Data Visualization

Giri Iyengar

Cornell University

gi43@cornell.edu

March 26, 2018

Overview

1. Overview
2. t-SNE

Overview

1. Overview
2. t-SNE

Agenda

- Are the models working as expected?
- Do the metrics make sense?
- Visualizing multi-dimensional data
- Trying to understand DL models

Plotting Residuals

Residual Errors

In a good model, it is expected that the errors that the model makes will not have any *systematic* nature to them. That is, the errors should be essentially *random*.

Plotting Residuals: No systematic errors in prediction

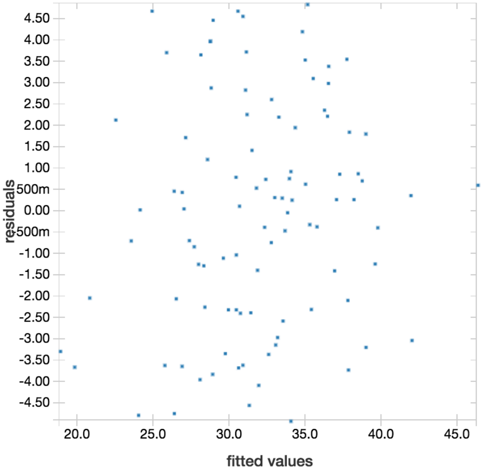


Figure: Source - Databricks blog

Plotting Residuals: Systematic errors in prediction

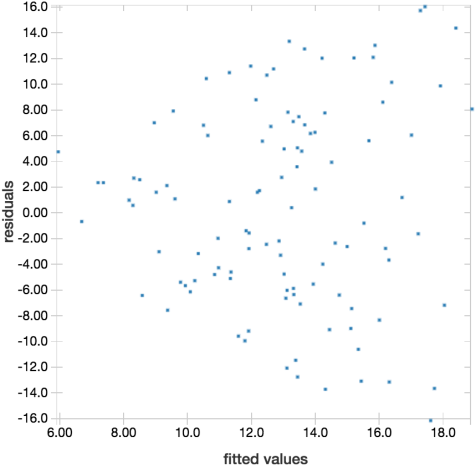


Figure: Source - Databricks blog

Visualizing KMeans fit

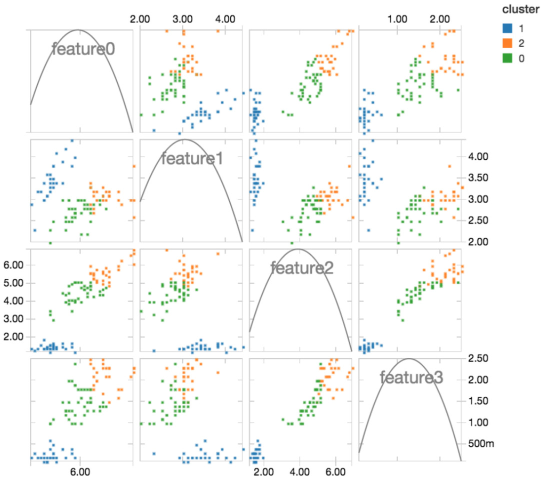


Figure: Source - Databricks blog

ROC Curve examples

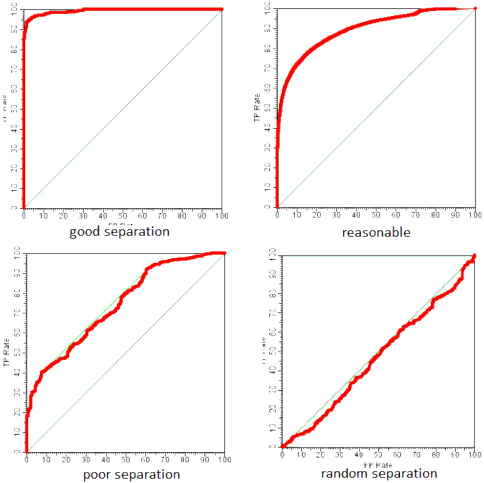


Figure: Source - MLWiki

Overview

1. Overview
2. t-SNE

t-distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a (prize-winning) technique for dimensionality reduction that is particularly well suited for the visualization of high-dimensional datasets. The technique can be implemented via Barnes-Hut approximations, allowing it to be applied on large real-world datasets. It has been applied on data sets with up to 30 million examples [1].

Visualizing/reducing dimensions of high-dimensional data

- PCA - preserves large distances
- ISOMAP - changes similarity function and then applies PCA
- Locally linear embedding

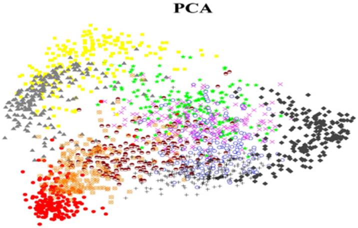


Figure: Source: Xiaofei He

ISOMAP

- ISOMAP reduces dimensions non-linearly
- Related to kernel PCA
- Instead of Euclidean distance, use a geodesic / manifold distance

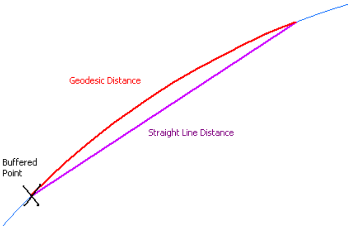


Figure: Source: ESRI

Locally Linear Embedding

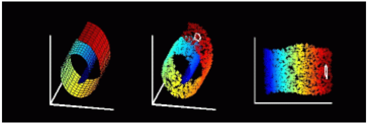


Figure: Source: Roweis and Saul

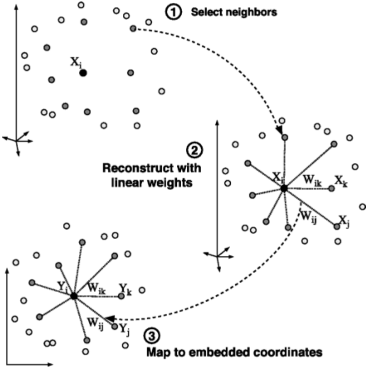


Figure: Source: Roweis and Saul

SNE Algorithm

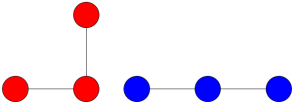
- Similar to LLE but use probabilities instead of distances
- Compute $p_{j|i}$, conditional probability that x_i would pick x_j as neighbor under a locally modeled pdf
- Formally $p_{j|i} = \frac{exp(-\frac{|x_i - x_j|^2}{2\sigma_i^2})}{\sum_{k \neq i} exp(-\frac{|x_i - x_k|^2}{2\sigma_i^2})}$
- Define $q_{j|i} = \frac{exp(-|y_i - y_j|^2)}{\sum_{k \neq i} exp(-|y_k - y_j|^2)}$
- Define $C = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$, the KL Divergence
- Perform gradient descent to minimize C

SNE Algorithm: KL Divergence

- KL Divergence is asymmetric
- Nearby points (large $p_{j|i}$) weigh more than far-away points (low $p_{j|i}$)
- Objective function strongly favors preserving distances between nearby points over far away points

t-SNE Algorithm

- Use $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$ instead
- Use $q_{ij} = \frac{(1+|y_i - y_j|^2)^{-1}}{\sum_{k \neq i}(1+|y_i - y_k|^2)^{-1}}$, the Student-t distribution
- Student-t distribution is heavy-tailed. Allows for a small probability for far-away points, forcing them to move further away in low-dim space



t-SNE: Barnes-Hut approximation

- As formulated, $O(n^2)$ algorithm

t-SNE: Barnes-Hut approximation

- As formulated, $O(n^2)$ algorithm
- Doesn't work for really large datasets

t-SNE: Barnes-Hut approximation

- As formulated, $O(n^2)$ algorithm
- Doesn't work for really large datasets
- What can we do to reduce the cost?

t-SNE: Barnes-Hut approximation

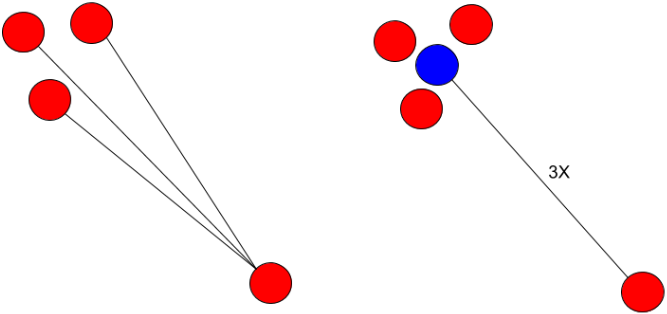
- As formulated, $O(n^2)$ algorithm
- Doesn't work for really large datasets
- What can we do to reduce the cost?
- **Insight**: Can we approximate roughly equally distance far away points?

t-SNE with Barnes-Hut approximation

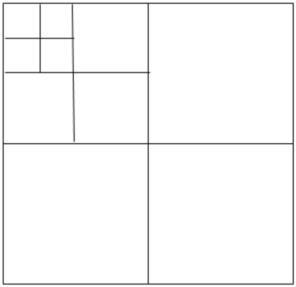
Barnes-Hut approximation

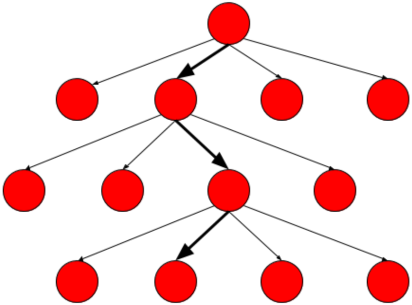
Barnes-Hut is an approximation algorith used in Astronomy to simulate n-body problem. It uses a octree representation to model bodies in a 3-D space and recursively groups them in this octree. In 2D, we replace octree with quadtree. Converts the n^2 search into an $n \log n$ search.

Barnes-Hut Approximation



Quadtree representation





t-SNE examples: MNIST Digits

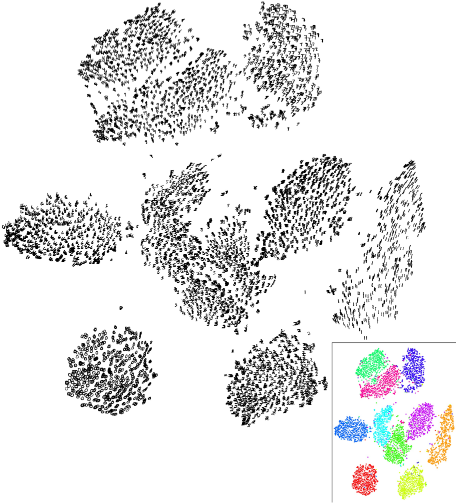


Figure: Source: Laurens van der Maaten

t-SNE examples: Netflix movies

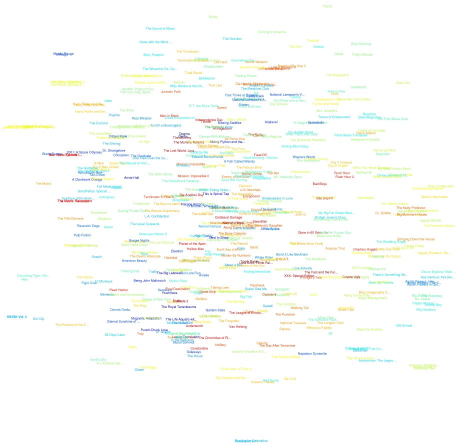


Figure: Source: Laurens van der Maaten

t-SNE examples: Words

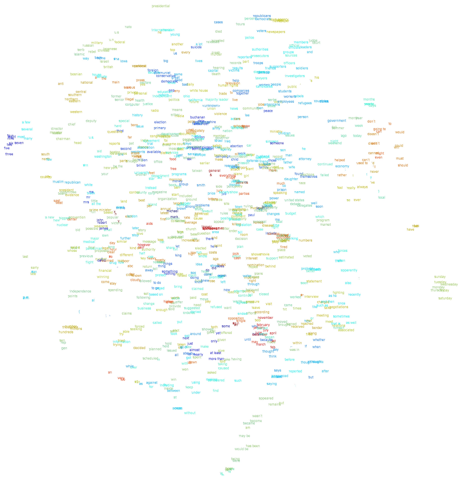


Figure: Source: Laurens van der Maaten

t-SNE Multiple Map extension

- Multiple word senses: e.g. (River, Bank, Bailout)
- In general, how do we deal with non-metric similarities?
- Extend $q_{ij} = \frac{\sum_m \pi_i^m \pi_j^m (1+|y_i^m - y_j^m|^2)^{-1}}{\sum_k \sum_{m'} \sum_{l \neq k} (1+|y_k^{m'} - y_l^{m'}|^2)^{-1}}$
- Now, you get multiple maps. Each map models a different similarity between words


рахмат
danke
謝謝
ngiyabonga
teşekkür ederim
Баярлалаа
спасибо
thank you
gracias
dank je
tapadh leat
bedankt
hvala
mauruuru
dziękuję
obrigado
mochchakkeram
go raibh maith agat
sukriya
kop khun krap
terima kasih
grazie
arigatô
takk
dakujem
мерси
merci
ευχαριστώ
감사합니다
ありがとう
хвала
sagolun
tanemirt
rahmet
dhanyavadagalu
shukriya
xiexie
diolch
enkosi
asante
manana