INFORMATIONSFABRIK

DATEN VERSTEHEN, ENTSCHEIDUNGEN TREFFEN

t-SNE (t-Distributed Stochastic Neighbor Embedding)

Short Introduction, Tobias Rippel

Data Science Meet-Up Münster, 06.07.2017

t-SNE (t-Distributed Stochastic Neighbor Embedding)

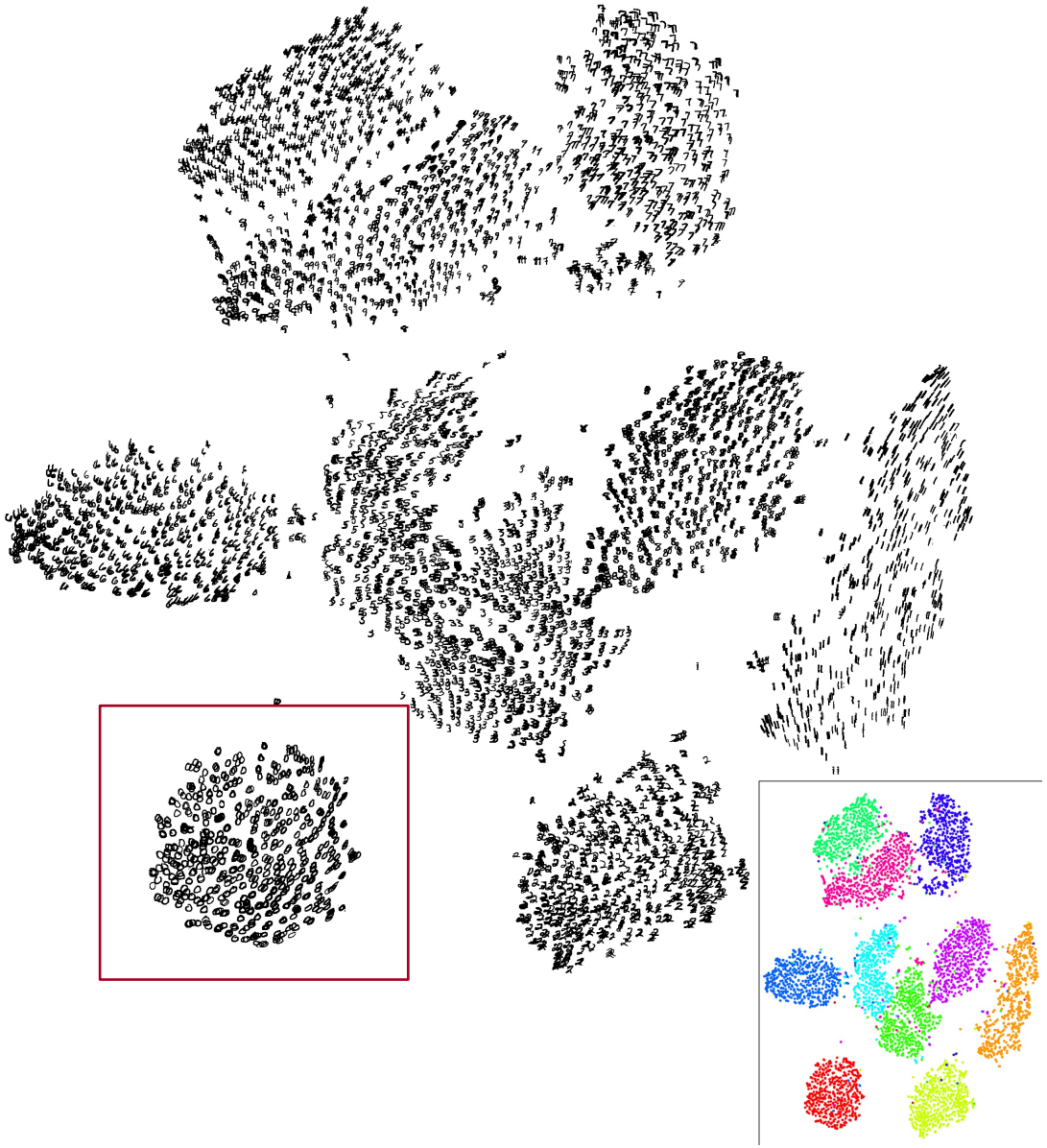
Overview

- What is t-SNE?
  - Unsupervised learning technique
  - Dimensionality reduction (mostly into 2 dimensional space)

  → Visualizing multidimensional/ high-dimensional data

t-SNE (t-Distributed Stochastic Neighbor Embedding)

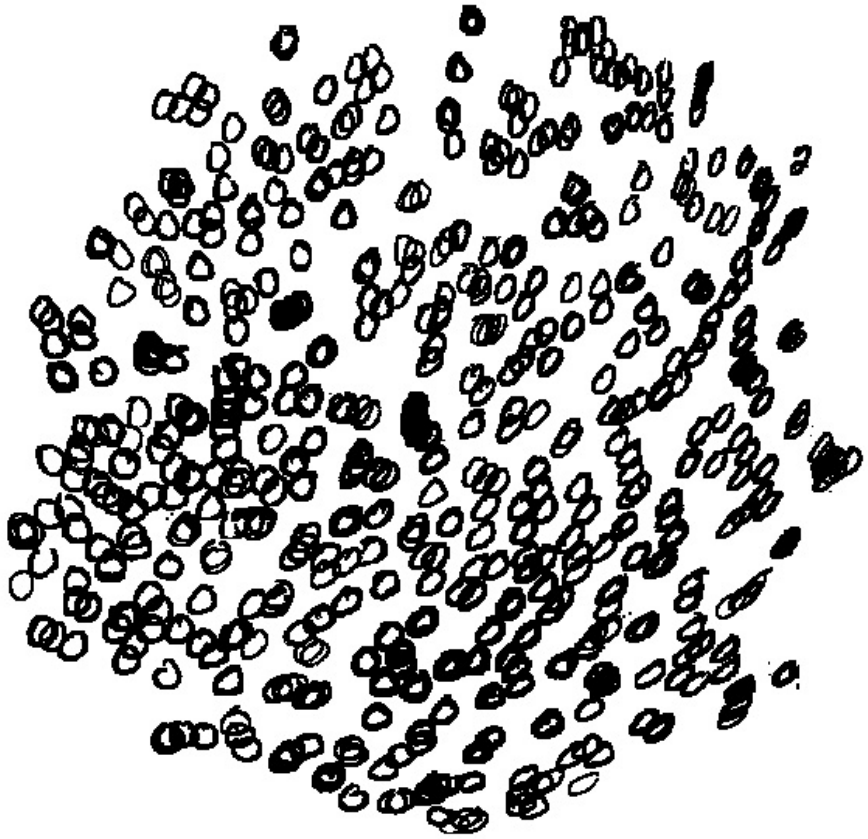
MNIST Example



Source: https://lvdmaaten.github.io/tsne/

t-SNE (t-Distributed Stochastic Neighbor Embedding)

MNIST Example

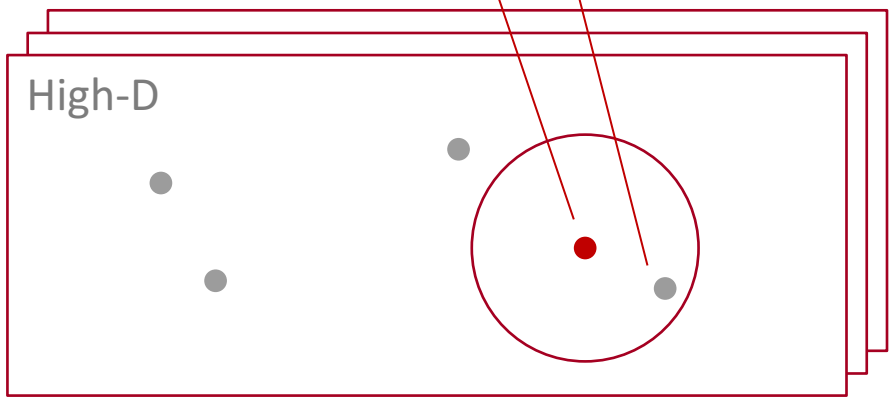


Source: https://lvdmaaten.github.io/tsne/

t-SNE (t-Distributed Stochastic Neighbor Embedding)

Algorithm Introduction

- Goal: Minimizing the discrepancy between similarities in the original data and similarities in the map
- Similarity in the original data: $p_{ij} = \frac{exp(-\|x_i - x_j\|^2 / 2\sigma^2)}{\sum_k \sum_{l \neq k} exp(-\|x_k - x_l\|^2 / 2\sigma^2)}$



t-SNE (t-Distributed Stochastic Neighbor Embedding)

Algorithm Introduction

- Goal: Minimizing the discrepancy between similarities in the original data and similarities in the map
- Similarity in the original data: $p_{ij} = \frac{exp(-\|x_i - x_j\|^2 / 2\sigma^2)}{\sum_k \sum_{l \neq k} exp(-\|x_k - x_l\|^2 / 2\sigma^2)}$
- Similarity in the map: $q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \|y_k - y_t\|^2)^{-1}}$ ← t-distribution
- Objective Function: Kullback-Leibler divergence $KL(P\|Q) = \sum_i \sum_{j \neq i} p_{ij} * log\left(\frac{p_{ij}}{q_{ij}}\right)$

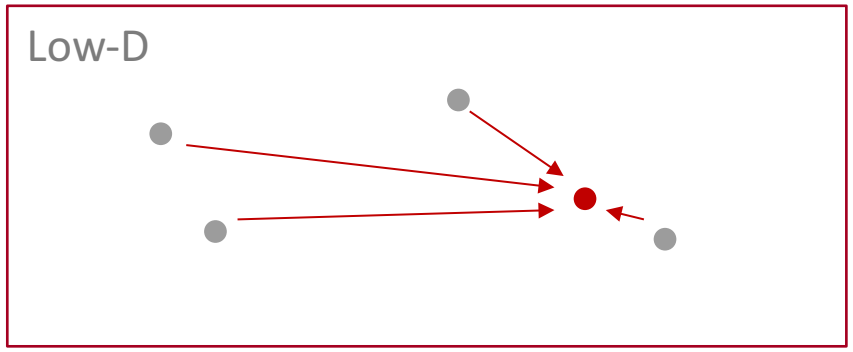
t-SNE (t-Distributed Stochastic Neighbor Embedding)

Algorithm Introduction

- **The embedding/training process**
  1. Initially calculate the similarity in the original data (p_{ij})
  2. Create a random generated initial mapping
  3. Loop x times
     a. Calculate the similarity in the mapping (q_{ij})
     b. Calculate the t-SNE gradient from the Kullback-Leibler divergence objective function
     c. Update the map according to the gradient of each point

$$\frac{\partial C}{\partial y_i} = 4\sum_{j \neq i}(p_{ij} - q_{ij})\left(1 + \|y_i - y_j\|^2\right)^{-1}(y_i - y_j)$$

excertion/compression



t-SNE: MNIST Example

Source: https://github.com/oreillymedia/t-SNE-tutorial/blob/master/images/animation.gif

t-SNE: Summary

- t-SNE is an unsupervised learning technique
- Used for dimensionality reduction -> visualize the data (can then be used e.g. in data exploring or to check if feature engineering is representing the expected properties of the data well)
- t-SNE preserves global + local structure of the data

Try it out yourself! Available in many Data Science languages + tools

Tutorials + Sources

- Google Talk (~45 min) by Laurens van der Maaten:
  https://www.youtube.com/watch?v=RJVL80Gg3lA&list=UUtXKDgv1AVoG88PLl8nGXmw

  https://www.analyticsvidhya.com/blog/2017/01/t-sne-implementation-r-python/
  http://distill.pub/2016/misread-tsne/#perplexity=32&epsilon=4&demo=1&demoParams=100,2
  https://github.com/oreillymedia/t-SNE-tutorial