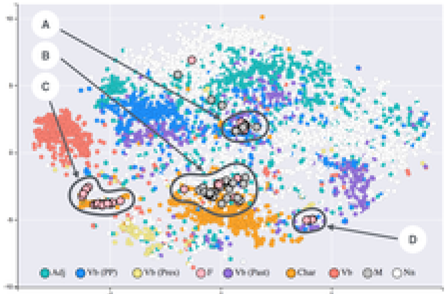
WIKIPEDIA

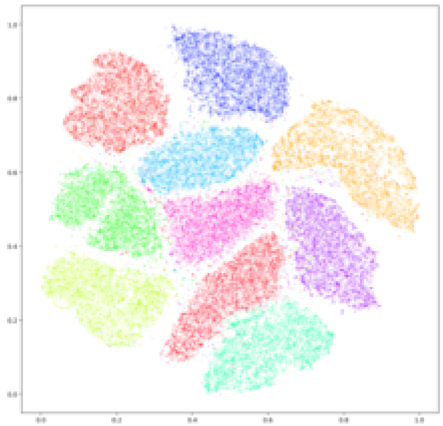
t-distributed stochastic neighbor embedding

t-distributed stochastic neighbor embedding (**t-SNE**) is a statistical method for visualizing high-dimensional data by giving each datapoint a location in a two or three-dimensional map. It is based on Stochastic Neighbor Embedding originally developed by Sam Roweis and Geoffrey Hinton,[1] where Laurens van der Maaten proposed the *t*-distributed variant.[2] It is a nonlinear dimensionality reduction technique well-suited for embedding high-dimensional data for visualization in a low-dimensional space of two or three dimensions. Specifically, it models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points with high probability.



T-SNE visualisation of word embeddings generated using 19th century literature

The t-SNE algorithm comprises two main stages. First, t-SNE constructs a probability distribution over pairs of high-dimensional objects in such a way that similar objects are assigned a higher probability while dissimilar points are assigned a lower probability. Second, t-SNE defines a similar probability distribution over the points in the low-dimensional map, and it minimizes the Kullback–Leibler divergence (KL divergence) between the two distributions with respect to the locations of the points in the map. While the original algorithm uses the Euclidean distance between objects as the base of its similarity metric, this can be changed as appropriate.


T-SNE embeddings of MNIST dataset

t-SNE has been used for visualization in a wide range of applications, including genomics, computer security research,[3] natural language processing, music analysis,[4] cancer research,[5] bioinformatics,[6] geological domain interpretation,[7][8][9] and biomedical signal processing.[10]

While t-SNE plots often seem to display clusters, the visual clusters can be influenced strongly by the chosen parameterization and therefore a good understanding of the parameters for t-SNE is necessary. Such "clusters" can be shown to even appear in non-clustered data,[11] and thus may be false findings. Interactive exploration may thus be necessary to choose parameters and validate results.[12][13] It has been demonstrated that t-SNE is often able to recover well-separated clusters, and with special parameter choices, approximates a simple form of spectral clustering.[14]

Contents

Details

Given a set of N high-dimensional objects $\mathbf{x}_1, \dots, \mathbf{x}_N$, t-SNE first computes probabilities p_{ij} that are proportional to the similarity of objects $\mathbf{x}_i$ and $\mathbf{x}_j$, as follows.

For $i \neq j$, define

$$p_{j|i} = \frac{\exp(-\lVert \mathbf{x}_i - \mathbf{x}_j \rVert^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\lVert \mathbf{x}_i - \mathbf{x}_k \rVert^2 / 2\sigma_i^2)}$$

and set $p_{i|i} = 0$. Note that $\sum_j p_{j|i} = 1$ for all i.

As Van der Maaten and Hinton explained: "The similarity of datapoint x_j to datapoint x_i is the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at x_i."[2]

Now define

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

and note that $p_{ij} = p_{ji}$, $p_{ii} = 0$, and $\sum_{i,j} p_{ij} = 1$.

The bandwidth of the Gaussian kernels σ_i is set in such a way that the perplexity of the conditional distribution equals a predefined perplexity using the bisection method. As a result, the bandwidth is adapted to the density of the data: smaller values of σ_i are used in denser parts of the data space.

Since the Gaussian kernel uses the Euclidean distance $\lVert x_i - x_j \rVert$, it is affected by the curse of dimensionality, and in high dimensional data when distances lose the ability to discriminate, the p_{ij} become too similar (asymptotically, they would converge to a constant). It has been proposed to adjust the distances with a power transform, based on the intrinsic dimension of each point, to alleviate this.[15]

t-SNE aims to learn a d-dimensional map $\mathbf{y}_1, \dots, \mathbf{y}_N$ (with $\mathbf{y}_i \in \mathbb{R}^d$ and d typically chosen as 2 or 3) that reflects the similarities p_{ij} as well as possible. To this end, it measures similarities q_{ij} between two points in the map $\mathbf{y}_i$ and $\mathbf{y}_j$, using a very similar approach. Specifically, for $i \neq j$, define q_{ij} as

$$q_{ij} = \frac{(1 + \lVert \mathbf{y}_i - \mathbf{y}_j \rVert^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \lVert \mathbf{y}_k - \mathbf{y}_l \rVert^2)^{-1}}$$

and set $q_{ii} = 0$. Herein a heavy-tailed Student t-distribution (with one-degree of freedom, which is the same as a Cauchy distribution) is used to measure similarities between low-dimensional points in order to allow dissimilar objects to be modeled far apart in the map.

The locations of the points $\mathbf{y}_i$ in the map are determined by minimizing the (non-symmetric) Kullback–Leibler divergence of the distribution P from the distribution Q, that is:

$$\mathrm{KL}\left(P \parallel Q\right) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

The minimization of the Kullback–Leibler divergence with respect to the points $\mathbf{y}_i$ is performed using gradient descent. The result of this optimization is a map that reflects the similarities between the high-dimensional inputs.

Software

- The R package Rtsne (https://CRAN.R-project.org/package=Rtsne) implements t-SNE in R.
- ELKI contains tSNE, also with Barnes-Hut approximation
- Scikit-learn, a popular machine learning toolkit in python implements t-SNE with both exact solutions and the Barnes-Hut approximation.
- Tensorboard, the visualization kit associated with TensorFlow, also implements t-SNE (online version (https://projector.tensorflow.org/))

References

1. Roweis, Sam; Hinton, Geoffrey (January 2002). *Stochastic neighbor embedding* (https://cs.nyu.edu/~roweis/papers/sne_final.pdf) (PDF). Neural Information Processing Systems.
2. van der Maaten, L.J.P.; Hinton, G.E. (Nov 2008). "Visualizing Data Using t-SNE" (http://jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf) (PDF). *Journal of Machine Learning Research*. **9**: 2579–2605.
3. Gashi, I.; Stankovic, V.; Leita, C.; Thonnard, O. (2009). "An Experimental Study of Diversity with Off-the-shelf AntiVirus Engines". *Proceedings of the IEEE International Symposium on Network Computing and Applications*: 4–11.
4. Hamel, P.; Eck, D. (2010). "Learning Features from Music Audio with Deep Belief Networks". *Proceedings of the International Society for Music Information Retrieval Conference*: 339–344.
5. Jamieson, A.R.; Giger, M.L.; Drukker, K.; Lui, H.; Yuan, Y.; Bhooshan, N. (2010). "Exploring Nonlinear Feature Space Dimension Reduction and Data Representation in Breast CADx with Laplacian Eigenmaps and t-SNE" (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2807447). *Medical Physics*. **37** (1): 339–351. doi:10.1118/1.3267037 (https://doi.org/10.1118%2F1.3267037). PMC 2807447 (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2807447). PMID 20175497 (https://pubmed.ncbi.nlm.nih.gov/20175497).
6. Wallach, I.; Liliean, R. (2009). "The Protein-Small-Molecule Database, A Non-Redundant Structural Resource for the Analysis of Protein-Ligand Binding" (https://doi.org/10.1093%2Fbioinformatics%2Fbtp035). *Bioinformatics*. **25** (5): 615–620. doi:10.1093/bioinformatics/btp035 (https://doi.org/10.1093%2Fbioinformatics%2Fbtp035). PMID 19153135 (https://pubmed.ncbi.nlm.nih.gov/19153135).
7. Balamurali, Mehala; Silversides, Katherine L.; Melkumyan, Arman (2019-04-01). "A comparison of t-SNE, SOM and SPADE for identifying material type domains in geological data" (https://www.sciencedirect.com/science/article/pii/S0098300418306010). *Computers & Geosciences*. **125**: 78–89. doi:10.1016/j.cageo.2019.01.011 (https://doi.org/10.1016%2Fj.cageo.2019.01.011). ISSN 0098-3004 (https://www.worldcat.org/issn/0098-3004). S2CID 67926902 (https://api.semanticscholar.org/CorpusID:67926902).

8. Balamurali, Mehala; Melkumyan, Arman (2016). Hirose, Akira; Ozawa, Seiichi; Doya, Kenji; Ikeda, Kazushi; Lee, Minho; Liu, Derong (eds.). "t-SNE Based Visualisation and Clustering of Geological Domain" (https://link.springer.com/chapter/10.1007/978-3-319-46681-1_67). *Neural Information Processing*. Lecture Notes in Computer Science. Cham: Springer International Publishing. **9950**: 565–572. doi:10.1007/978-3-319-46681-1_67 (https://doi.org/10.1007%2F978-3-319-46681-1_67). ISBN 978-3-319-46681-1.
9. Leung, Raymond; Balamurali, Mehala; Melkumyan, Arman (2021-01-01). "Sample Truncation Strategies for Outlier Removal in Geochemical Data: The MCD Robust Distance Approach Versus t-SNE Ensemble Clustering" (https://doi.org/10.1007/s11004-019-09839-z). *Mathematical Geosciences*. **53** (1): 105–130. doi:10.1007/s11004-019-09839-z (https://doi.org/10.1007%2Fs11004-019-09839-z). ISSN 1874-8953 (https://www.worldcat.org/issn/1874-8953). S2CID 208329378 (https://api.semanticscholar.org/CorpusID:208329378).
10. Birjandtalab, J.; Pouyan, M. B.; Nourani, M. (2016-02-01). *Nonlinear dimension reduction for EEG-based epileptic seizure detection*. *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. pp. 595–598. doi:10.1109/BHI.2016.7455968 (https://doi.org/10.1109%2FBHI.2016.7455968). ISBN 978-1-5090-2455-1. S2CID 8074617 (https://api.semanticscholar.org/CorpusID:8074617).
11. "K-means clustering on the output of t-SNE" (https://stats.stackexchange.com/a/264647). *Cross Validated*. Retrieved 2018-04-16.
12. Pezzotti, Nicola; Lelieveldt, Boudewijn P. F.; Maaten, Laurens van der; Hollt, Thomas; Eisemann, Elmar; Vilanova, Anna (2017-07-01). "Approximated and User Steerable tSNE for Progressive Visual Analytics". *IEEE Transactions on Visualization and Computer Graphics*. **23** (7): 1739–1752. arXiv:1512.01655 (https://arxiv.org/abs/1512.01655). doi:10.1109/tvcg.2016.2570755 (https://doi.org/10.1109%2Ftvcg.2016.2570755). ISSN 1077-2626 (https://www.worldcat.org/issn/1077-2626). PMID 28113434 (https://pubmed.ncbi.nlm.nih.gov/28113434). S2CID 353336 (https://api.semanticscholar.org/CorpusID:353336).
13. Wattenberg, Martin; Viégas, Fernanda; Johnson, Ian (2016-10-13). "How to Use t-SNE Effectively" (https://distill.pub/2016/misread-tsne/). Distill. Retrieved 4 December 2017.
14. Linderman, George C.; Steinerberger, Stefan (2017-06-08). "Clustering with t-SNE, provably". arXiv:1706.02582 (https://arxiv.org/abs/1706.02582) [cs.LG (https://arxiv.org/archive/cs.LG)].
15. Schubert, Erich; Gertz, Michael (2017-10-04). *Intrinsic t-Stochastic Neighbor Embedding for Visualization and Outlier Detection*. SISAP 2017 – 10th International Conference on Similarity Search and Applications. pp. 188–203. doi:10.1007/978-3-319-68474-1_13 (https://doi.org/10.1007%2F978-3-319-68474-1_13).

External links

- Visualizing Data Using t-SNE (https://www.youtube.com/watch?v=RJVL80Gg3lA), Google Tech Talk about t-SNE
- Implementations of t-SNE in various languages (https://lvdmaaten.github.io/tsne/), A link collection maintained by Laurens van der Maaten