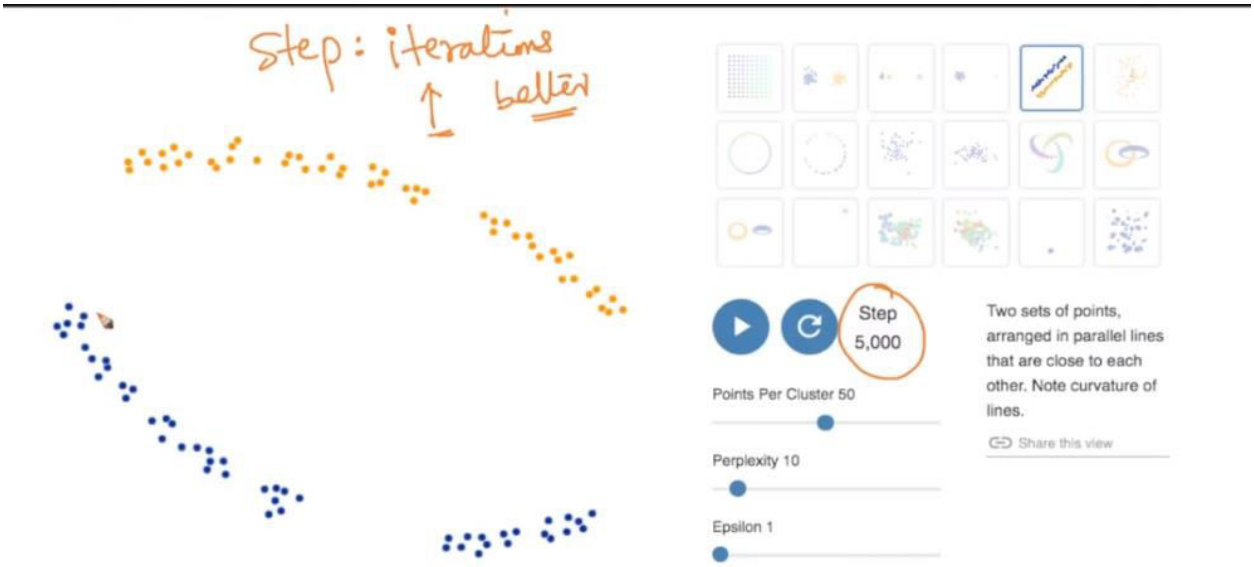
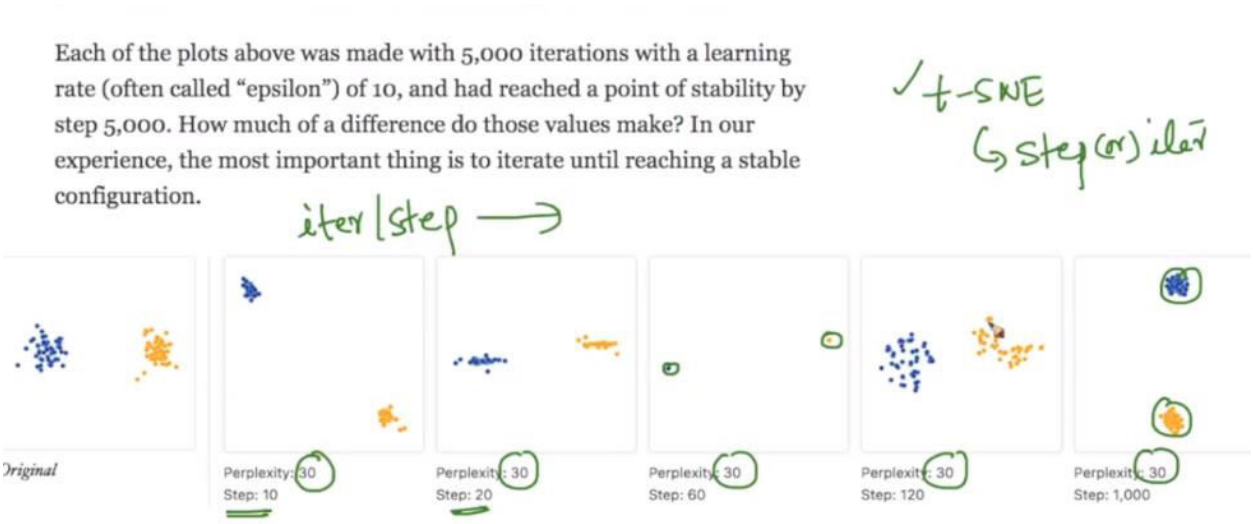
Step: Step is the no of iteration the embedding is perform.

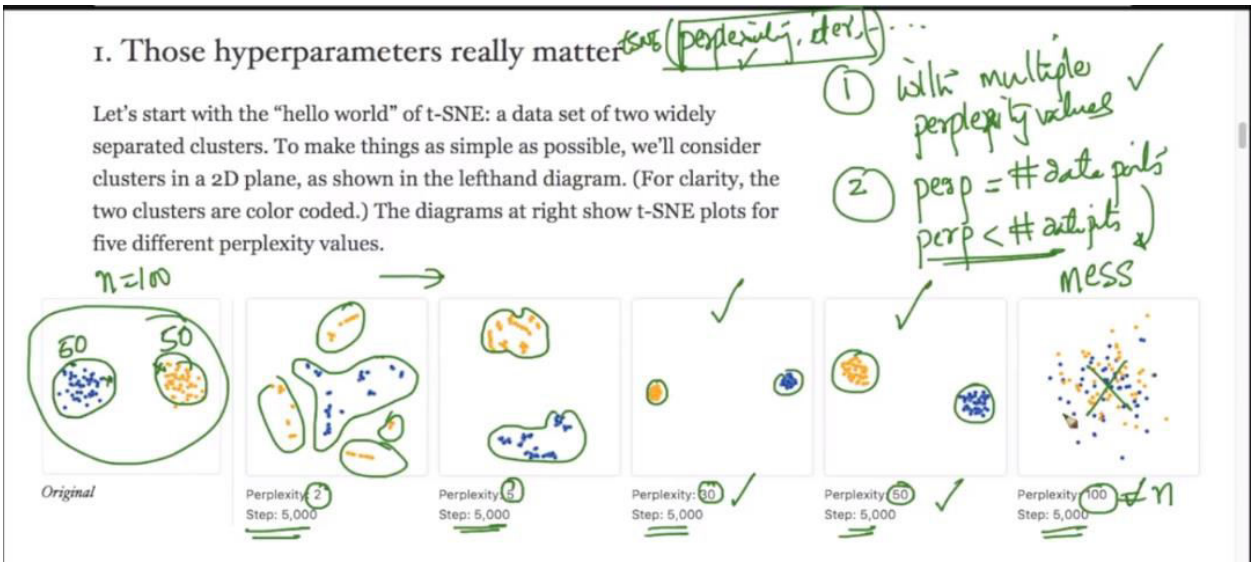
T-SNE tries to process all the data in iterations and eventually it wants to reach a stage where the clusters are no more moving. At every iteration, T-SNE tries to move the points, find and improve the embedding and preserve the neighborhoods as many as possible. For each iteration, we get a better solution. We should keep iterating till we get a stable configuration.



In below image we can see how it's changing results are changing with increase in step, and we are getting better results as step increases.



Perplexity: The perplexity can be interpreted as a smooth measure of the effective number of neighbors. The performance of tSNE is fairly robust to changes in the perplexity, and typical values are between 5 and 50. You can think Perplexity as the number of neighbours it considers while performing the algorithm



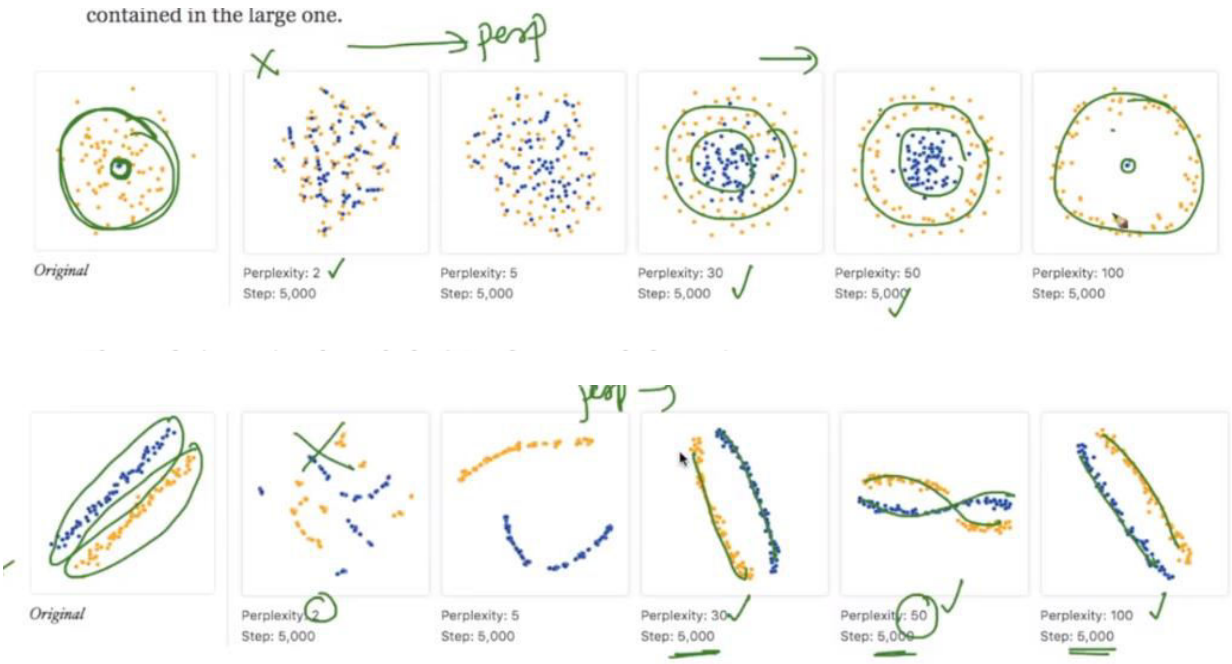
From above image we can say that we should run t-SNE with different no of perplexity.

With lower perplexity and much larger perplexity it will not generate good results, hence we can say that:

- Perplexity should not be equal to no of data points.
- And Perplexity should be less than no of datapoint.

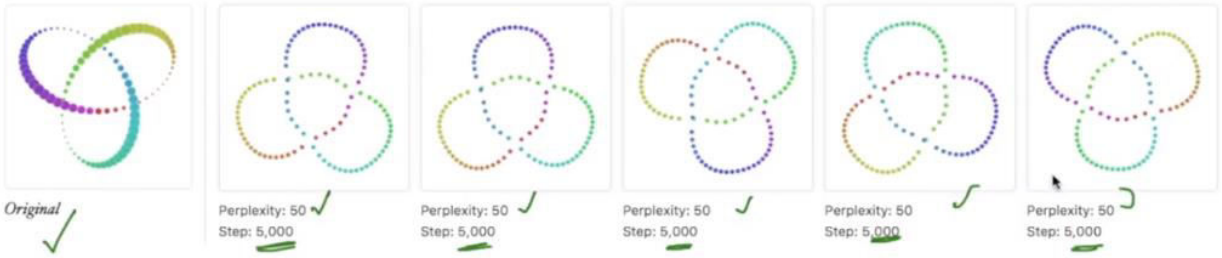
Importance of different perplexity numbers:

As you can see in below figure in original data blue cluster is within orange, but with lower cluster like 2 and 5, it's mixing, and as we are increasing perplexity we are getting results according to original dataset.



What does stochastic mean in t-SNE:

Here stochastic means probabilistic, t-SNE is not a deterministic algorithm (deterministic algorithm is that which generates same results on any time they run), because every time we run t-SNE with same parameters it generates slightly different result. As you can see in below figure it runs 5 times with constant perplexity and step, and every time it generates result with same shape but with some different rotations.

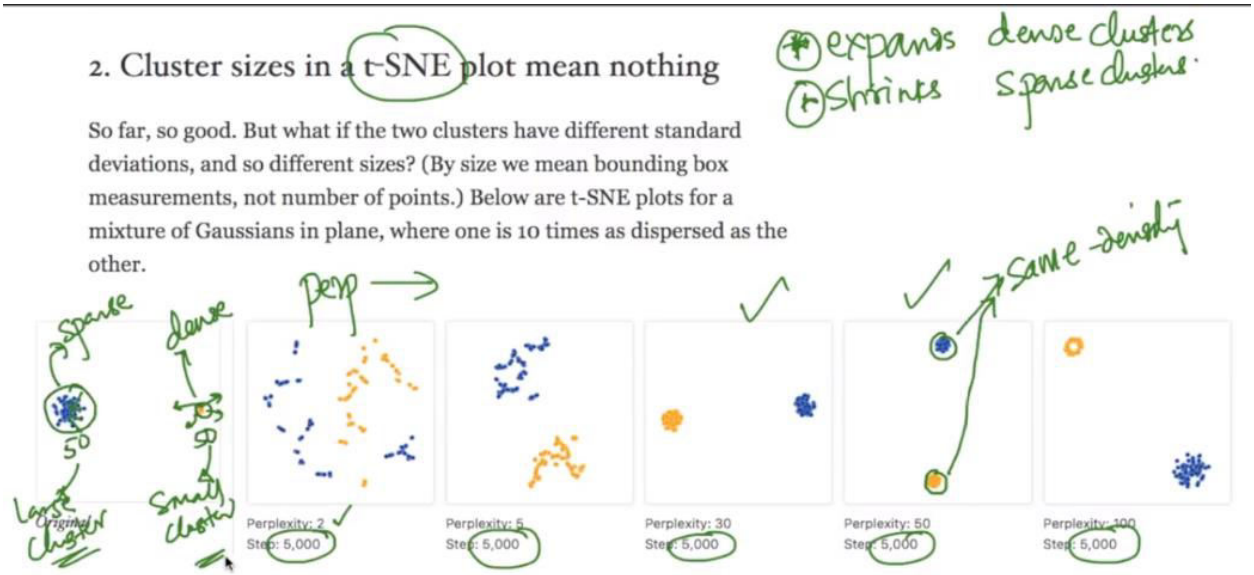


Cluster sizes in a t-SNE plot means nothing:

Means the size of obtained clusters does not say anything about data, because:

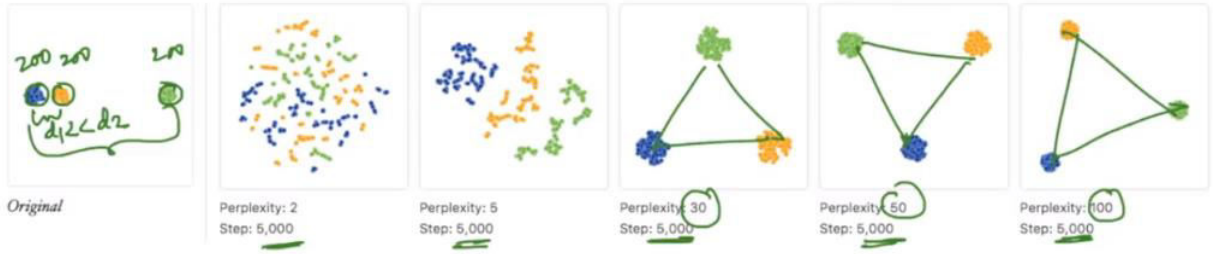
- t-SNE expands dense clusters
- t-SNE shrinks sparse clusters.

As in below fig we can see that there are 2 clusters in original data where orange cluster is much dense and blue cluster is sparse, but after plotting we can see that t-SNE is generating results with approximately same density.



t-SNE doesn't preserve distance between clusters:

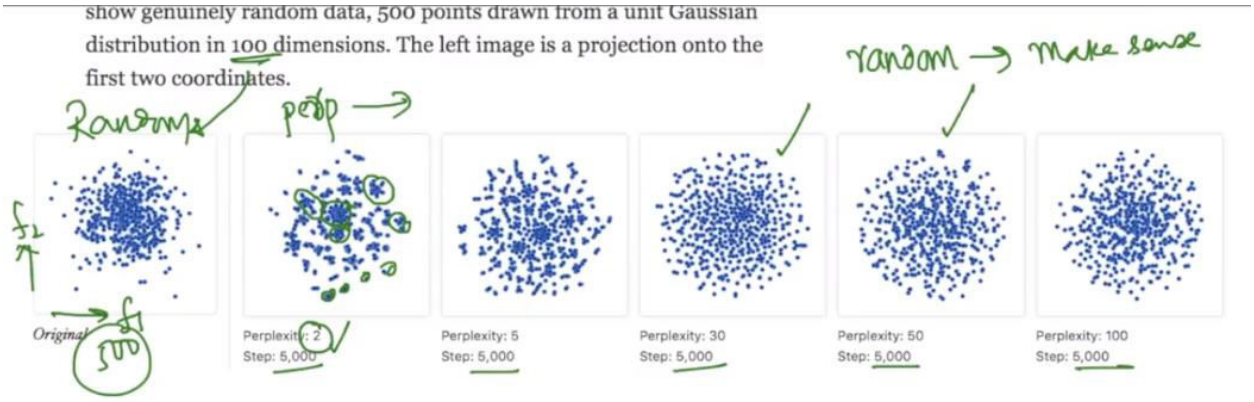
As we can see in below figure that distance between blue cluster and orange cluster is very small, but after t-SNE, this will make comparatively larger distance between them hence t-SNE doesn't preserve distance between clusters.



t-SNE for Random data:

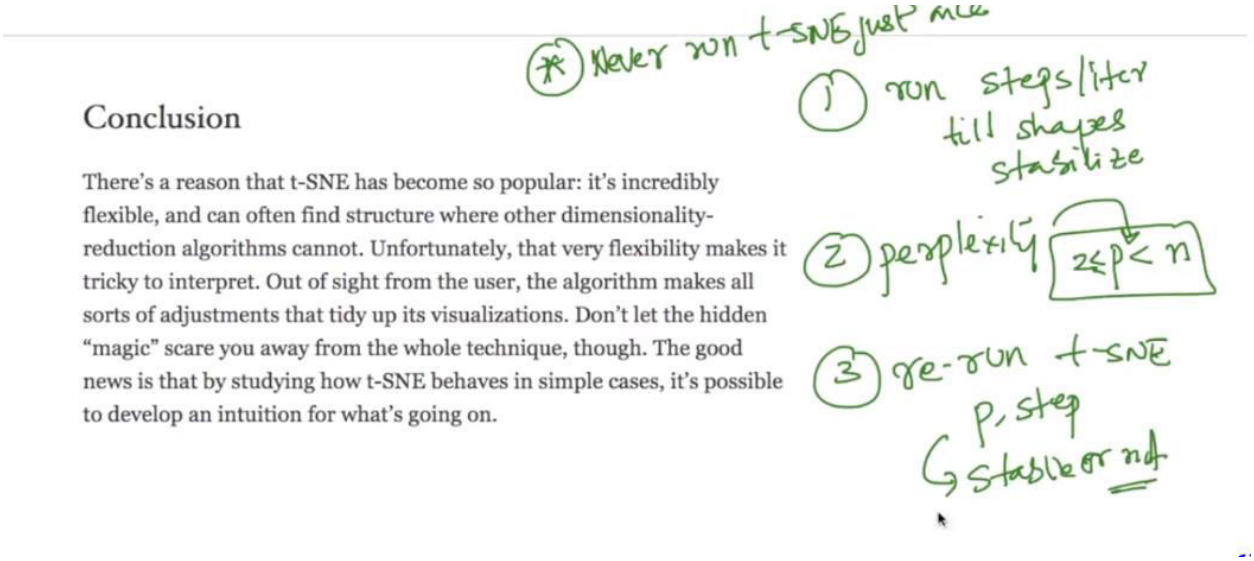
If given original data is random, then just by taking lower perplexity and seeing some clusters as relation between them is wrong, because it's random data and there is no relation between them,

Hence we should run t-SNE with different perplexity to make insight of data.



Conclusion:

- Never run t-SNE just once
- Run steps/iterations till shapes stabilize
- Take perplexity within 2 <= p < n
- Re run t-SNE with different perplexity



How to perform t-SNE:

first run with different steps till shape stabilizes then tune perplexity. 4)perplexity is number of points in a neighbourhood should it preserve

Note: this chapter is from https://distill.pub/2016/misread-tsne/ .

Usage of t-SNE in ML: https://soundcloud.com/applied-ai-course/tsne-in-modeling

Comments:

- If we dont know how data looks, and if we keep on changing perplexity how can we confirm and stop at right shape ?.
  We have to keep changing the values and at a particular set of values, you will find the shape of the visualization appears not to be changing. At the point, you have to stop the iterations.