

Stochastic Neighborhood Embedding

Weekly AI pills

Fabio Brau.

2020-11-13

SSSA, Emerging Digital Technologies, Pisa.

ISTITUTO
DI TECNOLOGIE DELLA
COMUNICAZIONE,
DELL'INFORMAZIONE
E DELLA
PERCEZIONE



Scuola Superiore
Sant'Anna



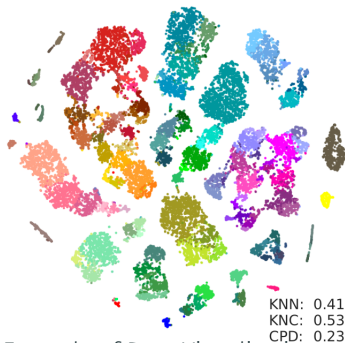
1. Entropy and Kullback–Leibler divergence
2. From SNE to t-SNE
3. Application for Visualization
4. Issues



(t)-Stochastic Neighborhood Embedding

Aims

- Dimensionality Reduction
- Data Visualization
 1. Exploration
 2. Visual Clustering



Example of Data Visualization
taken from (Tasic et al., 2018)

Algorithm



Algorithm: Workflow

1. Represent each sample $x_i \in \mathcal{X}$ with $y_i \in \mathcal{Y}$ in a low-dimensional space

$$\Phi : \underset{\subseteq \mathbb{R}^n}{\mathcal{X}} \longrightarrow \underset{\subseteq \mathbb{R}^2}{\mathcal{Y}}$$

2. Convert **Geometrical Information** to **Probabilistic Distribution**

$p_{i|j}$: “Probability that x_i is *similar* to x_j ”

$q_{i|j}$: “Probability that y_i is *similar* to y_j ”

3. Compare distribution \mathcal{P} of \mathcal{X} to distribution \mathcal{Q} of \mathcal{X}
4. Adjust the representation Φ to make distributions closer.

Observation

The embedding Φ is defined point-wise, i.e Φ is only defined over \mathcal{X} through the definition $\Phi(x_i) = y_i$. Another way to say “ y_i are the parameters of Φ ”.

Algorithm: Workflow

1. Represent each sample $x_i \in \mathcal{X}$ with $y_i \in \mathcal{Y}$ in a low-dimensional space

$$\Phi : \underset{\subseteq \mathbb{R}^n}{\mathcal{X}} \longrightarrow \underset{\subseteq \mathbb{R}^2}{\mathcal{Y}}$$

2. Convert **Geometrical Information** to **Probabilistic Distribution**

$p_{i|j}$: “Probability that x_i is *similar* to x_j ”

$q_{i|j}$: “Probability that y_i is *similar* to y_j ”

3. Compare distribution \mathcal{P} of \mathcal{X} to distribution \mathcal{Q} of \mathcal{X}
4. Adjust the representation Φ to make distributions closer.

Observation

The embedding Φ is defined **point-wise**, i.e Φ is only defined over \mathcal{X} through the definition $\Phi(x_i) = y_i$. Another way to say “ y_i are the parameters of Φ ”.

Algorithm: Similarity and Probability Distribution

Definition (Similarity with Gaussian Kernel)

For each x_i we consider the similarity

$$\forall j \neq i, \quad p_{ij} = \frac{\exp(-\|x_i - x_j\|^2 / (2\sigma_i)^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / (2\sigma_i)^2)} \quad (\mathcal{P}_i)$$

where σ_i is an hyper-parameter depending on x_i .

How σ_i impacts on p_{ij} ?

Algorithm: Similarity and Probability Distribution

Definition (Similarity with Gaussian Kernel)

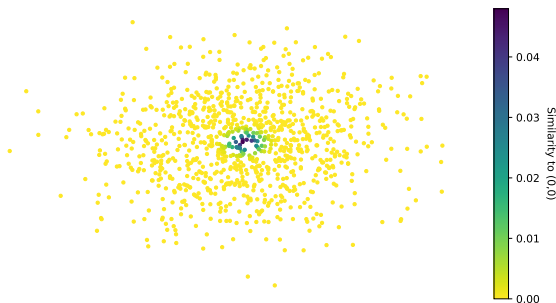
For each x_i we consider the similarity

$$\forall j \neq i, \quad p_{i|j} = \frac{\exp(-\|x_i - x_j\|^2 / (2\sigma_i)^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / (2\sigma_i)^2)} \quad (\mathcal{P}_i)$$

where σ_i is an hyper-parameter depending on x_i .

How σ_i impacts on $p_{i|j}$?

Similarity to (0,0) for sigma=0.1



Algorithm: Similarity and Probability Distribution

Definition (Similarity with Gaussian Kernel)

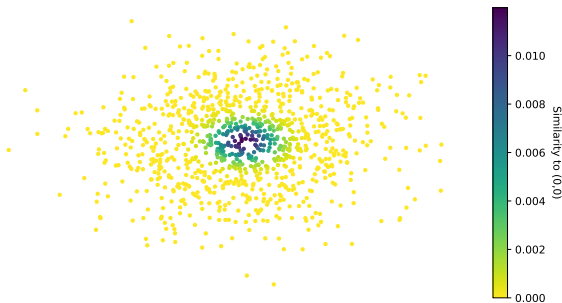
For each x_i we consider the similarity

$$\forall j \neq i, \quad p_{i|j} = \frac{\exp(-\|x_i - x_j\|^2 / (2\sigma_i)^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / (2\sigma_i)^2)} \quad (\mathcal{P}_i)$$

where σ_i is an hyper-parameter depending on x_i .

How σ_i impacts on $p_{i|j}$?

Similarity to (0,0) for sigma=0.2



Algorithm: Similarity and Probability Distribution

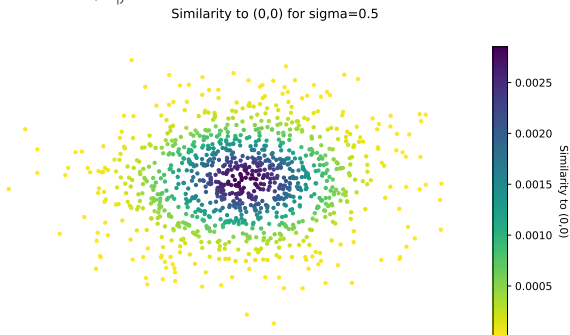
Definition (Similarity with Gaussian Kernel)

For each x_i we consider the similarity

$$\forall j \neq i, \quad p_{ij} = \frac{\exp(-\|x_i - x_j\|^2 / (2\sigma_i)^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / (2\sigma_i)^2)} \quad (\mathcal{P}_i)$$

where σ_i is an hyper-parameter depending on x_i .

How σ_i impacts on p_{ij} ?



Algorithm: Similarity and Probability Distribution

Definition (Similarity with Gaussian Kernel)

For each x_i we consider the similarity

$$\forall j \neq i, \quad p_{ij} = \frac{\exp(-\|x_i - x_j\|^2 / (2\sigma_i)^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / (2\sigma_i)^2)} \quad (\mathcal{P}_i)$$

where σ_i is an hyper-parameter depending on x_i .

How σ_i impacts on p_{ij} ?

Similarity to (0,0) for sigma=1.0



Algorithm: Similarity and Probability Distribution

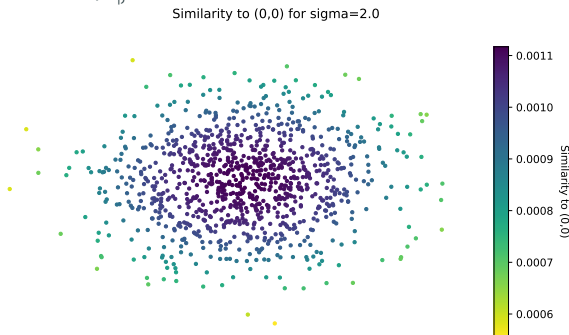
Definition (Similarity with Gaussian Kernel)

For each x_i we consider the similarity

$$\forall j \neq i, \quad p_{ij} = \frac{\exp(-\|x_i - x_j\|^2 / (2\sigma_i)^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / (2\sigma_i)^2)} \quad (\mathcal{P}_i)$$

where σ_i is an hyper-parameter depending on x_i .

How σ_i impacts on p_{ij} ?



Algorithm: Similarity and Probability Distribution

In the same manner we can define similarity in \mathcal{Y} .

$$\forall j \neq i, \quad q_{i|j} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq i} \exp(-\|y_i - y_k\|^2)} \quad (\mathcal{Q}_i)$$

Observation

\mathcal{P}_i and \mathcal{Q}_i are probability distributions for each i .

1. $0 \leq p_{i|j}, q_{i|j} \leq 1$ for each $j \neq i$
2. $\sum_{j \neq i} p_{i|j} = \sum_{j \neq i} q_{i|j} = 1$



Algorithm: Kullback-Leibler Divergence

Definition (K.L Divergence)

Let $\mathcal{P} = \{p_1, \dots, p_n\}$ and $\mathcal{Q} = \{q_1, \dots, q_n\}$ distributions

$$KL(\mathcal{P}, \mathcal{Q}) = \sum_i p_i \log_2 \left(\frac{p_i}{q_i} \right) \quad (1)$$

We can compare $\mathcal{P}_i, \mathcal{Q}_i$ for each i by taking

$$C(\mathcal{Y}) := \sum_i KL(\mathcal{P}_i, \mathcal{Q}_i) \quad (2)$$

Observation: The cost function C is differentiable in y_i

$$\frac{\partial C}{\partial y_i} = 2 \sum_{j \neq i} \left[\underbrace{(p_{ij} + p_{ji}) (y_i - y_j)}_{\text{Attractive}} \right] - \left[\underbrace{(q_{ij} + q_{ji}) (y_i - y_j)}_{\text{Repulsive}} \right] \quad (3)$$

Intepretation: C penalizes close x_i, x_j and far y_i, y_j .

Algorithm: Kullback-Leibler Divergence

Definition (K.L Divergence)

Let $\mathcal{P} = \{p_1, \dots, p_n\}$ and $\mathcal{Q} = \{q_1, \dots, q_n\}$ distributions

$$KL(\mathcal{P}, \mathcal{Q}) = \sum_i p_i \log_2 \left(\frac{p_i}{q_i} \right) \quad (1)$$

We can compare $\mathcal{P}_i, \mathcal{Q}_i$ for each i by taking

$$C(\mathcal{Y}) := \sum_i KL(\mathcal{P}_i, \mathcal{Q}_i) \quad (2)$$

Observation: The cost function C is differentiable in y_i

$$\frac{\partial C}{\partial y_i} = 2 \sum_{j \neq i} \left[\underbrace{(p_{ij} + p_{ji}) (y_i - y_j)}_{\text{Attractive}} \right] - \left[\underbrace{(q_{ij} + q_{ji}) (y_i - y_j)}_{\text{Repulsive}} \right] \quad (3)$$

Intepretation: C penalizes close x_i, x_j and far y_i, y_j .

SNE Algorithm

1. Choose $\Phi^{(0)}$ embedding, i.e choose $\{y_i^{(0)}\}$
2. Compute \mathcal{P}_i and \mathcal{Q}_i distributions.
3. Minimize $C(\mathcal{Y})$ through Gradient Descent with momentum

$$y_i^{(t+1)} = y_i^{(t)} - \eta \frac{\partial C}{\partial y_i} + \alpha (y_i^{(t)} - y_i^{(t-1)}), \quad \forall i$$

How to find σ_i ?

SNE Algorithm

1. Choose $\Phi^{(0)}$ embedding, i.e choose $\{y_i^{(0)}\}$
2. Compute \mathcal{P}_i and \mathcal{Q}_i distributions.
3. Minimize $C(\mathcal{Y})$ through Gradient Descent with momentum

$$y_i^{(t+1)} = y_i^{(t)} - \eta \frac{\partial C}{\partial y_i} + \alpha \left(y_i^{(t)} - y_i^{(t-1)} \right), \quad \forall i$$

How to find σ_i ?

Perplexity



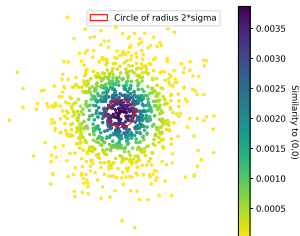
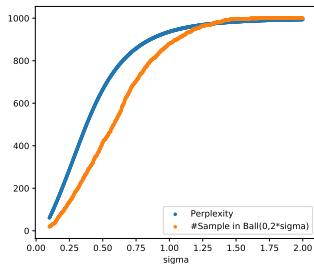
Perplexity: Shannon Entropy and Neighborhood

Let $\mathcal{P} = \{p_1, \dots, p_n\}$ be a distribution

$$\text{Shannon Entropy} \quad \mathbb{H}(\mathcal{P}) = - \sum_i p_i \log_2(p_i)$$

$$\text{Perplexity} \quad \text{Perp}(\mathcal{P}) = 2^{\mathbb{H}(\mathcal{P})}$$

How σ_i and Perplexity are related to each other?



Perplexity measures the number of samples in a neighborhood.^a

^aOriginal paper doesn't provide a proof, let us take it as an intuition.

Perplexity: Derivation of σ

Observation

Perplexity of \mathcal{P}_i is continuous monotonic in σ_i .

Binary Search: Find σ_i such that $Perp(\mathcal{P}_i(\sigma_i)) = K$

$\sigma^{(l)}$ such that $Perp(\mathcal{P}_i(\sigma^{(l)})) < K$;

$\sigma^{(r)}$ such that $Perp(\mathcal{P}_i(\sigma^{(r)})) > K$;

while $|\sigma^{(r)} - \sigma^{(l)}| < \varepsilon$ **do**

$\bar{\sigma} \leftarrow (\sigma^{(l)} + \sigma^{(r)})/2$;

$p \leftarrow Perp(\mathcal{P}_i(\bar{\sigma}))$;

if $p < K$ **then**

$\sigma^{(l)} \leftarrow \bar{\sigma}$;

else

$\sigma^{(r)} \leftarrow \bar{\sigma}$;

end

end

return $\bar{\sigma}$

t-SNE is t+SNE



t-SNE: Symmetric + Student

In order to achieve better results Maaten & Hinton modified SNE by

1. Using Student t-distribution for \mathcal{Y}

$$\forall i, \quad q_{i|j} = \frac{(1 + \|y_i - y_j\|)^{-1}}{\sum_{k \neq i} (1 + \|y_i - y_k\|)^{-1}}, \quad \forall j \neq i$$

2. Using symmetric versions of $p_{i|j}$, $q_{i|j}$ by defining

$$p_{ij} = (p_{i|j} + p_{j|i}) / (2n), \quad q_{ij} = (q_{i|j} + q_{j|i}) / (2n)$$

so that $\mathcal{P} = \{p_{ij}\}$ and $\mathcal{Q} = \{q_{ij}\}$ are joint probabilities.

3. The cost function becomes

$$C(\mathcal{Y}) = KL(\mathcal{P}, \mathcal{Q})$$

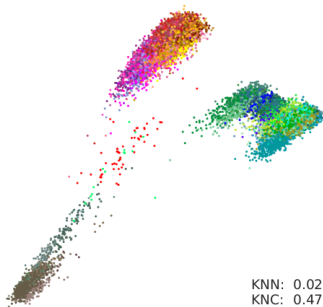
Examples



t-SNE on cortex's cells

b

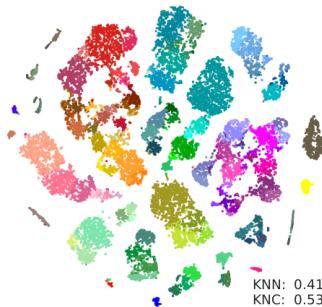
PCA



KNN: 0.02
KNC: 0.47
CPD: 0.91

c

Default t-SNE
(perplexity 30, random init., $\eta = 200$)



KNN: 0.41
KNC: 0.53
CPD: 0.23

Image taken from Kobak-Barens paper.

An hand made hierarchical clustering of cortex's cell is visualized through t-SNE. PCA is able to find the main three clusters but not to distinguish the sub-clusters.

t-SNE on cortex' cells: Metrics Evaluations

- **KNN:** local metric. For each samples $x_i \in \mathcal{X}$ and its image $y_i \in \mathcal{Y}$ we take the first k-closest points K_i and H_i and we evaluate

$$KNN := \frac{1}{|\mathcal{X}|} \sum_i \frac{|H_i \cap \Phi(K_i)|}{k}$$

- **KNC:** Cluster metric. The same as KNN but applied to c_1, \dots, c_s cluster's centroids.
- **CPD:** Global metric. Defined as the Spearman correlation.

These metrics in the figure 3 shows that t-SNE performs better in preserving local structures and worst in preserving the global structure of the data¹.

¹Kobak-Barens paper for more details.

t-SNE on MNIST

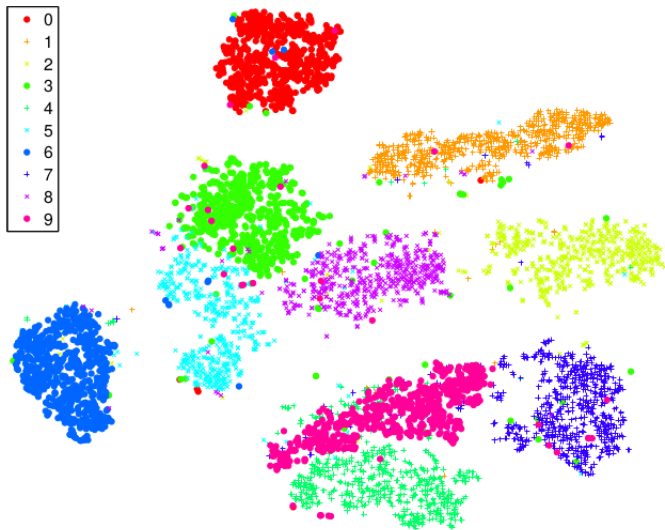


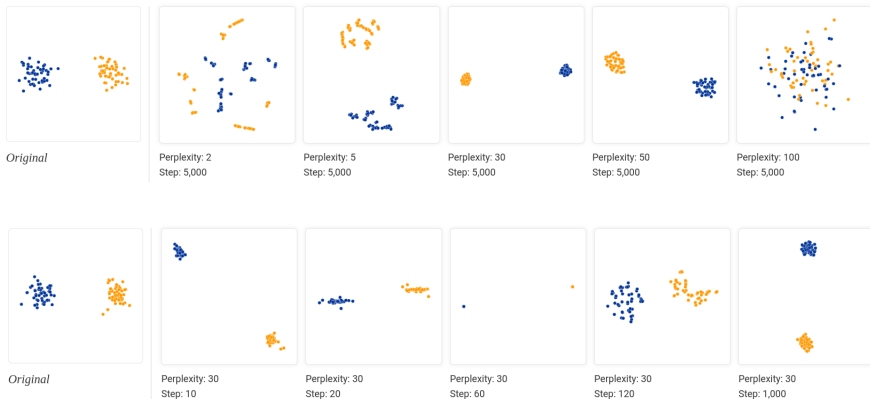
Image taken from Maaten - Hinton paper

t-SNE Warnings



How to use t-SNE²

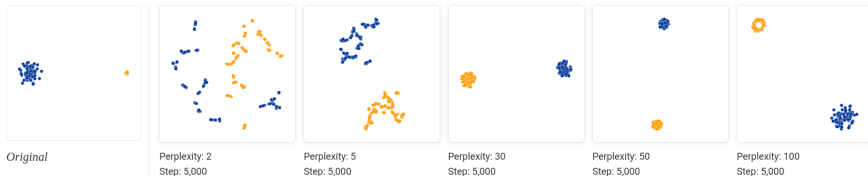
Hyper-Parameters really matter.



²In this section we will show the results in [Wattenberg - online paper](#).

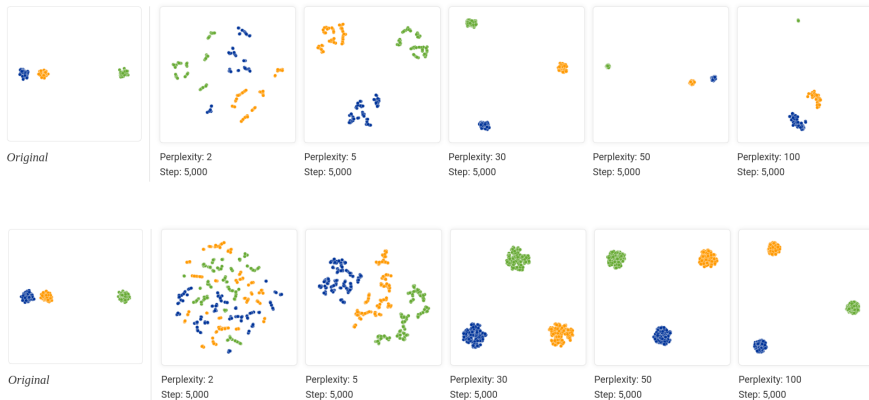
How to use t-SNE

Cluster sizes in a t-SNE plot has no meaning.



How to use t-SNE

There is not control of the distances between clusters.



Different initializations produce visualizations within clusters at different distances.

How to use t-SNE

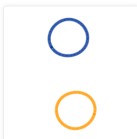
Topologists are not glad.



Original



Perplexity: 2
Step: 5,000



Perplexity: 5
Step: 5,000



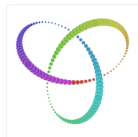
Perplexity: 30
Step: 5,000



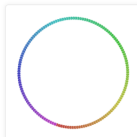
Perplexity: 50
Step: 5,000



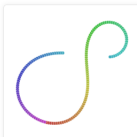
Perplexity: 100
Step: 5,000



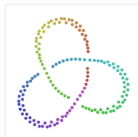
Original



Perplexity: 2
Step: 5,000



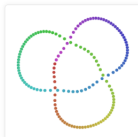
Perplexity: 5
Step: 5,000



Perplexity: 30
Step: 5,000



Perplexity: 50
Step: 5,000



Perplexity: 100
Step: 5,000

Thanks for the attention.