t-SNE – effective usage, avoiding pitfalls, and alternatives

Caroline Weis

December 5, 2019

Machine Learning and Computational Biology
Intradepartmental seminar

t-SNE in a nutshell

t-SNE algorithm

- t-SNE: t-distributed Stochastic Neighbor Embedding
- machine learning algorithm to visualize high-dimensional data

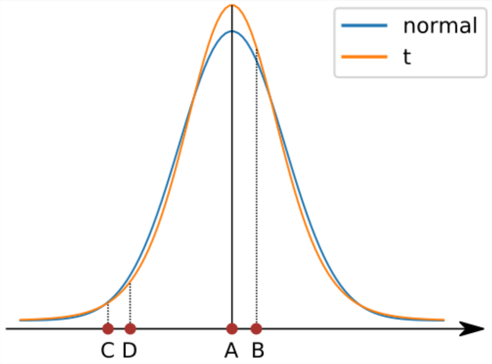
- Hinton et al developed SNE [1]
- van der Maaten introduced the t-distribution [2]

- non-linear dimensionality reduction

t-SNE conserves probabilities, <u>not</u> distances

t-SNE algorithm

conditional probability $p_{B|A}$:
probability that a point A would choose point B as its neighbor



t-SNE algorithm

- high-dimensional space $p_{j|i}$: normal distributed
- low-dimensional space $q_{j|i}$: t-distributed (with one degree of freedom)
- cost function: Kullback-Leibler divergence between $p_{j|i}$ and $q_{j|i}$

t-SNE minimizes the difference between two probability distributions

t-SNE is not deterministic

t-SNE parameters

cost function parameter

- **perplexity** $\approx$ number of neighbors considered / weigh emphasis on local or global aspects of your data [3]

  recommended [5,50]

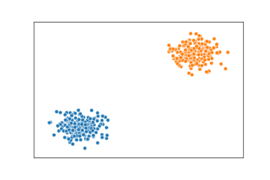
optimization parameters

- **number iterations** > 250
- **learning rate** recommended [10,1000]

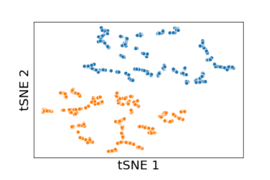
perplexity

perplexity

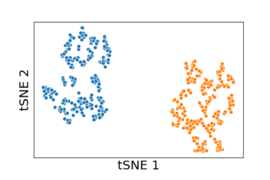
400 points



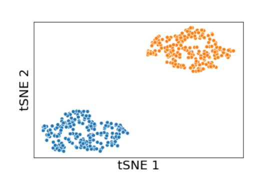
original



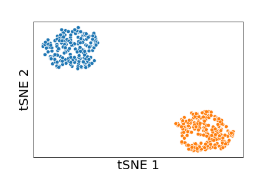
perplexity: 5



perplexity: 15



perplexity: 30

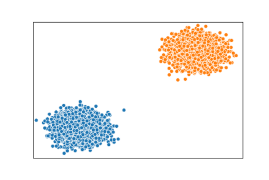


perplexity: 50

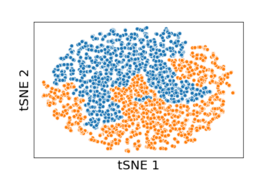
Within the recommended range, results look different with varying perplexities

perplexity and number of samples I

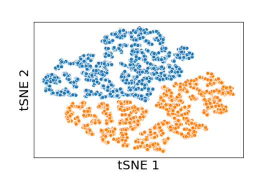
7000 points



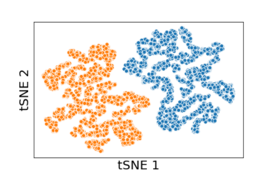
original



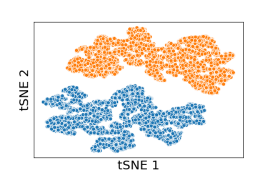
perplexity: 5



perplexity: 15



perplexity: 30

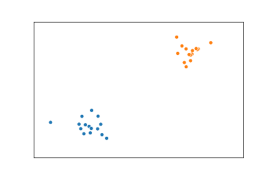


perplexity: 50

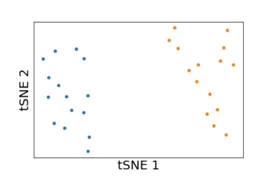
With more data points the perplexity generally needs to increase to obtain the same results

perplexity and number of samples II

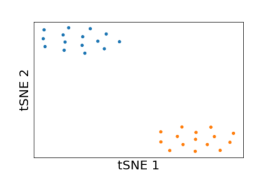
30 points



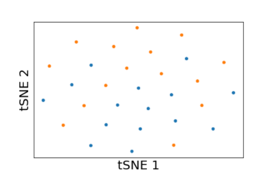
original



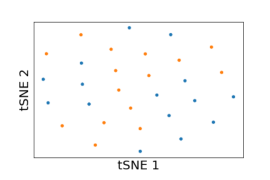
perplexity: 5



perplexity: 15



perplexity: 30



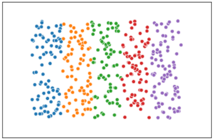
perplexity: 50

The perplexity should be smaller than the number of points - t-SNE can give unexpected behavior otherwise

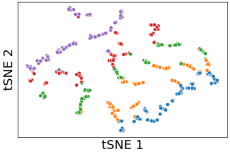
t-SNE behaviour

random noise

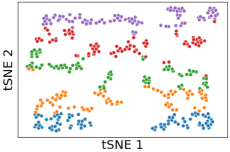
400 points



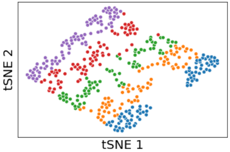
original



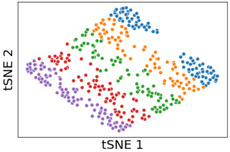
perplexity: 5



perplexity: 15



perplexity: 30

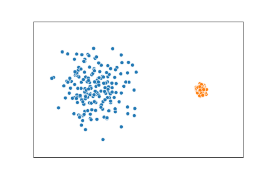


perplexity: 50

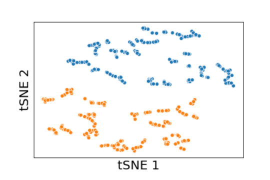
Random noise doesn't always look random

cluster size

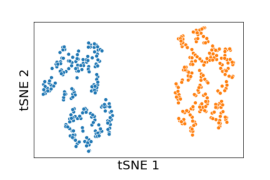
400 points



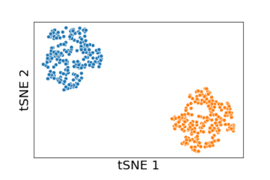
original



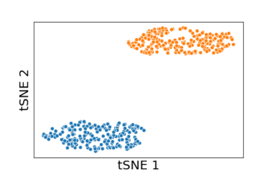
perplexity: 5



perplexity: 15



perplexity: 30

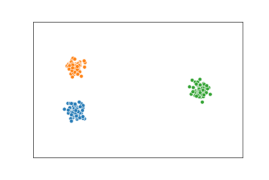


perplexity: 50

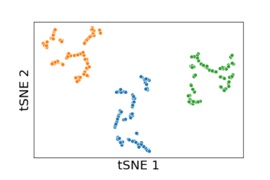
Cluster sizes are not conserved by t-SNE

cluster distances

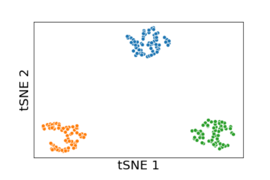
200 points



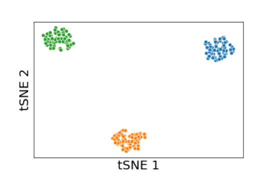
original



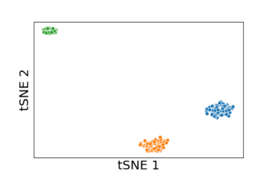
perplexity: 5



perplexity: 15



perplexity: 30



perplexity: **100**

Cluster distances are not always conserved by t-SNE

summary

- increase perplexity with number of points
- always choose perplexity $<$ number of points

- random noise might look non-random
- t-SNE does not conserve cluster sizes
- t-SNE does not always conserve cluster distances

effective use of t-SNE

- Run t-SNE for several suitable perplexity values.
- Don't "optimize" perplexity with Kullback-Leibler divergence!

- Use Kullback-Leibler divergence to find best optimization.

  (With constant dataset and perplexity, KL divergences are comparable. It is perfectly fine to run t-SNE several times, and select the solution with the lowest KL divergence)

- Don't overinterpret results!

- Try other dimensionality reduction methods as well.

alternative algorithms

low dimensional embedding algorithms

- Principal Component Analysis (**PCA**)

- Multi-Dimensional Scaling (**MDS**)
- Local Linear Embedding (LLE)
- Isomap
- Spectral Embedding

slides and code on GitHub

*https://github.com/***cvweis***/2019-12-tSNE-intradepartmental-seminar*

- slides
- code to reproduce all plots (Jupyter Notebooks and Python code)

references

- Geoffrey E Hinton and Sam T. Roweis.
  Stochastic neighbor embedding.
  pages 857–864, 2003.
- Laurens van der Maaten and Geoffrey Hinton.
  Visualizing data using t-SNE.
  volume 9, pages 2579–2605. 2008.
- Martin Wattenberg, Fernanda Viégas, and Ian Johnson.
  How to use t-sne effectively.
  Distill, 2016.

Questions?

formulars

- high-dimensional space: $p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_i - x_k\|^2 / 2\sigma_i^2\right)}$
- low-dimensional space: $q_{j|i} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|y_k - y_l\|^2\right)^{-1}}$
- cost function Kullback-Leibler divergence
  $C = \sum_i KL\left(P_i \| Q_i\right) = \sum_i \sum_j p_{ji} \log \frac{p_{j|i}}{q_{j|i}}$