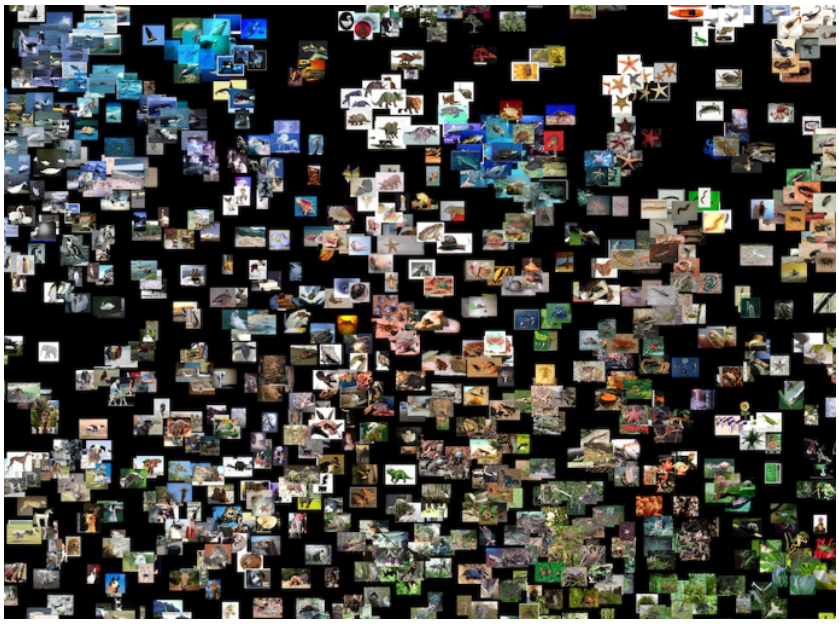
t-SNE Implementation Issues

t-SNE (t-distributed stochastic neighbor embedding) is a machine learning algorithm for visualization. It is a nonlinear dimensionality reduction technique well-suited for embedding high-dimensional data for visualization in a low-dimensional space of two or three dimensions. t-SNE has been used for visualization in a wide range of applications, including computer-security research, music analysis, cancer research, bioinformatics, and biomedical signal processing. It is often used to visualize high-level representations learned by an Artificial Neural Network. If you have some data and you can measure their pairwise differences, t-SNE visualization can help you identify various clusters. For instance, Jimmy Fallon is looking for a popular Twitter personality who is shares his interests. He identified 500 most followed accounts on Twitter, downloaded 200 of their tweets and computed distances similarities between them based on the types of things they talk about (bigram tfidf dot products). Fallon's computed similarities are then fed to t-SNE and visualized in 2 dimensions. His visualization then shows people who use similar words and phrases as him nearby to each other. Fallon made sure to use the KL-divergence formulation too since the algorithm preferentially cares about preserving the local structure of the high-dimensional data. Using t-SNE, Jimmy found that Kevin Hart is most like him and hence invites him to be his next guest.

The following are the implementation issues associated with t-SNE:

- The t-SNE method is not used in classification models since it does not learn a function from the original space to the lower dimensional space. Hence, when you try to use your classifier on new/ unseen data, you will not be able to map or pre-process the

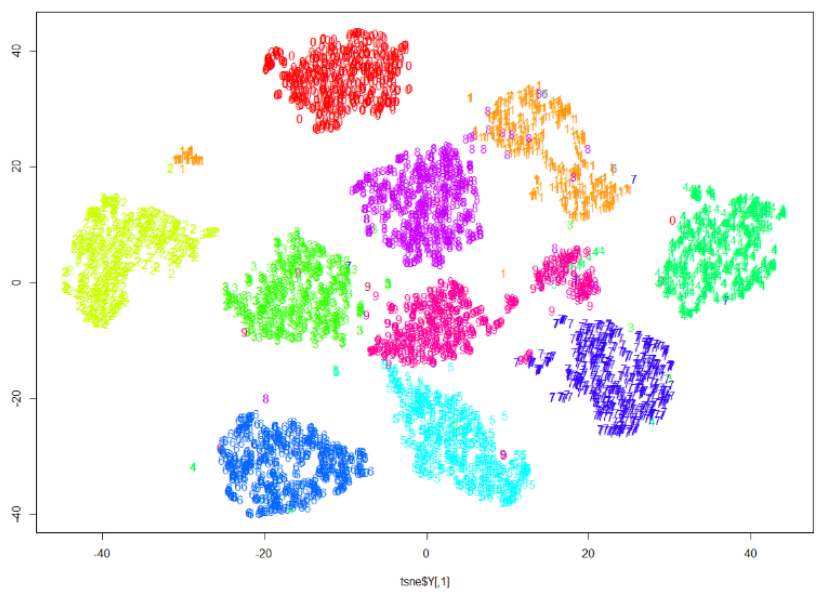
new data according to the previous t-SNE results.



The figure above depicts a t-SNE of images. There usually is a lot of content in a figure containing so many images and t-SNE helps browse though in a more organized manner by classifying the images closer to other images that share common features

- The relatively local nature of t-SNE makes it sensitive to the curse of the intrinsic dimensionality of the data. In datasets with high intrinsic dimensionality and underlying manifold that is highly varying, the local linearity assumption on the manifold that t-SNE makes may be violated. And, hence it might be less successful when applied to data with high intrinsic dimensionality (Maaten, Hinton, 2008).
- t-SNE is computationally expensive. As the algorithm finds similarity between pairs of points, it has quadratic time and space complexity as the size of the dataset. t-SNE could take several hours on million-sample datasets while Principle Component Analysis (PCA) would finish in seconds.

- The Barnes-Hut t-SNE method is limited to two or three dimensional embeddings and requires sufficient training samples to maintain preferable performance. Barnes-Hut is also an approximation of the exact method. The approximation is parametrized with the angle parameter and hence the angle parameter is unused when an exact method is required (Zheng, Zhang, Cattani, Wang, 2014).
- The algorithm is stochastic and requires multiple restarts with different seeds to yield different embeddings.
- t-SNE works only in batch mode. It has no incremental version i.e. it is not possible to run t-SNE on a dataset, then gather few more sample rows and update the t-SNE output with the new samples. You would need to re-run the model from scratch on the full dataset (previous and new samples).



The figure above depicts that the t-SNE visualization shows highly separated clusters and indicates that the components can be linearly on non-linearly separable

- t-SNE's behavior when reducing data to two or three dimensions cannot readily be extrapolated to d ¿ 3 dimensions because of the heavy tails of the student t-distribution. In high-dimensional spaces, the heavy tails comprise a relatively large portion of the probability mass under the student t-distribution, which might lead to d-dimensional data representations that do not preserve the local structure of the data well (Maaten, Hinton, 2008).
- t-SNE might be less successful when applied to datasets with intrinsic dimensionality. This is a result of the local linearity assumption on the manifold that t-SNE makes by using the Euclidean distance to preserve the similarity between datapoints.
- The cost function of t-SNE is not convex. This leads to the problem with several optimization parameters need to be chosen and the constructed solutions depending on these parameters may be different each time t-SNE runs from an initial random configuration of map points. It is not guaranteed to converge to converge to the global optimum of its cost function.

Zheng, J. Zhang, H. Cattani, C. Wang, W. Dimensionality Reduction by Supervised Neighbor Embedding Using Laplacian Search. 2014. Retrieved from: https://www.ncbi.nlm.nih.gov

Maaten, L. Hinton, G. Visualizing Data using t-SNE. 2008. Retrieved from: http://www.jmlr.org