Analyzing Effectiveness of Dimensionality Reduction Techniques

Joshua Tsai

July 2019

1 Abstract

2 Background

In this paper, we wish to determine the strength of clustering using three data dimension reduction techniques, t-SNE, principal Component Analysis (PCA), and UMAP. In order to discover the strengths and weaknesses of the three techniques, we make use of the Silhouette Coefficient, the Calinski-Harabasz Index, and the Davies-Bouldin Index. Because of the ever-increasing amounts of high dimensional data in various fields, from "" to "" , the importance of having accurate and reliable visual representations of data clusters is crucial. This brings about the necessity to rigorously explore several types of dimensional reduction techniques and their effectiveness to best serve their purpose of clustering large quantities of data. Being able to determine the factors which bring about the most effective clustering will

3 Visualization Algorithms

3.1 PCA

3.1.1 Overview

PCA is a linear dimension reduction technique that converts a set of variables that possibly could be related to a set of linearly uncorrelated variables which are the principal components. Each principal component carries a fraction of the variability of the original data set. Since it is one of the most popular algorithms used for dimensional data reduction and also one of the oldest, we are motivated to test its performance in comparison with other dimension reduction techniques [1].

3.1.2 Function

In order to determine the set of principal components, PCA first takes a data set and finds the centroid of the data. It then does a simple basis change which makes the centroid the new origin of the coordinate system, translating all other points so that they keep the same distance from the centroid. The new points can then be expressed as vectors. PCA creates a matrix, which we call Z, that is filled with these vectors. A scaled co-variance matrix, C, is created by multiplying Z^T by Z, which measures the co-variance between the variables in Z. Next, it finds the eigenvectors and eigenvalues of C by decomposing C into PDP^{-1} where D is a diagonal matrix with the eigenvalues on its main diagonal and P is a matrix of the eigenvectors whose position corresponds to the position of the eigenvalues. The eigenvalues are then sorted by magnitude, greatest first, with the

eigenvectors also being sorted correspondingly. The new sorted matrix of eigenvectors, P', then can be multiplied by Z to get the vectors of the data points whose components' weights have been determined by the direction of the eigenvectors. The eigenvalues determine the relative variance that each eigenvector carries. Thus, PCA can help reduce dimensionality by letting the user choose which eigenvectors that correspond to the greatest amounts of variance and throw away those that do not contribute as much variance.

A limitation that will be explored is the fact that PCA can only deal with linear transformations and doesn't deal with non-linear clustering as well as t-SNE or UMAP.

3.2 t-SNE

3.2.1 Overview

t-SNE is a non-linear dimensionality reduction technique used to visualize high dimensional data in just two or three dimensions. It presents a method to combat the crowding issue which comes with data in high dimensions so that the relationships between the data points is preserved even through dimension reduction [2]. Because t-SNE is so commonly used when dealing with high dimensional data, it is also a good candidate for this experiment. However, there also exists additional problems with t-SNE that arise because of its added flexibility compared with other data dimension reduction techniques. One which we wish to test is the variance of the clusters based on varying the perplexity. Density and distance between clusters may not be preserved from the original data set.

3.2.2 Function

To project data into lower dimensions, t-SNE first calculates conditional probabilities, $p_{i|j}$ based on the similarities between data points x_i and x_j. $p_{i|j}$ is calculated by the following equation:

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

Now, t-SNE defines a symmetrical similarity p_{ij} which is by definition

$$p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \text{ for } N \text{ points.}$$

Moving on to the new mapped points, y, a similar similarity q_{ij} is defined for points y_i and y_j:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k\neq i}(1 + \|y_i - y_j\|^2)^{-1}}$$

To preserve the relationships between the data points as much as possible, t-SNE minimizes the Kullback-Leibler divergence (KB) of q from p which is defined to be:

$$KB(p||q) = \sum_{i\neq j} p_{ij}\log\frac{p_{ij}}{q_{ij}}$$

3.3 UMAP

3.3.1 Overview

UMAP is another non-linear dimension reduction technique that makes use of a topological representation of a data-set and then matches it to the best fitting low dimension topological representation.

3.3.2 Function

UMAP first constructs a fuzzy topological map out of simplicies using looking at each data point's nearest neighbors. It then tries to create a lower dimension representation as accurately as possible, minimizing the cross entropy to retain the maximum correlation. To optimize the lower dimension model, it approximates the membership strength function with curves of the form $\frac{1}{1+ax^{2b}}$ and then uses stochastic gradient descent on it.

4 Evaluation Methods

4.1 Silhouette Coefficient

The Silhouette Coefficient is the first of the three metrics which we used to determine the strength of clustering. To define the Silhouette Coefficient (s) for each data point, we introduce two values, $a(i)$ and $b(i)$.

We define $a(i)$ to be

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j)$$

where C_i is the cluster in which the data point lies and $d(i, j)$ represents the distance between two points, x_i and x_j. $a(i)$ represents the mean distance of a data point to the rest of the data points in its cluster.

We define $b(i)$ to be

$$b(i) = \min \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \text{ when } k \neq i$$

$b(i)$ represents the minimum mean distance of a data point to all the data points in another cluster.

Finally,

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

The Silhouette Coefficient of the entire data set is the mean of the Silhouette Coefficients of the points. Because of its definition, the Silhouette Coefficient is restricted to be between -1 and +1. The closer it is to +1, the better concentrated and separated the clusters are. The closer it is to -1, the more the clustering can be seen as incorrect.

4.2 Calinski-Harabasz Index

The Calinski-Harabasz Index $c(k)$ is defined for k clusters to be

$$c(k) = \frac{\mathrm{Tr}(B_k)}{\mathrm{Tr}(W_k)} \times \frac{N - k}{k - 1}$$

where B_k is the between group dispersion matrix, W_k is the within cluster dispersion matrix, and N is the number of data points. The larger the Calinski-Harabasz Index is, the denser and more defined the clusters are.

4.3 Davies-Bouldin Index

To define the Davies-Bouldin Index (DB), we introduce S_i, the mean distance between every point in cluster i and the centroid of the cluster, d_{ij}, the distance between the centroids of clusters i and j, and R_{ij} which is $\frac{S_i+S_j}{dij}$ for clusters i and j. R_{ij} is known as the similarity between clusters. Then, we can define the Davies-Bouldin Index to be

$$DB = \frac{1}{k}\sum_{i=1}^{k} \max\, R_{ij,i\neq j}$$

Since the Davies-Bouldin Index measures the similarity between the clusters of the data set, a lower value closer to 0 will correspond to a better distribution of the clusters, which aim to have less similarities with each other.

5 Experiment

The main aspect of the three dimensional data reduction techniques that we want to test is their ability to create robust and accurate clusters which best reflect the main data set. In order to test this, we shall use three metrics, the Silhouette Coefficient, Calinski-Harabasz Index, and Davies-Bouldin Index. To prepare the data, we first scaled it down using a modified Min-Max feature scaling, which scales data by the following process:

$$x'_n = \frac{x_n - \min(x_n)}{\max\{\max(X_m) - \min(X_m)\}} \;\forall\; m \in [1, k]$$

where x_n is an arbitrary coordinate of the data and X_m is an arbitrary set of x_n. Since UMAP, t-SNE, and PCA all project the clusters into different sub-spaces and the metrics used to test the clusters largely depend on that sub-space, scaling the data to be within the same parameters made the results of the metrics more comparable to each other.

It should also be noted that while PCA is a dimension reduction algorithm, it has its drawbacks in its ineffective modelling of high dimensional data in just two or three dimensions. Because the total variance cannot be expected to be well explained by just two or three dimensions using the PCA method, many more dimensions have to be used in order to preserve most of the original variance. For the experiment, we used the threshold of 95% of the total variance, which was explained by 154 dimensions–as compared to the original 784. The value was chosen because it covered most of the variance–enough to give a good understanding of how the original data was like–and also because of its practicality–100 more dimensions would have given roughly only 2% more variance, which isn't that significant to have for that much more data. However, since the data is expressed in 154 dimensions, it is impossible to plot it as a comprehensible visual graph like the other two algorithms.

Another path to consider is the fact that t-SNE naturally tries to make the densities of clusters consistent, spreading out denser clusters and contracts less dense ones [3]. Because of this feature of t-SNE, it might be unfair to compare the clustering of t-SNE and other dimension reduction algorithms that revolve around measuring the densities of the clusters. Instead, alternative approaches can be sought in the form of metrics which make use of...

Additionally, there is the issue of distances between clusters. Since t-SNE also doesn't consistently preserve global distances between clusters, even when varying the perplexity[3], and because UMAP conversely wants to preserve the original structure of the data, taking scores from metrics which rely heavily upon distances between clusters may not be entirely accurate.

6 Results

6.1 Perplexity for t-SNE

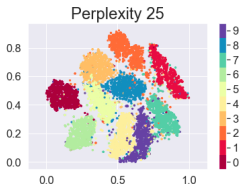


Figure 1: t-SNE plot with perplexity = 25

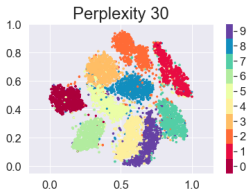


Figure 2: t-SNE plot with perplexity = 30

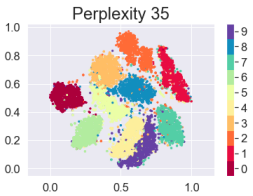


Figure 3: t-SNE plot with perplexity = 35

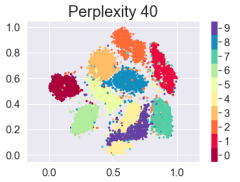


Figure 4: t-SNE plot with perplexity = 40

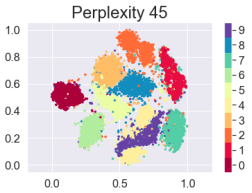


Figure 5: t-SNE plot with perplexity = 45

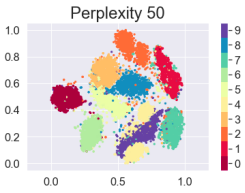
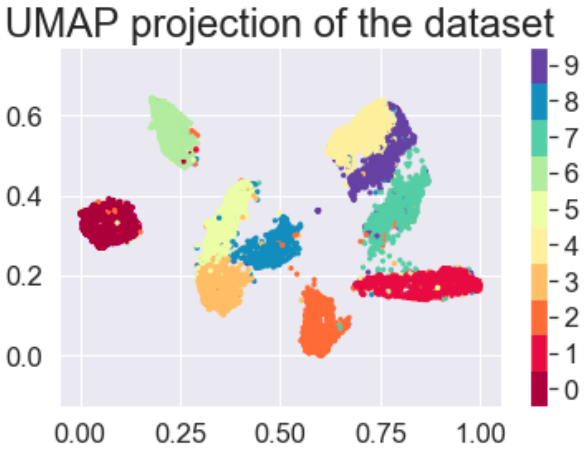


Figure 6: t-SNE plot with perplexity = 50

Perplexity for t-SNE			
Perplexity	Silhouette Coefficient	Calinski-Harabasz Index	Davies-Bouldin Index
25	0.366	6506.983	1.124
30	0.370	6501.650	1.141
35	0.362	6549.344	1.085
40	0.354	6211.495	1.743
45	0.354	6095.871	2.695
50	0.353	6007.021	4.333

6.2 UMAP



Scores			
Algorithm	Silhouette Coefficient	Calinski-Harabasz Index	Davies-Bouldin Index
PCA	0.0544	331.354	3.689
t-SNE*	0.370	6549.344	1.085
UMAP	0.454	11048.565	0.966

*t-SNE has been optimized

7 Conclusion

The relative performance of each of the visualization methods was the same for each of the metrics; PCA, t-SNE, and UMAP. This leads us to believe that UMAP has the most reliable clustering techniques.

8 References

[1] Jolliffe, Ian T., and Jorge Cadima. "Principal Component Analysis: a Review and Recent Developments." Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 374, no. 2065, 2016, p. 20150202., doi:10.1098/rsta.2015.0202.
[2] Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing Data Using t-SNE." Journal of Machine Learning Research, 2008., Retrieved from http://www.jmlr.org
[3] Wattenberg, Martin, et al. "How to Use t-SNE Effectively." Distill, Google Brain, 13 Oct. 2016, distill.pub/2016/misread-tsne/.
[4] McInnes, Leland. "How UMAP Works¶." How UMAP Works - Umap 0.3 Documentation, 2018, umap-learn.readthedocs.io/en/latest/how_umap_works.html.