

# SNE and t-SNE

Statistical Machine Learning (ENGG\*6600\*08)

School of Engineering,  
University of Guelph, ON, Canada

Course Instructor: Benyamin Ghogh  
Fall 2023

## Introduction

# Introduction



- **Stochastic Neighbor Embedding (SNE)** (2003) [1] is a manifold learning and dimensionality reduction method which can be used for feature extraction [2].
- It has a probabilistic approach. It fits the data in the embedding space locally hoping to preserve the global structure of data [3].
- The idea of SNE is to consider every point as neighbors of other points with some probability where the closer points are neighbors with higher probability. Therefore, rather than considering  $k$  nearest neighbors in a binary manner (whether being neighbors or not), it considers neighbors in a stochastic way (for how probable it is to be neighbors).
- It tries to preserve the probability of neighborhoods in the low-dimensional embedding space.
- There exist some other similar probabilistic dimensionality reduction methods which make use of Gaussian distribution for neighborhood. Some examples are Neighborhood Component Analysis (NCA) [4], deep NCA [5], and Proxy-NCA [6].
- SNE uses the Gaussian distribution for neighbors in both the input and embedding spaces. The Student-t distributed SNE, or so-called **t-SNE** [7], considers the **Student-t and Gaussian distributions** in the input and embedding spaces, respectively. The reason of using Student-t distribution in t-SNE is because of its heavier tails so it can include more information from the high-dimensional data.
- t-SNE is one of the state-of-the-art methods for **data visualization**; for example, it has been used for **DNA and single-cell** data visualization [8].
- The goal of SNE is to embed the high-dimensional data  $\{\mathbf{x}_i\}_{i=1}^n$  into the lower dimensional data  $\{\mathbf{y}_i\}_{i=1}^n$  where  $n$  is the number of data points. We denote the dimensionality of high- and low-dimensional spaces by  $d$  and  $p$ , respectively, i.e.  $\mathbf{x}_i \in \mathbb{R}^d$  and  $\mathbf{y}_i \in \mathbb{R}^p$ . We usually have  $p \ll d$ . For data visualization, we have  $p \in \{2, 3\}$ .

## Stochastic Neighbor Embedding (SNE)

# Stochastic Neighbor Embedding (SNE)

- In **SNE** (2003) [1], we consider a Gaussian probability around every point  $\mathbf{x}_i$  where the distribution is for probability of accepting any other point as the neighbor of  $\mathbf{x}_i$ ; the farther points are neighbors with less probability. Hence, the variable is distance, denoted by  $d \in \mathbb{R}$ , and the Gaussian probability is:

$$\star f(d) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{d^2}{2\sigma^2}\right), \quad (1)$$

where the mean of distribution is assumed to be zero.

- The fixed multiplier  $\frac{1}{\sqrt{2\pi\sigma^2}}$  can be **dropped**; however,  $\exp(-d^2/2\sigma^2)$  does not add (integrate) to one and thus it is not a probability density function. In order to tackle this problem, we can do a trick and divide  $\exp(-d^2/2\sigma^2)$  by the summation of all possible values of  $\exp(-d^2/2\sigma^2)$  to have a **softmax** function. Therefore, the probability that the point  $\mathbf{x}_i \in \mathbb{R}^d$  takes  $\mathbf{x}_j \in \mathbb{R}^d$  as its neighbor is:

$$\mathbb{R} \ni p_{ij} := \frac{\exp(-d_{ij}^2)}{\sum_{k \neq i} \exp(-d_{ik}^2)}, \quad (2)$$

where:

$$\mathbb{R} \ni d_{ij}^2 := \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma_i^2}. \quad (3)$$

# Stochastic Neighbor Embedding (SNE)

- The  $\sigma_i^2$  is the **variance** which we consider for the Gaussian distribution used for the  $\mathbf{x}_i$ . It can be set to a **fixed number** or determined by a **binary search** to make the entropy of distribution some **specific value** [1]. Note that according to the distribution of data in the input space, the best value for the variance of Gaussian distributions can be found.
- In the low-dimensional embedding space, we again consider a **Gaussian probability** distribution for the point  $\mathbf{y}_i \in \mathbb{R}^p$  to take  $\mathbf{y}_j \in \mathbb{R}^p$  as its neighbor:

$$\star \quad \underbrace{\mathbb{R} \ni q_{ij}} := \frac{\exp(-z_{ij}^2)}{\sum_{k \neq i} \exp(-z_{ik}^2)}, \quad (4)$$

where:

$$\mathbb{R} \ni \underbrace{z_{ij}^2} := \|\mathbf{y}_i - \mathbf{y}_j\|_2^2. \quad \leftarrow (5)$$

- It is noteworthy that the variance of distribution is not used (or is set to  $\sigma_i^2 = 0.5$  to cancel 2 in the denominator) because the variance of distribution in the **embedding space** is the choice of algorithm.

# Stochastic Neighbor Embedding (SNE)

- We want the probability distributions in both the input and embedded spaces to be as similar as possible; therefore, the cost function to be minimized can be summation of the **Kullback-Leibler (KL) divergences** [9] over the  $n$  points:

$$\min_{\{y_i\}} \mathbb{R} \ni c_1 := \underbrace{\sum_{i=1}^n \text{KL}(P_i || Q_i)}_{\text{KL divergence}} = \underbrace{\sum_{i=1}^n \sum_{j=1, j \neq i}^n p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right)}_{\text{KL divergence}}, \quad (6)$$

where  $p_{ij}$  and  $q_{ij}$  are the Eqs. (2) and (4).

- Note that divergences other than the KL divergence can be used for the SNE optimization; e.g., see [10].
- The gradient of  $c_1$  with respect to  $y_i$  is:

$$\mathbb{R}^p \ni \frac{\partial c_1}{\partial y_i} = 2 \underbrace{\sum_{j=1}^n (p_{ij} - q_{ij} + p_{ji} - q_{ji})(y_i - y_j)}_{\text{gradient}}, \quad (7)$$

where  $p_{ij}$  and  $q_{ij}$  are the Eqs. (2) and (4), and  $p_{ii} = q_{ii} = 0$ .

- For proof of this, refer to our tutorial "Stochastic neighbor embedding with Gaussian and student-t distributions: Tutorial and survey" [11] or our textbook.

# Stochastic Neighbor Embedding (SNE)

- The update of the embedded point  $\mathbf{y}_i$  is done by gradient descent. Every iteration is:

$$\begin{aligned} \Delta \mathbf{y}_i^{(t)} &:= -\eta \frac{\partial c_1}{\partial \mathbf{y}_i} + \alpha(t) \Delta \mathbf{y}_i^{(t-1)}, \\ \mathbf{y}_i^{(t)} &:= \mathbf{y}_i^{(t-1)} + \Delta \mathbf{y}_i^{(t)}, \end{aligned} \quad (8)$$

*Handwritten annotations:* Brackets group the terms in the equations. An arrow points from the word "momentum" to  $\alpha(t)$ . Another arrow points from the word "momentum" to the term  $\Delta \mathbf{y}_i^{(t-1)}$ .

where **momentum** is used for better convergence [12].

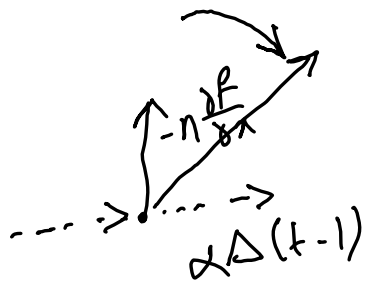
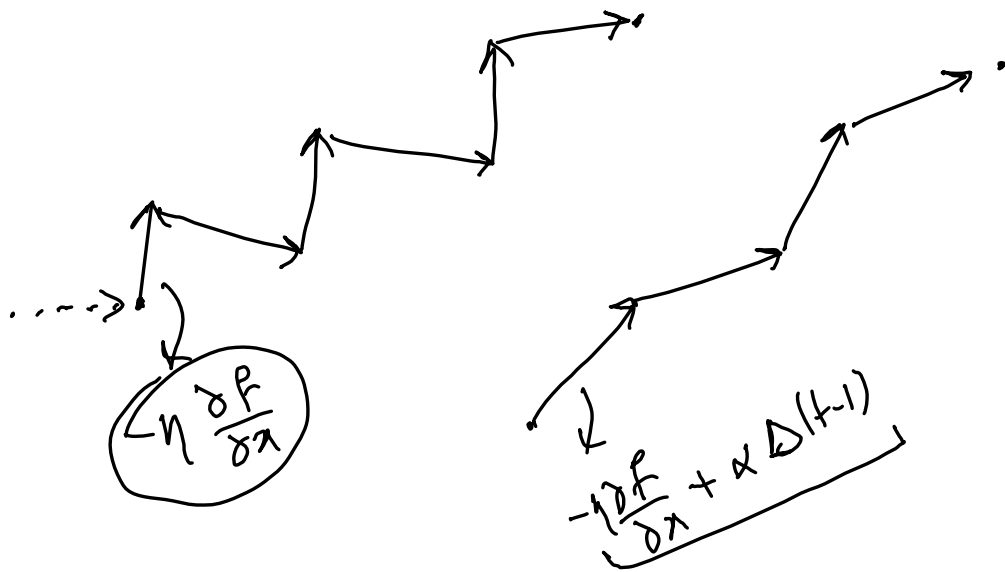
- The  $\alpha(t)$  is the **momentum**. It can be smaller for initial iterations and larger for further iterations. For example, we can have [7]:

$$\alpha(t) := \begin{cases} \frac{0.5}{0.8} & t < 250, \\ \frac{0.8}{0.8} & t \geq 250. \end{cases} \quad (9)$$

In the original paper of SNE [1], the momentum term is not mentioned but it is suggested in [7].

- The  $\eta$  is the learning rate which can be a small positive constant (e.g.,  $\eta = 0.1$ ) or can be updated adaptively according to [13].
- Moreover, in both [1] and [7], it is mentioned that in SNE we should add some Gaussian noise (random jitter) to the solution of the first iterations before going to the next iterations. It helps avoiding the local optimum solutions.





## Symmetric Stochastic Neighbor Embedding

# Symmetric Stochastic Neighbor Embedding

- In symmetric SNE (2008) [7], we consider a Gaussian probability around every point  $\mathbf{x}_i$ . The probability that the point  $\mathbf{x}_j \in \mathbb{R}^d$  takes  $\mathbf{x}_j \in \mathbb{R}^d$  as its neighbor is:

$$\mathbb{R} \ni p_{ij} := \frac{\exp(-d_{ij}^2)}{\sum_{k \neq i} \exp(-d_{ik}^2)}, \quad (10)$$

where:

$$\mathbb{R} \ni d_{ij}^2 := \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma_i^2}. \quad (11)$$

- Note that the **denominator** of Eq. (10) for all points is fixed and thus it is symmetric for  $i$  and  $j$ . Compare this with Eq. (2):

$\ddot{j} \rightarrow \dot{j}$   $\rightarrow$   $\mathbb{R} \ni p_{ij} := \frac{\exp(-d_{ij}^2)}{\sum_{k \neq i} \exp(-d_{ik}^2)},$

which is not symmetric.

# Symmetric Stochastic Neighbor Embedding

- The Eq. (10):

$$\mathbb{R} \ni p_{ij} := \frac{\exp(-\widehat{d_{ij}^2})}{\sum_{k \neq i} \exp(-\widehat{d_{ki}^2})}$$

*Handwritten notes: A large arrow points from the denominator to the text "small for the outlier". There is a large 'X' with an arrow pointing to the denominator, indicating a problem with this formula.*

has a **problem with outliers**. If the point  $\mathbf{x}_i$  is an outlier, its  $p_{ij}$  will be **extremely small** because the denominator is fixed for every point and numerator will be **small for the outlier**.

- However, If we use Eq. (2) for  $p_{ij}$ :

$$\mathbb{R} \ni p_{ij} := \frac{\exp(-\widehat{d_{ij}^2})}{\sum_{k \neq i} \exp(-\widehat{d_{ik}^2})}$$

*Handwritten notes: The formula is enclosed in a box. A large arrow points from the denominator to the text "the denominator for an outlier will also be small".*

the denominator for all the points is not the same and therefore, **the denominator for an outlier will also be small waving out the problem of small numerator.**

- For this mentioned problem, we do not use Eq. (10) and rather we use:

$$\mathbb{R} \ni \underline{p_{ij}} := \frac{\widehat{p_{i|j}} + \widehat{p_{j|i}}}{2n}, \quad (12)$$

where:

$$\mathbb{R} \ni \underline{p_{j|i}} := \frac{\exp(-\widehat{d_{ij}^2})}{\sum_{k \neq i} \exp(-\widehat{d_{ik}^2})}, \quad (13)$$

is the probability that  $\mathbf{x}_i \in \mathbb{R}^d$  takes  $\mathbf{x}_j \in \mathbb{R}^d$  as its neighbor.

# Symmetric Stochastic Neighbor Embedding

- In the low-dimensional embedding space, we consider a Gaussian probability distribution for the point  $\mathbf{y}_i \in \mathbb{R}^p$  to take  $\mathbf{y}_j \in \mathbb{R}^p$  as its neighbor and we make it **symmetric (fixed denominator for all points)**:

$$\star \quad \mathbb{R} \ni q_{ij} := \frac{\exp(-z_{ij}^2)}{\sum_{k \neq i} \exp(-z_{ki}^2)}, \quad \leftarrow \quad (14)$$

where:

$$\mathbb{R} \ni z_{ij}^2 := \|\mathbf{y}_i - \mathbf{y}_j\|_2^2. \quad (15)$$

- Note that the Eq. (14) does **not have the problem of outliers** as in Eq. (10) because even for an **outlier**, the **embedded points are initialized close together and not far**.

# Symmetric Stochastic Neighbor Embedding

- We want the probability distributions in both the input and embedded spaces to be as similar as possible; therefore, the cost function to be minimized can be summation of the **Kullback-Leibler (KL) divergences** [9] over the  $n$  points:

$$\mathbb{R} \ni \underbrace{c_2 := \sum_{i=1}^n \text{KL}(P_i || Q_i) = \sum_{i=1}^n \sum_{j=1, j \neq i}^n p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right)}_{\text{KL divergences}}, \quad (16)$$

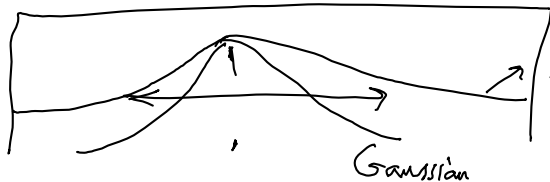
where  $p_{ij}$  and  $q_{ij}$  are the Eqs. (12) and (14).

- The gradient of  $c_2$  with respect to  $\mathbf{y}_i$  is:

$$\mathbb{R}^p \ni \underbrace{\frac{\partial c_2}{\partial \mathbf{y}_i} = 4 \sum_{j=1}^n (p_{ij} - q_{ij})(\mathbf{y}_i - \mathbf{y}_j)}_{\text{gradient}}, \quad (17)$$

where  $p_{ij}$  and  $q_{ij}$  are the Eqs. (12) and (14), and  $p_{ii} = q_{ii} = 0$ .

- For proof of this, refer to our tutorial “Stochastic neighbor embedding with Gaussian and student-t distributions: Tutorial and survey” [11] or our textbook.

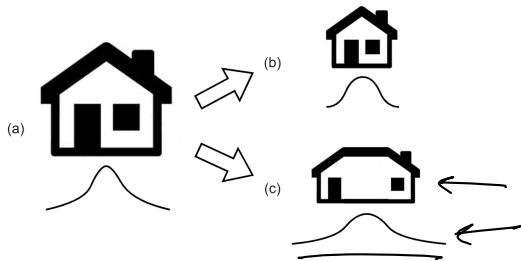


Student-t distribution  
special case:  
(Cauchy distribution)

t-distributed  
Stochastic Neighbor  
Embedding (t-SNE)

# The Crowding Problem

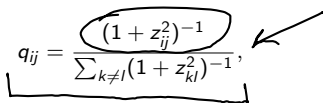
- In SNE [1], we are considering **Gaussian** distribution for both input and embedded spaces.
- That is okay for the **input space** because it already has a high dimensionality.
- However, when we **embed the high-dimensional data into a low-dimensional space**, it is very hard to fit the **information** of all the points in the same neighborhood area.
- For better clarification, suppose the dimensionality is like the size of a room, as depicted in this figure. In high dimensionality, we have a large hall including a huge crowd of people. Now, we want to fit all the people into a small room; of course, we cannot! This problem is referred to as the **crowding problem**.



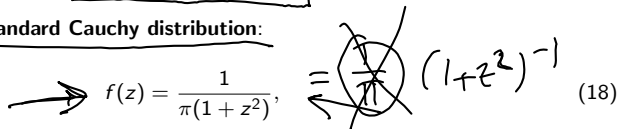


# t-distributed Stochastic Neighbor Embedding (t-SNE)

- The main idea of **t-SNE** (2008) [7] is addressing the **crowding problem** which exists in SNE [1].
- In the example of fitting people in a room, t-SNE **enlarges** the room to solve the crowding problem (see the figure).
- Therefore, in the formulation of t-SNE, we use **Student-t distribution [14]** rather than **Gaussian distribution** for the **low-dimensional embedded space**.
- This is because the Student-t distribution has **heavier tails** than Gaussian distribution, which is like a larger room, and can fit the information of high dimensional data in the low dimensional embedding space.
- As we will see later, the  $q_{ij}$  in t-SNE is:

$$q_{ij} = \frac{(1 + z_{ij}^2)^{-1}}{\sum_{k \neq l} (1 + z_{kl}^2)^{-1}},$$


which is based on the standard Cauchy distribution:

$$\rightarrow f(z) = \frac{1}{\pi(1 + z^2)}, \quad \text{where } \frac{1}{\pi} \text{ is canceled out in the final expression } (1 + z^2)^{-1} \quad (18)$$


where  $\pi$  is canceled from the numerator and the normalizing denominator in  $q_{ij}$  (similar to the technique of **softmax**).

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- If the Student-t distribution [14] with the general degrees of freedom  $\delta$  is used, we would have:

$$f(z) = \frac{\Gamma(\frac{\delta+1}{2})}{\sqrt{\delta \times \pi} \Gamma(\frac{\delta}{2})} (1 + \frac{z^2}{\delta})^{-\frac{\delta+1}{2}}, \quad \leftarrow \quad (19)$$

where  $\Gamma$  is the gamma function.

- Cancelling out the scaling factors from the numerator and denominator, we would have [15]:

$$q_{ij} = \frac{(1 + z_{ij}^2/\delta)^{-(\delta+1)/2}}{\sum_{k \neq l} (1 + z_{kl}^2/\delta)^{-(\delta+1)/2}}. \quad (20)$$

- However, as the first degree of freedom has the heaviest tails amongst different degrees of freedom, it is the most suitable for the crowding problem; hence, we use the **first degree of freedom** which is the **Cauchy distribution**. Note that the t-SNE algorithm, which uses the Cauchy distribution, may also be called the **Cauchy-SNE**.
- Later, t-SNE with general degrees of freedom was proposed [15].

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- In t-SNE [7], we consider a **Gaussian** probability around every point  $\mathbf{x}_i$  in the input space because the **crowding problem does not exist in the high dimensional data**. The probability that the point  $\mathbf{x}_i \in \mathbb{R}^d$  takes  $\mathbf{x}_j \in \mathbb{R}^d$  as its neighbor is:

$$\mathbb{R} \ni p_{j|i} := \frac{\exp(-d_{ij}^2)}{\sum_{k \neq i} \exp(-d_{ik}^2)}, \quad (21)$$

where:

$$\mathbb{R} \ni d_{ij}^2 := \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma_i^2}. \quad (22)$$

- Note that Eq. (21) is not symmetric for  $i$  and  $j$  because of the denominator. We take the **symmetric**  $p_{ij}$  as the scaled average of  $p_{i|j}$  and  $p_{j|i}$ :

$$\mathbb{R} \ni p_{ij} := \frac{p_{i|j} + p_{j|i}}{2n}. \quad (23)$$

- In the low-dimensional **embedding space**, we consider a **Student's  $t$ -distribution with one degree of freedom (Cauchy distribution)** for the point  $\mathbf{y}_i \in \mathbb{R}^p$  to take  $\mathbf{y}_j \in \mathbb{R}^p$  as its neighbor:

$$\mathbb{R} \ni q_{ij} := \frac{(1 + z_{ij}^2)^{-1}}{\sum_{k \neq i} (1 + z_{ik}^2)^{-1}}, \quad (24)$$

where:

$$\mathbb{R} \ni z_{ij}^2 := \|\mathbf{y}_i - \mathbf{y}_j\|_2^2. \quad (25)$$

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- We want the probability distributions in both the input and embedded spaces to be as similar as possible; therefore, the cost function to be minimized can be summation of the **Kullback-Leibler (KL) divergences** [9] over the  $n$  points:

$$\mathbb{R} \ni \underline{c_3} := \sum_{i=1}^n \text{KL}(P_i || Q_i) = \sum_{i=1}^n \sum_{j=1, j \neq i}^n p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right), \quad (26)$$

where  $p_{ij}$  and  $q_{ij}$  are the Eqs. (23) and (24).

- The gradient of  $c_3$  with respect to  $\mathbf{y}_i$  is:

$$\frac{\partial c_3}{\partial \mathbf{y}_i} = 4 \underbrace{\sum_{j=1}^n (p_{ij} - q_{ij})(1 + \|\mathbf{y}_i - \mathbf{y}_j\|_2^2)^{-1}(\mathbf{y}_i - \mathbf{y}_j)}, \quad (27)$$

where  $p_{ij}$  and  $q_{ij}$  are the Eqs. (23) and (24), and  $p_{ii} = q_{ii} = 0$ .

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- Proof: Proof is according to [7]. Let:

$$\mathbb{R} \ni \underline{r_{ij}} := \underline{z_{ij}^2} = \underline{\|\mathbf{y}_i - \mathbf{y}_j\|_2^2}. \quad (28)$$

- By changing  $\mathbf{y}_i$ , we only have change impact in  $\underline{z_{ij}}$  and  $\underline{z_{ji}}$  for all  $j$ 's. According to chain rule, we have:

$$\mathbb{R}^p \ni \frac{\partial c_3}{\partial \mathbf{y}_i} = \sum_j \left( \frac{\partial c_3}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial \mathbf{y}_i} + \frac{\partial c_3}{\partial r_{ji}} \frac{\partial r_{ji}}{\partial \mathbf{y}_i} \right).$$

- According to Eq. (28), we have:

$$\begin{aligned} \underline{r_{ij}} = \|\mathbf{y}_i - \mathbf{y}_j\|_2^2 &\Rightarrow \underline{\frac{\partial r_{ij}}{\partial \mathbf{y}_i}} = 2(\mathbf{y}_i - \mathbf{y}_j), \\ \underline{r_{ji}} = \|\mathbf{y}_j - \mathbf{y}_i\|_2^2 = \|\mathbf{y}_i - \mathbf{y}_j\|_2^2 &\Rightarrow \underline{\frac{\partial r_{ji}}{\partial \mathbf{y}_i}} = 2(\mathbf{y}_i - \mathbf{y}_j). \end{aligned}$$

- Therefore:

$$\therefore \frac{\partial c_3}{\partial \mathbf{y}_i} = 2 \sum_j \left( \frac{\partial c_3}{\partial r_{ij}} + \frac{\partial c_3}{\partial r_{ji}} \right) (\mathbf{y}_i - \mathbf{y}_j). \quad (29)$$

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- The cost function can be re-written as:

$$\begin{aligned} \star c_3 &= \sum_k \sum_{l \neq k} p_{kl} \log\left(\frac{p_{kl}}{q_{kl}}\right) = \sum_{k \neq l} p_{kl} \log\left(\frac{p_{kl}}{q_{kl}}\right) \\ &= \sum_{k \neq l} \left( \underbrace{p_{kl} \log(p_{kl})}_{\text{constant}} - \underbrace{p_{kl} \log(q_{kl})}_{\text{term to minimize}} \right), \end{aligned}$$

whose first term is a constant with respect to  $q_{kl}$  and thus to  $r_{kl}$ .

- We have:

$$\mathbb{R} \ni \frac{\partial c_3}{\partial r_{ij}} = \ominus \sum_{k \neq l} p_{kl} \frac{\partial (\log(q_{kl}))}{\partial r_{ij}}.$$

- According to Eq. (24):

$$\mathbb{R} \ni q_{ij} := \frac{(1 + z_{ij}^2)^{-1}}{\sum_{k \neq l} (1 + z_{kl}^2)^{-1}},$$

the  $q_{kl}$  is:

$$q_{kl} := \frac{(1 + \hat{z}_{kl}^2)^{-1}}{\sum_{m \neq f} (1 + \hat{z}_{mf}^2)^{-1}} = \frac{(1 + \hat{r}_{kl})^{-1}}{\sum_{m \neq f} (1 + \hat{r}_{mf})^{-1}}.$$

- We take the denominator of  $q_{kl}$  as:

$$\beta := \sum_{m \neq f} (1 + \hat{z}_{mf}^2)^{-1} = \sum_{m \neq f} (1 + \hat{r}_{mf})^{-1}. \quad (30)$$

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- We had:

$$\mathbb{R} \ni \frac{\partial c_3}{\partial r_{ij}} = - \sum_{k \neq l} p_{kl} \frac{\partial(\log(q_{kl}))}{\partial r_{ij}}, \quad q_{kl} := \frac{(1 + z_{kl}^2)^{-1}}{\sum_{m \neq f} (1 + z_{mf}^2)^{-1}} = \frac{(1 + r_{kl})^{-1}}{\sum_{m \neq f} (1 + r_{mf})^{-1}},$$

$$\beta := \sum_{m \neq f} (1 + z_{mf}^2)^{-1} = \sum_{m \neq f} (1 + r_{mf})^{-1}.$$

- We have  $\log(q_{kl}) = \log(q_{kl}) + \log \beta - \log \beta = \log(q_{kl}\beta) - \log \beta$ . Therefore:

$$\begin{aligned} \therefore \frac{\partial c_3}{\partial r_{ij}} &= - \sum_{k \neq l} p_{kl} \frac{\partial(\log(q_{kl}\beta) - \log \beta)}{\partial r_{ij}} = - \sum_{k \neq l} p_{kl} \left[ \frac{\partial(\log(q_{kl}\beta))}{\partial r_{ij}} - \frac{\partial(\log \beta)}{\partial r_{ij}} \right] \\ &= - \sum_{k \neq l} p_{kl} \left[ \frac{1}{q_{kl}\beta} \frac{\partial(q_{kl}\beta)}{\partial r_{ij}} - \frac{1}{\beta} \frac{\partial \beta}{\partial r_{ij}} \right]. \end{aligned}$$

- The  $q_{kl}\beta$  is:

$$(q_{kl}\beta) = \frac{(1 + r_{kl})^{-1}}{\sum_{m \neq f} (1 + r_{mf})^{-1}} \times \sum_{m \neq f} (1 + r_{mf})^{-1} = (1 + r_{kl})^{-1}.$$

- Therefore, we have:

$$\therefore \frac{\partial c_3}{\partial r_{ij}} = - \sum_{k \neq l} p_{kl} \left[ \frac{1}{q_{kl}\beta} \frac{\partial((1 + r_{kl})^{-1})}{\partial r_{ij}} - \frac{1}{\beta} \frac{\partial \beta}{\partial r_{ij}} \right].$$

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- We found:

$$\star \frac{\partial c_3}{\partial r_{ij}} = \sum_{k \neq l} p_{kl} \left[ \frac{1}{q_{kl}\beta} \frac{\partial((1+r_{kl})^{-1})}{\partial r_{ij}} \frac{1}{\beta} \frac{\partial \beta}{\partial r_{ij}} \right]$$

- The  $\partial((1+r_{kl})^{-1})/\partial r_{ij}$  is non-zero for only  $k=i$  and  $l=j$ ; therefore:

$$\frac{\partial((1+r_{ij})^{-1})}{\partial r_{ij}} = -(1+r_{ij})^{-2},$$

$$\frac{\partial \beta}{\partial r_{ij}} = \frac{\partial(\sum_{m \neq l} (1+r_{mf})^{-1})}{\partial r_{ij}} = \frac{\partial(1+\hat{r}_{ij})^{-1}}{\partial r_{ij}} = -(1+r_{ij})^{-2}$$

- Therefore:

$$\therefore \frac{\partial c_3}{\partial r_{ij}} = \left( p_{ij} \left[ \frac{-1}{q_{ij}\beta} (1+r_{ij})^{-2} \right] + 0 + \dots + 0 \right) \frac{1}{\beta} (1+r_{ij})^{-2}$$

- We have  $\sum_{k \neq l} p_{kl} = 1$  because summation of all possible probabilities is one. Thus:

$$\star \frac{\partial c_3}{\partial r_{ij}} = p_{ij} \left[ \frac{-1}{q_{ij}\beta} (1+r_{ij})^{-2} \right] \frac{1}{\beta} (1+r_{ij})^{-2} = (1+r_{ij})^{-1} \frac{(1+r_{ij})^{-1}}{\beta} \left[ \frac{p_{ij}}{q_{ij}} - 1 \right]$$

$\underline{\underline{= q_{ij}}}$

$$= (1+r_{ij})^{-1} (p_{ij} - q_{ij}).$$



# t-distributed Stochastic Neighbor Embedding (t-SNE)

- We found:

$$\star \left( \frac{\partial c_3}{\partial r_{ij}} \right) = (1 + \underline{r_{ij}})^{-1} (\underline{p_{ij}} - \underline{q_{ij}}).$$

- Similarly, we have:

$$\star \left( \frac{\partial c_3}{\partial r_{ji}} \right) = (1 + \underline{r_{ji}})^{-1} (\underline{p_{ji}} - \underline{q_{ji}}) \stackrel{(a)}{=} (1 + \underline{r_{ij}})^{-1} (\underline{p_{ij}} - \underline{q_{ij}}),$$

where (a) is because in t-SNE, the  $p_{ij}$ ,  $q_{ij}$ , and  $r_{ij}$  are symmetric for  $i$  and  $j$  according to Eqs. (23), (24), and (28).

- Substituting the obtained derivatives in Eq. (29):

$$\therefore \frac{\partial c_3}{\partial \mathbf{y}_i} = \underbrace{2 \sum_j \left( \frac{\partial c_3}{\partial r_{ij}} + \frac{\partial c_3}{\partial r_{ji}} \right) (\mathbf{y}_i - \mathbf{y}_j)}_{\text{gives us:}}$$

gives us:

$$\frac{\partial c_3}{\partial \mathbf{y}_i} = 4 \sum_j (\underline{p_{ij}} - \underline{q_{ij}}) (1 + \underline{r_{ij}})^{-1} \underline{(\mathbf{y}_i - \mathbf{y}_j)},$$

which is the gradient mentioned before. Q.E.D.

# t-distributed Stochastic Neighbor Embedding (t-SNE)

- The update of the embedded point  $\mathbf{y}_i$  is done by gradient descent whose every iteration is as Eq. (8) where  $c_1$  is replaced by  $c_3$ :

$$\begin{cases} \Delta \mathbf{y}_i^{(t)} := -\eta \frac{\partial c_1}{\partial \mathbf{y}_i} + \alpha(t) \Delta \mathbf{y}_i^{(t-1)}, \\ \mathbf{y}_i^{(t)} := \mathbf{y}_i^{(t-1)} + \Delta \mathbf{y}_i^{(t)}. \end{cases}$$

- For t-SNE, there is no need to add jitter to the solution of initial iterations [7] because it is **more robust than SNE**.
- In t-SNE, it is better to multiply all  $p_{ij}$ 's by a constant (e.g., 4) in the initial iterations:

$$\underline{p_{ij} := p_{ij} \times 4}, \quad (31)$$

which is called early exaggeration. This heuristic helps the optimization focus on the large  $p_{ij}$ 's (close neighbors) more in the early iterations.

- This is because **large  $p_{ij}$ 's are affected more by multiplying by 4 than the small  $p_{ij}$ 's**.
- **After the neighbours are embedded close to one another**, we are free not to do this multiplication any more and let far-away points be embedded using the probabilities without multiplication. Note that the early exaggeration is optional and not mandatory.

# t-distributed Stochastic Neighbor Embedding (t-SNE)

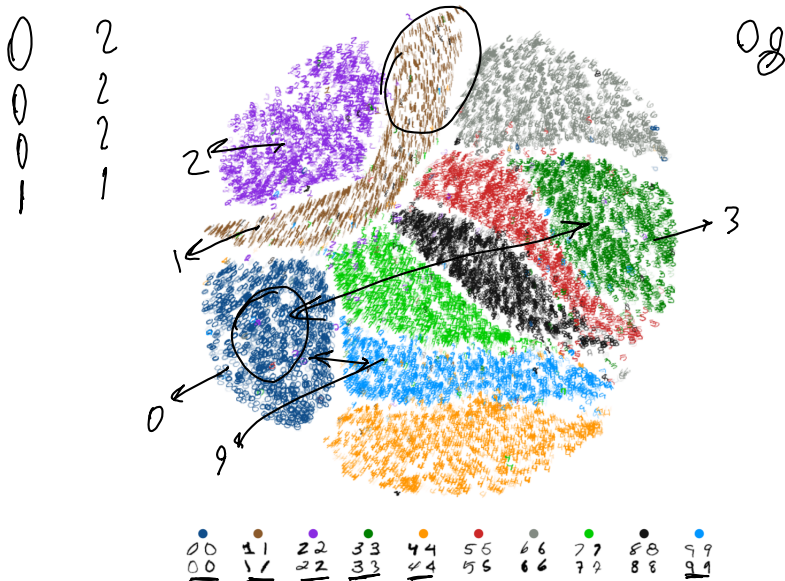
- We can have **general degrees of freedom** for **Student-t distribution** in t-SNE [15].
- As we saw in Eqs. (19) and (20), we can have any degrees of freedom for  $q_{ij}$  (note that  $\alpha$  is a positive integer). We repeat Eq. (20) here for more convenience:

$$q_{ij} = \frac{(1 + z_{ij}^2/\delta)^{-(\delta+1)/2}}{\sum_{k \neq l} (1 + z_{kl}^2/\delta)^{-(\delta+1)/2}}. \quad (32)$$

- If  $\delta \rightarrow \infty$ , the Student-t distribution formulated in Eq. (19) tends to Gaussian distribution used in SNE [1].
- SNE and t-SNE use degrees  $\delta \rightarrow \infty$  and  $\delta = 1$  in Eq. (32), respectively.



# Example of t-SNE Embedding (Digit Dataset)



Credit of image: [16]

# Acknowledgment

- Some slides are based on our tutorial paper: "Stochastic neighbor embedding with Gaussian and student-t distributions: Tutorial and survey" [11]
- For more information on SNE and t-SNE, refer to our tutorial paper [11].
- Some slides of this slide deck are inspired by teachings of Prof. Ali Ghodsi at University of Waterloo, Department of Statistics.
- The code of SNE and t-SNE in my GitHub: <https://github.com/bghojogh/SNE-tSNE>
- t-SNE in sklearn: <https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html>

# References

- [1] G. E. Hinton and S. T. Roweis, “Stochastic neighbor embedding,” in *Advances in neural information processing systems*, pp. 857–864, 2003.
- [2] B. Ghojogh, M. N. Samad, S. A. Mashhadi, T. Kapoor, W. Ali, F. Karray, and M. Crowley, “Feature selection and feature extraction in pattern analysis: A literature review,” *arXiv preprint arXiv:1905.02845*, 2019.
- [3] L. K. Saul and S. T. Roweis, “Think globally, fit locally: unsupervised learning of low dimensional manifolds,” *Journal of machine learning research*, vol. 4, no. Jun, pp. 119–155, 2003.
- [4] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov, “Neighbourhood components analysis,” in *Advances in neural information processing systems*, pp. 513–520, 2005.
- [5] X. Liu, X. Yang, M. Wang, and R. Hong, “Deep neighborhood component analysis for visual similarity modeling,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 11, no. 3, pp. 1–15, 2020.
- [6] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh, “No fuss distance metric learning using proxies,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 360–368, 2017.
- [7] L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

## References (cont.)

UMAP

- [8] D. Kobak and P. Berens, “The art of using t-SNE for single-cell transcriptomics,” *Nature communications*, vol. 10, no. 1, pp. 1–14, 2019.
- [9] S. Kullback, *Information theory and statistics*. Courier Corporation, 1997.
- [10] D. J. Im, N. Verma, and K. Branson, “Stochastic neighbor embedding under f-divergences,” *arXiv preprint arXiv:1811.01247*, 2018.
- [11] B. Ghojogh, A. Ghodsi, F. Kararay, and M. Crowley, “Stochastic neighbor embedding with gaussian and student-t distributions: Tutorial and survey,” *arXiv preprint arXiv:2009.10301*, 2020.
- [12] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural networks*, vol. 12, no. 1, pp. 145–151, 1999.
- [13] R. A. Jacobs, “Increased rates of convergence through learning rate adaptation,” *Neural networks*, vol. 1, no. 4, pp. 295–307, 1988.
- [14] W. S. Gosset (Student), “The probable error of a mean,” *Biometrika*, pp. 1–25, 1908.
- [15] L. van der Maaten, “Learning a parametric embedding by preserving local structure,” in *Artificial Intelligence and Statistics*, pp. 384–391, 2009.
- [16] N. Pezzotti, “Dimensionality-reduction algorithms for progressive visual analytics,” 2019.