

Weekly Report 3 - t-SNE

Ganji Varshitha
AI20BTECH11009

Introduction

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a technique for dimensionality reduction that is particularly well suited for the visualization of high-dimensional datasets.

This is a part of manifold learning also known as non-linear dimensionality reduction. The algorithm was developed by Laurens van der Maaten and Geoffrey Hilton.

Algorithm

Algorithm 1: Simple version of t-Distributed Stochastic Neighbor Embedding.

Data: data set $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$,
cost function parameters: perplexity $Perp$,
optimization parameters: number of iterations T , learning rate η , momentum $\alpha(t)$.
Result: low-dimensional data representation $\mathcal{Y}^{(T)} = \{y_1, y_2, \dots, y_n\}$.
begin
 compute pairwise affinities $p_{j|i}$ with perplexity $Perp$ (using Equation 1)
 set $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$
 sample initial solution $\mathcal{Y}^{(0)} = \{y_1, y_2, \dots, y_n\}$ from $\mathcal{N}(0, 10^{-4}I)$
 for $t=1$ **to** T **do**
 compute low-dimensional affinities q_{ij} (using Equation 4)
 compute gradient $\frac{\delta C}{\delta \mathcal{Y}}$ (using Equation 5)
 set $\mathcal{Y}^{(t)} = \mathcal{Y}^{(t-1)} + \eta \frac{\delta C}{\delta \mathcal{Y}} + \alpha(t) (\mathcal{Y}^{(t-1)} - \mathcal{Y}^{(t-2)})$
 end
end

In contrast to reduce the high-dimensional dataset to low-dimensional data, we refer the low-dimensional data representation \mathcal{Y} as a bijective map, and to the low-dimensional representations y_i of individual datapoints as map points.

We try to achieve the same structure of high-dimensional data in the low-dimensional map i.e the points which are closer to a point are placed near that point in low dimensional space also.

The algorithm has 2 stages:

1. Convert distances to probabilities that data point x_i will choose data point x_j as its neighbour.

Conditional probability between two points is given by

$$p_{j|i} = \frac{\exp(-||x_i - x_j||^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-||x_i - x_k||^2 / 2\sigma_i^2)} \quad (1)$$

For t-SNE, we use symmetric SNE, i.e

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N} \text{ where } N \text{ is number of datapoints} \quad (2)$$

2. We need to evaluate the map from $\mathcal{X} \rightarrow \mathcal{Y}$

- We can compute conditional probability for map points using t-Student distribution with one degree of freedom.

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_k \sum_{k \neq l} (1 + ||y_k - y_l||^2)^{-1}} \quad (3)$$

- Since we want both the probabilities to be very close to each other, we minimise the kullback - Leiber divergence between the two distributions

$$KL(P||Q) = \sum_{i,j} p_{ij} \log \frac{p_{ij}}{q_{ij}}. \quad (4)$$

We achieve this using gradient descent stated in the algorithm above.