Concept of t-SNE

t-Distributed Stochastic Neighbor Embedding (t-SNE)

- a ML algorithm using for **dimensionality reduction**
- a variation of Stochastic Neighbor Embedding (SNE)

Main contributions

- a **symmetrized SNE** cost function with simpler gradients
- the similarity computed by a **Student-t distribution** in low-dimensional space

Symmetric SNE

Using joint probability distribution rather than conditional probability distribution

the pairwise similarities in the high-dimensional space pij:

$$p_{ij} = \frac{\exp\left(-\|x_i - x_j\|^2/2\sigma^2\right)}{\sum_{k \neq l} \exp\left(-\|x_k - x_l\|^2/2\sigma^2\right)}$$

the pairwise similarities in low-dimensional map qij:

$$q_{ij} = \frac{\exp\left(-\|y_i - y_j\|^2\right)}{\sum_{k \neq l} \exp\left(-\|y_k - y_l\|^2\right)}$$

The cost function (Kullback-Leibler divergences):

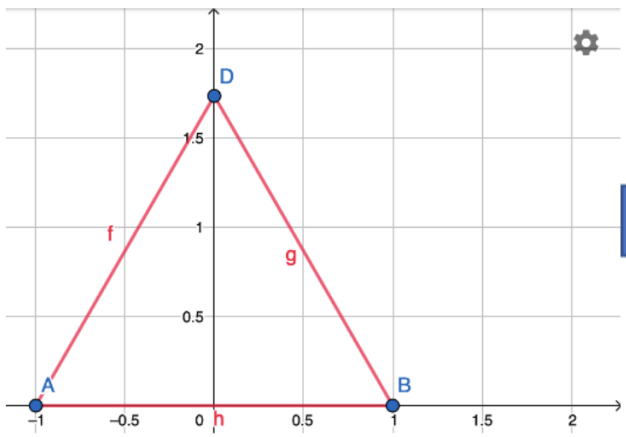
$$C = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

Therefore, the gradient of symmetric SNE will be optimized:

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)$$

The 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.



Where we place point D?

Design of t-SNE

High-dimensional space: Gaussian distribution

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)$$

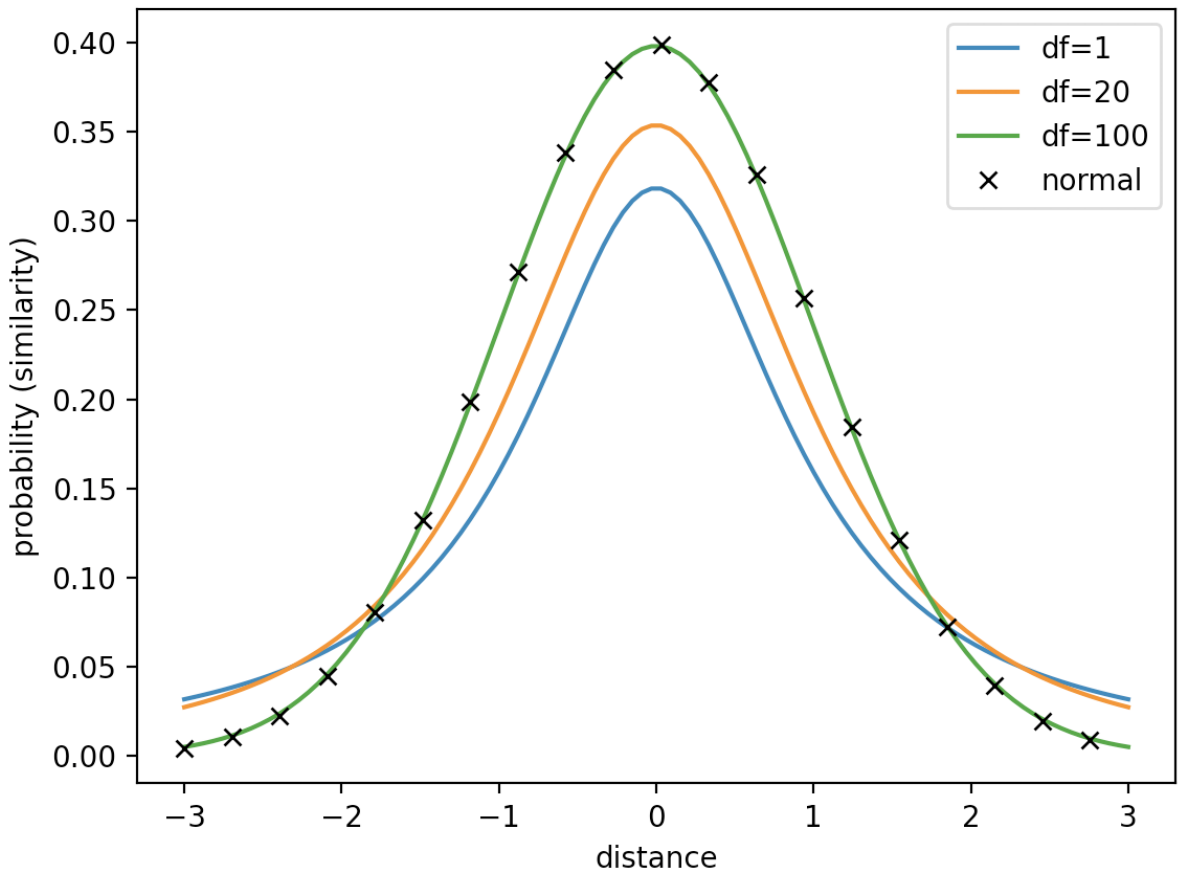
Low-dimensional space: **Student t-distribution**

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}}$$

The gradient of the Kullback-Leibler divergence:

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j) \left(1 + \|y_i - y_j\|^2\right)^{-1}$$

Differences of t-dis and norm-dis



Weekness of t-SNE

Dimensionality reduction for other purposes

- CANNOT be extrapolated to d¿3 dimensions

Curse of intrinsic dimensionality

- LESS successful on a dataset with a high intrinsic dimensionality

Non-convexity of the t-SNE cost function

- the cost function is NOT convex
- MORE dependent on the optimization parameters