12 APRIL 2018 / SERIES

Dimension Reduction - t-SNE

This tutorial is from a 7 part series on Dimension Reduction:

1. Understanding Dimension Reduction with Principal Component Analysis (PCA)
2. Diving Deeper into Dimension Reduction with Independent Components Analysis (ICA)
3. Multi-Dimension Scaling (MDS)
4. LLE
5. **t-SNE**
6. IsoMap
7. Autoencoders

(A more mathematical notebook with code is available the github repo)

t-SNE is a new award-winning technique for dimension reduction and data visualization. t-SNE not only captures the local structure of the higher dimension but also preserves the global structures of the data like clusters. It has stunning ability to produce well-defined segregated clusters. t-SNE is based on stochastic neighbor embedding(SNE). t-SNE was developed to address some of the problems in SNE. So let's have a basic understanding of SNE.

by preserving the neighborhood structure of the dataset. A Gaussian probability distribution centered on each point is defined over all the potential neighbors of this point. SNE aims to minimize the difference in probability distribution in the higher dimension and lower dimension.

For each object, i and it's neighbor j , we compute a $P_{i|j}$ which reflects the probability that j is neighbor of i

$P_{i|j} = \exp(-d^2_{ij})\Sigma_{k\neq i}\exp(-d^2_{ij}))$ where d^2_{ij} is the dissimilarity between element i and j given as input or calculated from the dataset provided.

The dissimilarity between x_i and x_j can be calculated using the following formula

$d^2_{ij} = ||x_i - x_j||^2 / (2\sigma^2_i)$, where σ_i generally calculated through a binary search by equating the entropy of the distribution centered at x_i to perplexity which is chosen by hand. This method generates a probability matrix which is asymmetric.

Now, a random solution is chosen as the starting point for the low dimensional embedding. A probability distribution is defined on it in the same way as done above but with a constant $\sigma=0.5$ for all points.

SNE tries to minimize the difference between these two distributions. We can calculate the difference between two distributions using Kullback-Liebler divergence. For two discrete distirbution P and Q KL divergence is given by

$DKL(P||Q)=\Sigma_i P_i(P_i/Q_i)$.

SNE defines a cost function based of the difference between p_{ij} and q_{ij} which is given by

$C=\Sigma_i\Sigma_j(P_{ij}\log(P_{ij}/q_{ij}))$

While embedding the dataset in lower dimension, two kinds of error can occur, first neighbors are mapped as faraway points(p_{ij} is large and q_{ij} is small) and points which are far away mapped as neighbors(p_{ij} is small while q_{ij} is large).Look closely at the cost

large q_{ij} . SNE heavily penalizes if the neighbors are mapped faraway from each other.

Some of the shortcomings of SNE approach are asymmetric probability matrix P, crowding problem. As pointed out earlier the probability matrix P is asymmetric. Suppose a point X_i is far away from other points, it's P_{ij} will be very small for all j. So, It will have little effect on the cost function and embedding it correctly in the lower dimension will be hard.

Any n-dimensional Euclidean space can have an object with n+1 or less equidistant vertices not more than that. Now, when the intrinsic dimension of a dataset is high say 20, and we are reducing its dimensions from 100 to 2 or 3 our solution will be affected by crowding problem. The amount of space available to map close points in 10 or 15 dimensions will always be greater than the space available in 2 or 3 dimensions. In order to map close points properly, moderately distant points will be pushed too far. This will eat the gaps in original clusters and it will look like a single giant cluster.

We need to brush up few more topics before we move to t-SNE.

Student-t distribution -- Student-t distribution is a continuous symmetric probability distribution function with heavy tails. It has only one parameter degree of freedom. As the degree of freedom increases, it approaches the normal distribution function. When degree of freedom =1, it takes the form of Cauchy distribution function and its probability density function is given by $f(t)=1/\pi(1+t2)$

Entropy: Entrophy is measure of the average information contained in a data. For a variable x with pdf p(x), it is given by

$$H(x) = -\Sigma_i(p(x_i) \times log_2(p(x_i)))$$

Perpexility: In information theory, perplexity measures how good a probability distribution predicts a sample. A low perplexity indicates that distribution function is good at predicting sample. It

$Perpx(x)=2^{H(x)}$, where $H(x)$ is the entropy of the distribution.

t-SNE

t-SNE differs from SNE in two ways, first it uses a student-t distribution to measure the similarity between points Y_i and Y_j in the lower dimension,secondly for the higher dimension it uses symmetric probability distribution such that $P_{ji}=P_{ij}$.

Steps of t-SNE algorithm:

1. compute pairwise similarity P_{ij} for every i and j.
2. make P_{ij} symmetric.
3. choose a random solution Y_0
4. While not done:
   compute pairwise similairites for Y_i
   compute the gradient
   update the solution
   if i>max_iter break
   else
   i = i+1

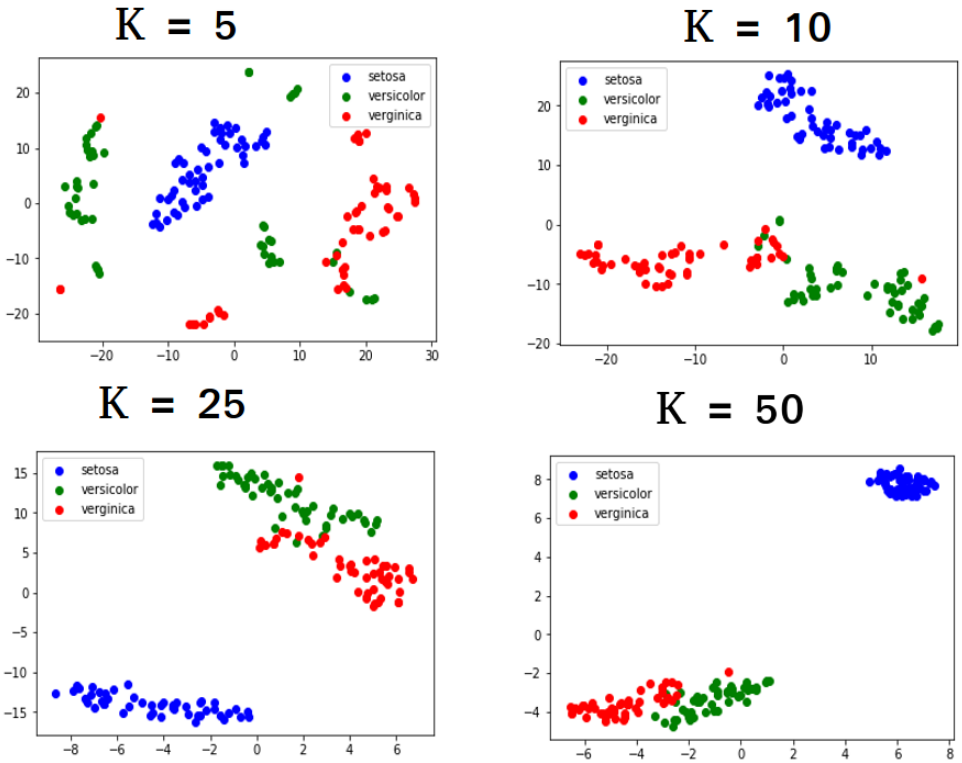
Computing probability distribution:

For computing pairwise similarities we need to know the variance σ_i for the Gaussian centered at x_i. One might think why not set a single value of σ_i for every x_i. The density of data is likely to vary, we need smaller σ_i for places with higher densities and bigger σ_i for places where points are far away. The entropy of the Gaussian distribution centered at x_i increases as σ_i increases. To get the σ_i we need to perform a binary search such that perplexity of the Gaussian distribution centered at x_i is equal to the perplexity specified by the user.

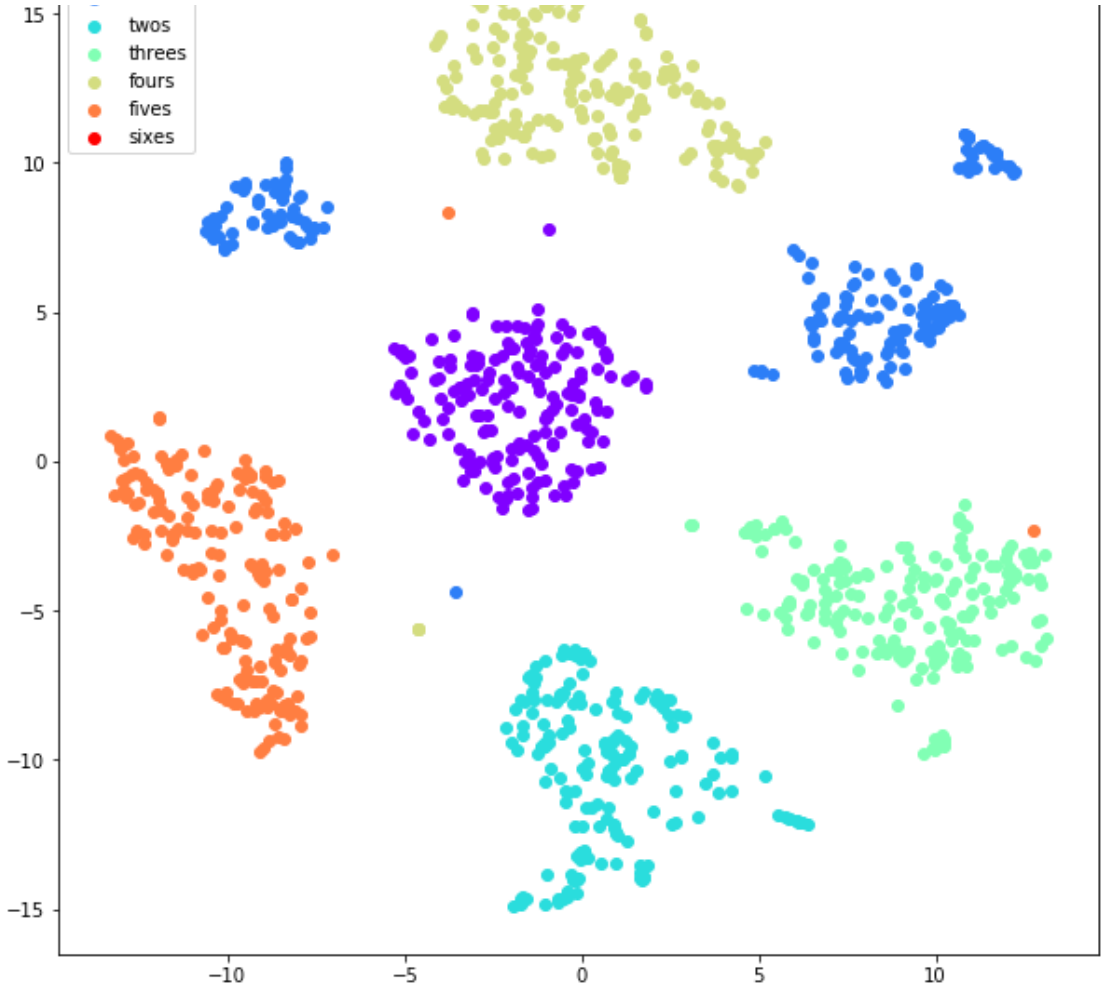
think of perplexity as a measure of the number of neighbors.

Impact of parameters on embedding:

For t-SNE to be meaningful we have to choose right value of perplexity. Perplexity balances the local and global aspects of the dataset. A Very high value will lead to the merging of clusters into a single big cluster and low will produce many close small clusters which will be meaningless. Images below show the effect of perplexity on t-SNE on iris dataset.



When K(number of neighbors) = 5 t-SNE produces many small clusters. This will create problems when number of classes is high. As the number of neighbors increases the clusters from same classes merge. At K=25 and K=50 we have well-defined clusters for few classes. Also, the clusters become denser as the K increases.
A plot of a subset of MNIST dataset after t-SNE embedding.



t-SNE produces a well-defined and separate cluster for each of the digits.

Drawbacks of t-SNE

Problems with t-SNE arise when intrinsic dimensions are higher i.e. more than 2-3 dimensions. t-SNE has the tendency to get stuck in local optima like other gradient descent based algorithms. The basic t-SNE algorithm is slow due to nearest neighbor search queries.

Conclusion: We talked about another dimension reduction and visualization method **t-SNE** through this post. In the beginning, we discussed important topics related to t-SNE. Afterwards, a

Barnes-Hut t-SNE would be a good starting point.

In next post, we will learn about Isomap

Ashwini Kumar Pal
Read more posts by this author.

Read More

— Hello Paperspace —
Series

Intro to optimization in deep learning: Momentum, RMSProp and Adam

Dimension Reduction - Autoencoders

Dimension Reduction - IsoMap

See all 7 posts →

SERIES

Dimension Reduction - IsoMap

This tutorial is from a 7 part series on Dimension Reduction: Understanding Dimension Reduction with Principal Component Analysis (PCA) Diving Deeper into Dimension Reduction with Independent Components Analysis (ICA) Multi-Dimension Scaling (MDS) LLE

6 MIN READ

SERIES

Dimension Reduction - LLE

This tutorial is from a 7 part series on Dimension Reduction: Understanding Dimension Reduction with Principal Component Analysis (PCA) Diving Deeper into Dimension Reduction with Independent Components Analysis (ICA) Multi-Dimension Scaling (MDS) LLE