been proposed that differ in the type of structure they preserve. Traditional dimensionality reduction techniques such as Principal Components Analysis (PCA; Hotelling (1933)) and classical multidimensional scaling (MDS; Torgerson (1952)) are linear techniques that focus on keeping the low-dimensional representations of dissimilar datapoints far apart. For highdimensional data that lies on or near a low-dimensional, non-linear manifold it is usually more important to keep the low-dimensional representations of very similar datapoints close together, which is typically not possible with a linear mapping. A large number of nonlinear dimensionality reduction techniques that aim to preserve the local structure of data have been proposed, many of which are reviewed by Lee and Verleysen (2007). In particular, we mention the following seven techniques: (1) Sammon mapping (Sammon, 1969), (2) curvilinear components analysis (CCA; Demartines and Herault (1997)), (3) Stochastic Neighbor Embedding (SNE; Hinton and Roweis (2002)), (4) Isomap (Tenenbaum et al., 2000), (5) Maximum Variance Unfolding (MVU; Weinberger et al. (2004)), (6) Locally Linear Embedding (LLE; Roweis and Saul (2000)), and (7) Laplacian Eigenmaps (Belkin and Niyogi, 2002). Despite the strong performance of these techniques on artifificial datasets, they are often not very successful at visualizing real, high-dimensional data. In particular, most of the techniques are not capable of retaining both the local and the global structure of the data in a single map. For instance, a recent study reveals that even a semi-supervised variant of MVU is not capable of separating handwritten digits into their natural clusters (Song et al., 2007).

In this paper, we describe a way of converting a high-dimensional dataset into a matrix of pairwise similarities and we introduce a new technique, called “t-SNE”, for visualizing the resulting similarity data. t-SNE is capable of capturing much of the local structure of the high-dimensional data very well, while also revealing global structure such as the presence of clusters at several scales. We illustrate the performance of t-SNE by comparing it to the seven dimensionality reduction techniques mentioned above on fifive datasets from a variety of domains. Because of space limitations, most of the (7 + 1) × 5 = 40 maps are presented in the supplemental material, but the maps that we present in the paper are suffificient to demonstrate the superiority of t-SNE. The outline of the paper is as follows. In Section 2, we outline SNE as presented by Hinton and Roweis (2002), which forms the basis for t-SNE. In Section 3, we present t-SNE, which has two important differences from SNE. In Section 4, we describe the experimental setup and the results of our experiments. Subsequently, Section 5 shows how t-SNE can be modifified to visualize real-world datasets that contain many more than 10, 000 datapoints. The results of our experiments are discussed in more detail in Section 6. Our conclusions and suggestions for future work are presented in Section 7.

2. Stochastic Neighbor Embedding

Stochastic Neighbor Embedding (SNE) starts by converting the high-dimensional Euclidean distances between datapoints into conditional probabilities that represent similarities1 . The similarity of datapoint xj to datapoint xi is the conditional probability, pj|i , that xi would pick xj as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at xi . For nearby datapoints, pj|i is relatively high, whereas for widely separated datapoints, pj|i will be almost infifinitesimal (for reasonable values of the variance of the Gaussian, σi). Mathematically, the conditional probability pj|i is given by

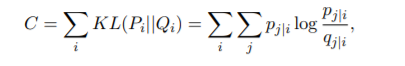
coding redundancy, and may not be efficient in combating bursty losses. Our studies show that MDMC and RPS without FEC are sufficient for the loss environment considered.



where σi is the variance of the Gaussian that is centered on datapoint xi . The method for determining the value of σi is presented later in this section. Because we are only interested in modeling pairwise similarities, we set the value of pi|i to zero. For the low-dimensional counterparts yi and yj of the high-dimensional datapoints xi and xj , it is possible to compute a similar conditional probability, which we denote by qj|i . We set2 the variance of the Gaussian that is employed in the computation of the conditional probabilities qj|i to √12 . Hence, we model the similarity of map point yj to map point yi by



Again, since we are only interested in modeling pairwise similarities, we set qi|i = 0. If the map points yi and yj correctly model the similarity between the high-dimensional datapoints xi and xj , the conditional probabilities pj|i and qj|i will be equal. Motivated by this observation, SNE aims to fifind a low-dimensional data representation that minimizes the mismatch between pj|i and qj|i . A natural measure of the faithfulness with which qj|i models pj|i is the Kullback-Leibler divergence (which is in this case equal to the cross-entropy up to an additive constant). SNE minimizes the sum of Kullback-Leibler divergences over all datapoints using a gradient descent method. The cost function C is given by

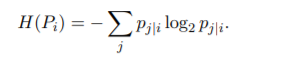


in which Pi represents the conditional probability distribution over all other datapoints given datapoint xi , and Qi represents the conditional probability distribution over all other map points given map point yi . Because the Kullback-Leibler divergence is not symmetric, different types of error in the pairwise distances in the low-dimensional map are not weighted equally. In particular, there is a large cost for using widely separated map points to represent nearby datapoints (i.e., for using a small qj|i to model a large pj|i ), but there is only a small cost for using nearby map points to represent widely separated datapoints. This small cost comes from wasting some of the probability mass in the relevant Q distributions. In other words, the SNE cost function focuses on retaining the local structure of the data in the map (for reasonable values of the variance of the Gaussian in the high-dimensional space, σi).

The remaining parameter to be selected is the variance σi of the Gaussian that is centered over each high-dimensional datapoint, xi . It is not likely that there is a single value of σi that is optimal for all datapoints in the dataset because the density of the data is likely to vary. In dense regions, a smaller value of σi is usually more appropriate than in sparser regions. Any particular value of σi induces a probability distribution, Pi , over all of the other datapoints. This distribution has an entropy which increases as σi increases. SNE performs a binary search for the value of σi that produces a Pi with a fifixed perplexity that is specifified by the user3 . The perplexity is defifined as



where H(Pi) is the Shannon entropy of Pi measured in bits



The perplexity can be interpreted as a smooth measure of the effective number of neighbors. The performance of SNE is fairly robust to changes in the perplexity, and typical values are between 5 and 50. The minimization of the cost function in Equation 3 is performed using a gradient descent method. The gradient has a surprisingly simple form



Physically, the gradient may be interpreted as the resultant force created by a set of springs between the map point yi and all other map points yj . All springs exert a force along the direction (yi − yj ). The spring between yi and yj repels or attracts the map points depending on whether the distance between the two in the map is too small or too large to represent the similarities between the two high-dimensional datapoints. The force exerted by the spring between yi and yj is proportional to its length, and also proportional to its stiffness, which is the mismatch (pj|i − qj|i + pi|j − qi|j ) between the pairwise similarities of the data points and the map points. The gradient descent is initialized by sampling map points randomly from an isotropic Gaussian with small variance that is centered around the origin. In order to speed up the optimization and to avoid poor local minima, a relatively large momentum term is added to the gradient. In other words, the current gradient is added to an exponentially decaying sum of previous gradients in order to determine the changes in the coordinates of the map points at each iteration of the gradient search. Mathematically, the gradient update with a momentum term is given by



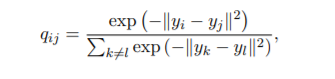
where Y (t) indicates the solution at iteration t, η indicates the learning rate, and α(t) represents the momentum at iteration t. In addition, in the early stages of the optimization, Gaussian noise is added to the map points after each iteration. Gradually reducing the variance of this noise performs a type of simulated annealing that helps the optimization to escape from poor local minima in the cost function. If the variance of the noise changes very slowly at the critical point at which the global structure of the map starts to form, SNE tends to find maps with a better global organization. Unfortunately, this requires sensible choices of the initial amount of Gaussian noise and the rate at which it decays. Moreover, these choices interact with the amount of momentum and the step size that are employed in the gradient descent. It is therefore common to run the optimization several times on a dataset to find appropriate values for the parameters4 . In this respect, SNE is inferior to methods that allow convex optimization and it would be useful to find an optimization method that gives good results without requiring the extra computation time and parameter choices introduced by the simulated annealing.

**3. t-Distributed Stochastic Neighbor Embedding**

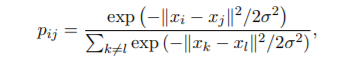
Section 2 discussed SNE as it was presented by Hinton and Roweis (2002). Although SNE constructs reasonably good visualizations, it is hampered by a cost function that is difficult to optimize and by a problem we refer to as the “crowding problem”. In this section, we present a new technique called “t-Distributed Stochastic Neighbor Embedding” or “t-SNE” that aims to alleviate these problems. The cost function used by t-SNE differs from the one used by SNE in two ways: (1) it uses a symmetrized version of the SNE cost function with simpler gradients that was briefly introduced by Cook et al. (2007) and (2) it uses a Student-t distribution rather than a Gaussian to compute the similarity between two points in the low-dimensional space. t-SNE employs a heavy-tailed distribution in the low-dimensional space to alleviate both the crowding problem and the optimization problems of SNE. In this section, we first discuss the symmetric version of SNE (subsection 3.1). Subsequently, we discuss the crowding problem (subsection 3.2), and the use of heavy-tailed distributions to address this problem (subsection 3.3). We conclude the section by describing our approach to the optimization of the t-SNE cost function (subsection 3.4). 3.1 Symmetric SNE As an alternative to minimizing the sum of the Kullback-Leibler divergences between the conditional probabilities pj|i and qj|i , it is also possible to minimize a single Kullback-Leibler divergence between a joint probability distribution, P, in the high-dimensional space and a joint probability distribution, Q, in the low-dimensional space:



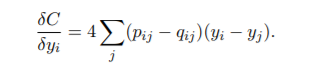
where again, we set pii and qii to zero. We refer to this type of SNE as symmetric SNE, because it has the property that pij = pji and qij = qji for ∀i, j. In symmetric SNE, the pairwise similarities in the low-dimensional map qij are given by



The obvious way to define the pairwise similarities in the high-dimensional space pij i



but this causes problems when a high-dimensional datapoint xi is an outlier (i.e., all pairwise distances kxi − xjk 2 are large for xi). For such an outlier, the values of pij are extremely small for all j, so the location of its low-dimensional map point yi has very little effect on the cost function. As a result, the position of the map point is not well determined by the positions of the other map points. We circumvent this problem by defining the joint probabilities pij in the high-dimensional space to be the symmetrized conditional probabilities, i.e., we set pij = pj|i+pi|j 2n P . This ensures that j pij > 1 2n for all datapoints xi , as a result of which each datapoint xi makes a significant contribution to the cost function. In the low-dimensional space, symmetric SNE simply uses Equation 9. The main advantage of the symmetric version of SNE is the simpler form of its gradient, which is faster to compute. The gradient of symmetric SNE is fairly similar to that of asymmetric SNE, and is given



In preliminary experiments, we observed that symmetric SNE seems to produce maps that are just as good as asymmetric SNE, and sometimes even a little better

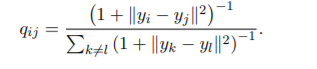
**3.2 The crowding problem**

Consider a set of datapoints that lie on a two-dimensional curved manifold which is approximately linear on a small scale, and which is embedded within a higher-dimensional space. It is possible to model the small pairwise distances between datapoints fairly well in a two-dimensional map, which is often illustrated on toy examples such as the “Swiss roll” dataset. Now suppose that the manifold has ten intrinsic dimensions5 and is embedded within a space of much higher dimensionality. There are several reasons why the pairwise distances in a two-dimensional map cannot faithfully model distances between points on the ten-dimensional manifold. For instance, in ten dimensions, it is possible to have 11 datapoints that are mutually equidistant and there is no way to model this faithfully in a two-dimensional map. A related problem is the very different distribution of pairwise distances in the two spaces. The volume of a sphere centered on datapoint i scales as r m, where r is the radius and m the dimensionality of the sphere. So if the datapoints are approximately uniformly distributed in the region around i on the ten-dimensional manifold, and we try to model the distances from i to the other datapoints in the two-dimensional map, we get the following “crowding problem”: the area of the two-dimensional map that is available to accommodate moderately distant datapoints will not be nearly large enough compared with the area available to accommodate nearby datapoints. Hence, if we want to model the small distances accurately in the map, most of the point

that are at a moderate distance from datapoint i will have to be placed much too far away in the two-dimensional map. In SNE, the spring connecting datapoint i to each of these too-distant map points will thus exert a very small attractive force. Although these attractive forces are very small, the very large number of such forces crushes together the points in the center of the map, which prevents gaps from forming between the natural clusters. Note that the crowding problem is not specific to SNE, but that it also occurs in other local techniques for multidimensional scaling such as Sammon mapping. An attempt to address the crowding problem by adding a slight repulsion to all springs was presented by Cook et al. (2007). The slight repulsion is created by introducing a uniform background model with a small mixing proportion, ρ. So however far apart two map points are, qij can never fall below 2ρ n(n−1) (because the uniform background distribution is over n(n − 1)/2 pairs). As a result, for datapoints that are far apart in the high-dimensional space, qij will always be larger than pij , leading to a slight repulsion. This technique is called UNI-SNE and although it usually outperforms standard SNE, the optimization of the UNI-SNE cost function is tedious. The best optimization method known is to start by setting the background mixing proportion to zero (i.e., by performing standard SNE). Once the SNE cost function has been optimized using simulated annealing, the background mixing proportion can be increased to allow some gaps to form between natural clusters as shown by Cook et al. (2007). Optimizing the UNI-SNE cost function directly does not work because two map points that are far apart will get almost all of their qij from the uniform background. So even if their pij is large, there will be no attractive force between them, because a small change in their separation will have a vanishingly small proportional effect on qij . This means that if two parts of a cluster get separated early on in the optimization, there is no force to pull them back together.

**3.3 Mismatched tails can compensate for mismatched dimensionalities**

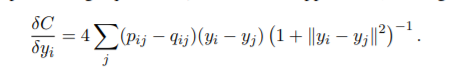
Since symmetric SNE is actually matching the joint probabilities of pairs of datapoints in the highdimensional and the low-dimensional spaces rather than their distances, we have a natural way of alleviating the crowding problem that works as follows. In the high-dimensional space, we convert distances into probabilities using a Gaussian distribution. In the low-dimensional map, we can use a probability distribution that has much heavier tails than a Gaussian to convert distances into probabilities. This allows a moderate distance in the high-dimensional space to be faithfully modeled by a much larger distance in the map and, as a result, it eliminates the unwanted attractive forces between map points that represent moderately dissimilar datapoints. In t-SNE, we employ a Student t-distribution with one degree of freedom (which is the same as a Cauchy distribution) as the heavy-tailed distribution in the low-dimensional map. Using this distribution, the joint probabilities qij are defined as



We use a Student t-distribution with a single degree of freedom, because it has the particularly nice property that 1 + kyiyjk21 approaches an inverse square law for large pairwise distances kyi

yjk in the low-dimensional map. This makes the map’s representation of joint probabilities (almost) invariant to changes in the scale of the map for map points that are far apart. It also means that large clusters of points that are far apart interact in just the same way as individual points, so the optimization operates in the same way at all but the fifinest scales. A theoretical justifification for ouselection of the Student t-distribution is that it is closely related to the Gaussian distribution, as the Student t-distribution is an infifinite mixture of Gaussians. A computationally convenient property is that it is much faster to evaluate the density of a point under a Student t-distribution than under a Gaussian because it does not involve an exponential, even though the Student t-distribution is equivalent to an infifinite mixture of Gaussians with different variances.

The gradient of the Kullback-Leibler divergence between P and the Student-t based joint probability distribution Q (computed using Equation 12) is derived in Appendix A, and is given by



In Figure 1(a) to 1(c), we show the gradients between two low-dimensional datapoints yi and yj as a function of their pairwise Euclidean distances in the high-dimensional and the low-dimensional space (i.e., as a function of kxi xjk and kyi yjk) for the symmetric versions of SNE, UNISNE, and t-SNE. In the fifigures, positive values of the gradient represent an attraction between the low-dimensional datapoints yi and yj , whereas negative values represent a repulsion between the two datapoints. From the fifigures, we observe two main advantages of the t-SNE gradient over the gradients of SNE and UNI-SNE.

First, the t-SNE gradient strongly repels dissimilar datapoints that are modeled by a small pairwise distance in the low-dimensional representation. SNE has such a repulsion as well, but its effect is minimal compared to the strong attractions elsewhere in the gradient (the largest attraction in our graphical representation of the gradient is approximately 19, whereas the largest repulsion is approximately 1). In UNI-SNE, the amount of repulsion between dissimilar datapoints is slightly larger, however, this repulsion is only strong when the pairwise distance between the points in the lowdimensional representation is already large (which is often not the case, since the low-dimensional representation is initialized by sampling from a Gaussian with a very small variance that is centered around the origin). Second, although t-SNE introduces strong repulsions between dissimilar datapoints that are modeled by small pairwise distances, these repulsions do not go to infifinity. In this respect, t-SNE differs from UNI-SNE, in which the strength of the repulsion between very dissimilar datapoints is proportional to their pairwise distance in the low-dimensional map, which may cause dissimilar datapoints

to move much too far away from each other. Taken together, t-SNE puts emphasis on (1) modeling dissimilar datapoints by means of large pairwise distances, and (2) modeling similar datapoints by means of small pairwise distances. Moreover, as a result of these characteristics of the t-SNE cost function (and as a result of the approximate scale invariance of the Student t-distribution), the optimization of the t-SNE cost function is much easier than the optimization of the cost functions of SNE and UNI-SNE. Specififically, t-SNE introduces long-range forces in the low-dimensional map that can pull back together two (clusters of) similar points that get separated early on in the optimization. SNE and UNI-SNE do not have such longrange forces, as a result of which SNE and UNI-SNE need to use simulated annealing to obtain reasonable solutions. Instead, the long-range forces in t-SNE facilitate the identifification of good local optima without resorting to simulated annealing

**References**

W.E. Arnoldi. The principle of minimized iteration in the solution of the matrix eigenvalue problem. Quarterly of Applied Mathematics, 9:17–25, 1951.

G.D. Battista, P. Eades, R. Tamassia, and I.G. Tollis. Annotated bibliography on graph drawing. Computational Geometry: Theory and Applications, 4:235–282, 1994.

M. Belkin and P. Niyogi. Laplacian Eigenmaps and spectral techniques for embedding and clustering. In Advances in Neural Information Processing Systems, volume 14, pages 585–591, Cambridge, MA, USA, 2002. The MIT Press.

Y. Bengio. Learning deep architectures for AI. Technical Report 1312, Universite de Montr ´ eal, ´ 2007.

N. Biggs. Algebraic graph theory. In Cambridge Tracts in Mathematics, volume 67. Cambridge University Press, 1974.

H. Chernoff. The use of faces to represent points in k-dimensional space graphically. Journal of the American Statistical Association, 68:361–368, 1973.

J.A. Cook, I. Sutskever, A. Mnih, and G.E. Hinton. Visualizing similarity data with a mixture of maps. In Proceedings of the 11th International Conference on Artificial Intelligence and Statistics, volume 2, pages 67–74, 2007.

M.C. Ferreira de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: A survey. IEEE Transactions on Visualization and Computer Graphics, 9(3):378–394, 2003.

V. de Silva and J.B. Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. In Advances in Neural Information Processing Systems, volume 15, pages 721–728, Cambridge, MA, USA, 2003. The MIT Press.

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

P. Doyle and L. Snell. Random walks and electric networks. In Carus mathematical monographs, volume 22. Mathematical Association of America, 1984.

D.R. Fokkema, G.L.G. Sleijpen, and H.A. van der Vorst. Jacobi–Davidson style QR and QZ algorithms for the reduction of matrix pencils. SIAM Journal on Scientific Computing, 20(1):94–125, 1999.

L. Grady. Random walks for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(11):1768–1783, 2006.

G.E. Hinton and S.T. Roweis. Stochastic Neighbor Embedding. In Advances in Neural Information Processing Systems, volume 15, pages 833–840, Cambridge, MA, USA, 2002. The MIT Press.

G.E. Hinton and R.R. Salakhutdinov. Reducing the dimensionality of data with neural networks. Science, 313(5786):504–507, 2006.

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

R.A. Jacobs. Increased rates of convergence through learning rate adaptation. Neural Networks, 1: 295–307, 1988.

S. Kakutani. Markov processes and the Dirichlet problem. Proceedings of the Japan Academy, 21: 227–233, 1945.

D.A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. IEEE Transactions on Visualization and Computer Graphics, 6(1):59–78, 2000.

S. Lafon and A.B. Lee. Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(9):1393–1403, 2006.

附件2

使用t-SNE可视化数据

Laurens van der Maaten，马斯特里赫特大学

Geoffrey Hinton, 多伦多大学

**摘要**

我们提出了一种称为“ t-SNE”的新技术，通过给每个数据指向二维或三维地图中的位置。该技术是随机的邻居嵌入（Hinton和Roweis，2002）更容易优化和生成通过减少将点拥挤在中心的趋势，显着改善可视化效果地图。在创建揭示结构的单个地图方面，t-SNE比现有技术更好在许多不同的规模。这对于位于多个位置的高维数据尤为要不同但相关的低维流形，例如来自多个类别的对象的图像从多个角度来看。为了可视化非常大的数据集的结构，我们展示了如何t-SNE可以在邻域图上使用随机游走，以允许所有数据以影响数据子集的显示方式。我们说明性能-SNE在各种数据集上的比较，并将其与许多其他非参数可视化进行比较技术，包括Sammon映射，Isomap和局部线性嵌入。 t-SNE产生的可视化效果明显优于其他技术。几乎所有的数据集。

关键字：可视化，降维，流形学习，嵌入算法，多维缩放

**简介**

高维数据的可视化是许多不同领域中的重要问题，并且处理维度变化很大的数据。与乳腺癌有关的细胞核，例如，用大约30个变量来描述（Street等，1993），而像素用于表示图像的强度矢量或用于表示文档的字数矢量通常具有数千个尺寸。在过去的几十年中，

已经提出了这种高维数据的可视化，其中许多已由Ferreira de Oliveira和Levkowitz（2003）。重要技术包括图像显示例如Chernoff的面孔（切尔诺夫，1973），基于像素的技术（Keim，2000）以及将数据的维表示为图形中的顶点（Battista等，1994）。这些大多数技术只是提供工具来显示两个以上的数据维度，并将数据的解释留给观察者，这严重限制了这些技术对包含数千个高维数据点的真实世界数据集的适用性。

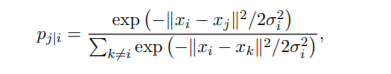
与上面讨论的可视化技术相比，降维方法将高维数据集X = {x1，x2，...，xn}转换为二维或三维数据Y = {y1，y2，... ，yn}可以显示在散点图中。在本文中，我们指的是低维数据表示Y作为地图，而将各个数据点的低维表示yi作为地图点。降维的目的是保留尽可能多的意义。在低维地图中，高维数据的结构可能不可行。已经提出了解决该问题的各种技术，它们保留的结构类型不同。传统降维技术，例如主成分分析（PCA； Hotelling）

（1933））和经典多维比例缩放（MDS; Torgerson（1952））是线性技术，着重于使相异数据点的低维表示相距较远。对于位于低维非线性流形上或附近的高维数据，通常会更

重要的是要使非常相似的数据点的低维表示形式紧密相连，使用线性映射通常是不可能的。

1. **随机邻居嵌入**

随机邻居嵌入（SNE）首先将数据点之间的高维欧氏距离转换为表示相似度的条件概率1。 相似点数据点xj到数据点xi是条件概率pj | i，如果在以高斯为中心的情况下按与概率密度成比例的比例选择邻居，则xi将选择xj作为其邻居。xi。 对于附近的数据点，pj | i相对较高，而对于广泛分离的数据点，pj | i将几乎是无限的（对于高斯方差σi的合理值）。 数学上条件概率pj | i是（谁）给的

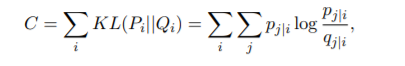


其中σi是以数据点xi为中心的高斯方差。 确定方法σi的值在本节稍后介绍。 因为我们只对成对建模感兴趣相似之处，我们设置pi | i的值归零。 对于低维的yi和yj高维数点xi和xj，可以计算出类似的条件概率，我们用qj | i表示。 我们set2计算中采用的高斯方差条件概率qj | i至√12。 因此，我们对映射点yj与地图的相似性进行建模



同样，由于我们只对建模成对相似性感兴趣，因此将qi | i设置为0。如果地图点yi和yj正确地建模了高维数据点之间的相似度xi和xj，条件概率pj | i和qj | i相等。 出于这一观察的动机，SNE的目标是找到一种低维数据表示形式，以最小化pj | i之间的不匹配

和qj | i。 qj | i对pj | i建模的真实性的自然度量是Kullback-Leibler散度（在这种情况下等于交叉熵，直到加法常数）。 SNE使用梯度下降法将所有数据点的Kullback-Leibler散度之和最小化。成本函数C由下式给出



其中Pi表示在给定数据点xi的情况下所有其他数据点的条件概率分布，而Qi表示在给定的所有其他地图点上的条件概率分布易点图。 由于Kullback-Leibler散度不是对称的，因此会产生不同类型的误差在低维图中的成对距离中的权重不均等。

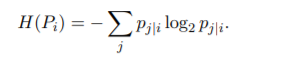
特别是那里使用广泛分离的地图点表示附近的数据点（例如，使用较小的qj | i建模大pj | i），但使用附近的地图点到代表广泛分离的数据点。 这么小的成本来自浪费一些可能性

相关Q分布中的质量。 换句话说，SNE成本函数侧重于保留地图中数据的局部结构（用于获得高斯方差的合理值）高维空间σi）。其余要选择的参数是高斯方差σi，该方差以每个中心为中心高维数据点，xi。 σi不可能只有一个值最适合所有人数据集中的数据点，因为数据的密度可能会变化。 在人口稠密的地区，较小的σi值通常比稀疏地区更合适。 σi的任何特定值诱发概率分布，Pi，覆盖所有其他数据点。 该分布具有一个熵随着σi增加

增加。 SNE对σi值执行二进制搜索产生一个Pi用户指定的固定困惑。 困惑被定义为



其中H（Pi）是Pi的Shannon熵，以位为单位



困惑可以解释为邻居有效数量的平滑度量。 的SNE的性能对于复杂性的变化相当强大，典型值在5之间和50。使用梯度下降法执行公式3中成本函数的最小化。渐变具有令人惊讶的简单形式



从物理上讲，坡度可解释为由一组弹簧之间产生的合力地图点yi和所有其他地图点yj。所有弹簧都沿方向（yi yj）施加力。yi和yj之间的弹簧取决于距离是否排斥或吸引地图点

地图中两者之间的距离太小或太大，无法代表两者之间的相似性高维数据点。弹簧在y​​i和yj之间施加的力与它的长度，也与其刚度成正比，即不匹配（pj | i qj | i + pi | j qi | j）数据点和映射点的成对相似性之间的关系。通过从各向同性高斯随机采样图点来初始化梯度下降以原点为中心的小方差。为了加快优化和为了避免较差的局部最小值，可以在梯度中添加相对较大的动量项。换一种说法，当前梯度被添加到先前梯度的指数衰减总和中，以便确定在每次梯度搜索迭代时地图点坐标的变化。数学上，动量项的梯度更新为



1. **t分布随机邻居嵌入**

第二节讨论了Hinton和Roweis（2002）提出的SNE。尽管SNE可以构建合理的可视化效果，但是它受到难以优化的成本函数的阻碍被一个问题称为“拥挤问题”。在本节中，我们介绍一种新技术旨在减轻这些问题的称为“ t分布随机邻居嵌入”或“ t-SNE”。 t-SNE使用的成本函数与SNE使用的成本函数有两个方面的不同：（1）它使用简要介绍了SNE成本函数的对称版本以及更简单的渐变由Cook等人撰写。 （2007）和（2），它使用Student-t分布而不是高斯分布来计算低维空间中两点之间的相似性。 t-SNE在低维空间中采用了重尾分布，以缓解拥挤问题和优化问题SNE的问题。在本节中，我们首先讨论SNE的对称版本（3.1节）。随后，我们讨论拥挤问题（3.2小节），以及使用重尾分布来解决这个问题（第3.3小节）。我们通过描述优化t-SNE成本函数的方法（第3.4节）来结束本节。

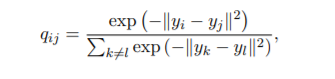
**3.1对称SNE**

作为最小化条件概率pj | i和qj | i之间的Kullback-Leibler散度之和的替代方法，也有可能使单个Kullback-Leibler差异最小化高维空间中的联合概率分布P与联合概率之间的关系低维空间中的分布Q：

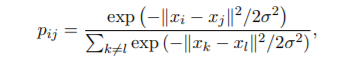


同样，我们将pii和qii设置为零。 我们将这种类型的SNE称为对称SNE，因为它

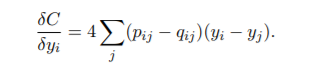
对于∀i，j具有pij = pji和qij = qji的性质。 在对称SNE中，低维映射qij中的成对相似性由下式给出



在高维空间pij中定义成对相似性的明显方法是



但这会在高维数据点xi产生问题是一个异常值（即所有成对距离kxi xjk2对于xi来说很大）。对于这样的离群值，pij的值对于全部为j，因此其低维地图点yi的位置对成本函数的影响很小。结果，地图点的位置不能很好地由其他地图的位置确定点。我们通过定义高维联合概率pij来规避此问题空间作为对称条件概率，即，我们设置pij = pj | i + pi | j 2n P。这样可以确保j pij> 12n对于所有数据点xi结果，每个数据点xi对成本函数有重大贡献。在低维空间中，对称SNE仅使用公式9。SNE的对称版本的主要优点是其渐变的简单形式，即计算速度更快。对称SNE的梯度与非对称SNE的梯度非常相似，并且是（谁）给的



在初步实验中，我们观察到对称SNE似乎会产生仅和不对称SNE一样好，有时甚至更好。

**3.2拥挤问题**

考虑位于二维曲面流形上的一组数据点，其近似为线性小规模，并嵌入到高维空间中。有可能在二维图中相当好地建模数据点之间的小成对距离通常在玩具示例（例如“瑞士卷”数据集）上进行说明。现在假设玛尼褶皱有十个内在维度5并嵌入到更高维度的空间中。

二维映射中的成对距离无法如实地存在的原因有几个模拟十维流形上各点之间的距离。例如，在十个维度中可能有11个相互等距的数据点，无法对此建模忠实于二维地图。一个相关的问题是成对的分布非常不同两个空间中的距离。以数据点i为中心的球体的体积缩放为rm，其中r为球形的半径和维数。因此，如果数据点大致均匀分布在10维流形上i周围的区域中，我们尝试对i到二维地图中其他数据点的距离进行建模，得到以下“拥挤问题”：可用于容纳中等距离的二维地图区域与可容纳附近的区域相比，数据点的大小将不足以容纳足够的空间数据点。因此，如果我们要在地图中准确地为小距离建模，则与数据点i距离适中的大多数点都必须放置在距离数据点太远的位置二维地图。在SNE中，弹簧将数据点i连接到这些太远的地图中的每一个因此，积分将施加很小的吸引力。尽管这些吸引力很小，

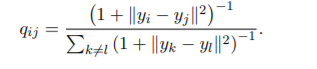
如此大量的力量将地图中心的点压在一起，防止自然簇之间形成间隙。注意，拥挤问题不是特定于SNE，但它也发生在其他用于多维缩放的本地技术中作为Sammon映射。

提出了通过向所有弹簧略加排斥来解决拥挤问题的尝试由Cook等人撰写。 （2007）。通过引入统一的背景模型可产生轻微的排斥混合比例ρ很小。因此，无论两个地图点相距多远，qij都永远不会低于2ρn（nn 1）（因为均匀的背景分布超过n（n 1）/ 2对）。结果，对于在高维空间中相距较远的数据，qij将始终大于pij，从而略有排斥。这种技术称为UNI-SNE，尽管它通常胜过标准SNE，但UNI-SNE成本函数的优化却很繁琐。最佳的优化方法已知的方法是先将背景混合比例设置为零（即通过执行标准SNE）。使用模拟退火优化SNE成本函数后，背景如图所示，可以增加混合比例以允许自然簇之间形成一些间隙由Cook等人撰写。 （2007）。直接优化UNI-SNE成本函数不起作用，因为有两个相距较远的地图点将从统一背景中获取几乎所有的qij。所以即使如果它们的pij大，则它们之间就不会有吸引力，因为它们的变化很小分离对qij的比例影响将消失得很小。这意味着，如果集群在优化的早期就被分离了，没有力量将它们拉回一起。

**3.3不匹配的尾巴可以补偿不匹配的尺寸**

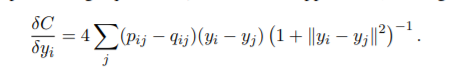
由于对称SNE实际上匹配高维和低维空间中的数据点对的联合概率，而不是它们之间的距离，因此我们有一种自然的方法缓解如下所述的拥挤问题。 在高维空间中，我们使用高斯分布将距离转换为概率。 在低维地图中，我们可以使用尾部比高斯重的概率分布来转换距离转化为概率。 这样可以在高维空间中忠实地保持距离通过地图中更大的距离进行建模，从而消除了不必要的吸引力表示适度不同数据点的地图点之间的力。

在t-SNE中，我们采用具有一个自由度的Student t分布（与柯西分布）作为低维图中的重尾分布。 使用这个分布，联合概率qij定义为



我们使用具有单一自由度的Student t分布，因为它具有特别好的1 + kyi yjk2 1的属性接近大成对距离的平方反比定律kyi yjk在低维图中。这使得地图可以表示联合概率对于相距较远的地图点，（几乎）不变。这也意味着彼此相距遥远的大型点簇以与单个点相同的方式进行交互，因此除了最佳规模外，优化工作的方式完全相同。我们选择学生t分布的理论依据是，它与高斯分布密切相关，因为学生t分布是高斯的无限混合。计算方便的属性是在学生t分布下评估点的密度要比在学生t分布下评估点的密度快得多高斯，因为它不涉及指数，即使学生t分布为等效于具有不同方差的高斯的无限混合。

P和基于Student-t的联合概率之间的Kullback-Leibler散度的梯度分布Q（使用公式12计算）在附录A中得出，由下式给出



**参考文献**

[1]W.E. Arnoldi. The principle of minimized iteration in the solution of the matrix eigenvalue problem. Quarterly of Applied Mathematics, 9:17–25, 1951.

[2]G.D. Battista, P. Eades, R. Tamassia, and I.G. Tollis. Annotated bibliography on graph drawing. Computational Geometry: Theory and Applications, 4:235–282, 1994.

[3]M. Belkin and P. Niyogi. Laplacian Eigenmaps and spectral techniques for embedding and clustering. In Advances in Neural Information Processing Systems, volume 14, pages 585–591, Cambridge, MA, USA, 2002. The MIT Press.

[3]Y. Bengio. Learning deep architectures for AI. Technical Report 1312, Universite de Montr ´ eal, ´ 2007.

[4]N. Biggs. Algebraic graph theory. In Cambridge Tracts in Mathematics, volume 67. Cambridge University Press, 1974.

[5]H. Chernoff. The use of faces to represent points in k-dimensional space graphically. Journal of the American Statistical Association, 68:361–368, 1973.

[6]J.A. Cook, I. Sutskever, A. Mnih, and G.E. Hinton. Visualizing similarity data with a mixture of maps. In Proceedings of the 11th International Conference on Artificial Intelligence and Statistics, volume 2, pages 67–74, 2007.

[7]M.C. Ferreira de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: A survey. IEEE Transactions on Visualization and Computer Graphics, 9(3):378–394, 2003.

[8]V. de Silva and J.B. Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. In Advances in Neural Information Processing Systems, volume 15, pages 721–728, Cambridge, MA, USA, 2003. The MIT Press.

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

[10]P. Doyle and L. Snell. Random walks and electric networks. In Carus mathematical monographs, volume 22. Mathematical Association of America, 1984.

[11]D.R. Fokkema, G.L.G. Sleijpen, and H.A. van der Vorst. Jacobi–Davidson style QR and QZ algorithms for the reduction of matrix pencils. SIAM Journal on Scientific Computing, 20(1):94–125, 1999.

[12]L. Grady. Random walks for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(11):1768–1783, 2006.

[13]G.E. Hinton and S.T. Roweis. Stochastic Neighbor Embedding. In Advances in Neural Information Processing Systems, volume 15, pages 833–840, Cambridge, MA, USA, 2002. The MIT Press.

[14]G.E. Hinton and R.R. Salakhutdinov. Reducing the dimensionality of data with neural networks. Science, 313(5786):504–507, 2006.

[15]H. Hotelling. Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, 24:417–441, 1933.

[16]R.A. Jacobs. Increased rates of convergence through learning rate adaptation. Neural Networks, 1: 295–307, 1988.

[17]S. Kakutani. Markov processes and the Dirichlet problem. Proceedings of the Japan Academy, 21: 227–233, 1945.

[18]D.A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. IEEE Transactions on Visualization and Computer Graphics, 6(1):59–78, 2000.

[19]S. Lafon and A.B. Lee. Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(9):1393–1403, 2006.