# Dimension Reduction Techniques for MNIST Clustering

Wai Ming Chau

# Motivation

- The clustering problem is always associated with the curse of high dimensionality.
- In addition to the significant computational burden, data becomes sparse in high dimensional space and the concepts of proximity, distance, or nearest neighbour may not even be qualitatively meaningful in high dimensional space.
- Therefore, dimensionality reduction techniques are important in converting high-dimensional data into lower-dimensional data by extracting meaningful features.

# Goal

1. We will convert high-dimensional data into two-dimensional data with different dimensionality reduction techniques.
2. Apply the K-means clustering algorithm to obtain clusters from the reduced-dimension data.
3. Conduct quantitative and qualitative analysis to determine the effectiveness of various dimension reduction techniques and their impact on the performance of the K-means clustering algorithm

# Dataset

- In this project, we are working with the MNIST dataset, consisting of 60,000 examples of handwritten digits ranging from 0 to 9.
- Each digit is represented by a $28 \times 28$ matrix with a unique handwriting style.
- Our intuition suggests that there should be 10 clusters present in the dataset

## Experimental setting

- Randomly select 5000 examples from the dataset of 60000 and apply the algorithm to each subset.
- Repeat this process 10 times.

# Dimension-reduction techniques

We will apply the following dimension-reduction techniques to convert high-dimensional data into two-dimensional data:

## Linear Methods

1. **Principal component analysis (PCA)** identifies the underlying linear structure of the data with the highest variance.

2. **Linear Discriminant Analysis(LDA)** is a supervised technique that projects data onto a lower-dimensional space to maximize class separation based on linear discriminant functions.

## Manifold learning

3. **Isometric Feature Mapping (ISOMAP)** preserves the geodesic distances in high-dimensional space to capture intrinsic geometric structures in a low-dimensional representation.

4. **Locally Linear Embedding (LLE)** preserves local linear relationships between nearby points in high-dimensional space for a low-dimensional representation.

5. **Laplacian Eigenmaps (LAP)** preserves global relationships between all points in high-dimensional space for a low-dimensional representation using the graph Laplacian.

6. **t-Distributed Stochastic Neighbor Embedding (t-SNE)** models similarity between nearby points in high and low-dimensional spaces to capture local structure in a low-dimensional representation.

# Clustering Analysis

## K-mean clustering for MNIST dataset

- K-means algorithm measures how similar or different things are using Euclidean distance. It groups things together based on how close they are to each other:

1. Initialize 10 cluster centroids randomly
2. Assign each data point to the nearest centroid based on the Euclidean distance.
3. Recalculate the centroid for each cluster as the mean of all data points assigned to it.
4. Repeat steps 3-4 until the cluster assignments no longer change or a maximum number of iterations is reached.

We will calculate the average of three metrics to evaluate the performance of clustering using various dimensionality reduction techniques in 10 numerical experiments.

1. **Purity** measures the homogeneity of the clusters with respect to the ground truth labels

$$\frac{1}{N}\sum_{k}\max_{j}\left|\text{cluster}_k \cap \text{class}_j\right|$$

2. **Normalized mutual information** measures the similarity between the ground truth labels and the labels assigned by a clustering algorithm while considering their mutual information and normalizing it by the entropy of each label set.

$$\text{NMI}(\text{cluster}, \text{class}) = \frac{2I(\text{cluster}; \text{class})}{H(\text{cluster}) + H(\text{class})}$$

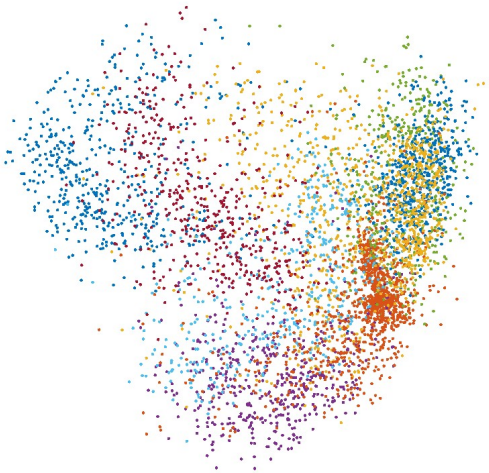
# Numerical results

| Method | PCA | LDA | ISOMAP | LLE | LAP | t-SNE | NONE |
|---|---|---|---|---|---|---|---|
| Purity | 0.4101 | 0.5247 | 0.4563 | 0.5873 | 0.5397 | 0.7458 | 0.5758 |
| Normalized mutual information | 0.3544 | 0.4652 | 0.3575 | 0.6030 | 0.5421 | 0.7038 | 0.4912 |

- The results suggest that t-SNE is the most effective clustering method, performing better than other reduction techniques and even full-dimensional clustering.
- LLE, LDA, and LAP show similar performance to full-dimensional clustering, while PCA and ISOMAP perform worse.

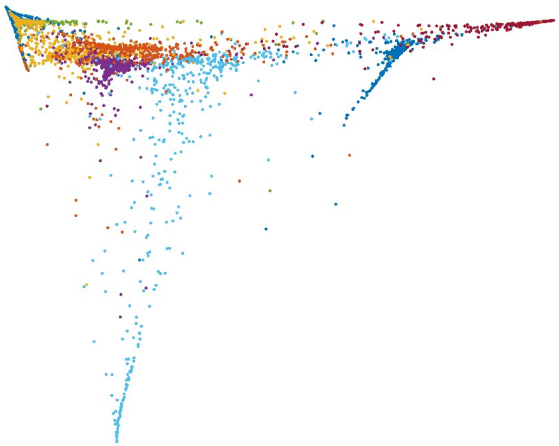
(a)

(b)



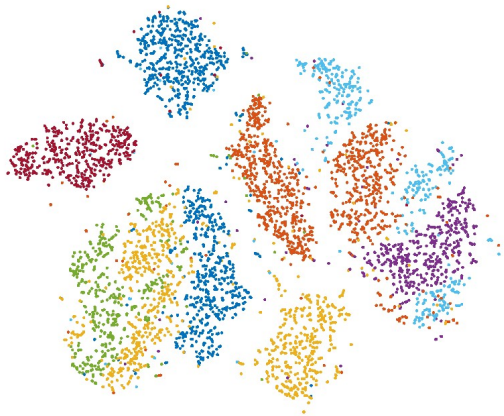
(c)

Figure: (a) PCA (b)LDA (c)ISOMAP



(d)

(e)



(f)

Figure: (d)LLE (e)LAP (f)t-SNE

# Analysis

Clustering algorithms rely on both **Between-Class Variability** (BCV), which measures the degree to which groups are spread apart from each other

$$\text{BCV} = \sum_{i=1}^{k} n_i (\boldsymbol{m}_i - \boldsymbol{m})^T (\boldsymbol{m}_i - \boldsymbol{m})$$

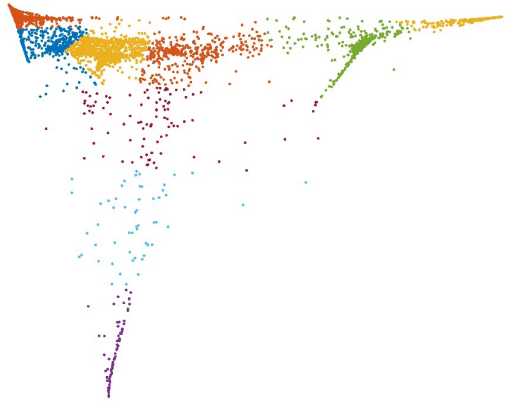
and **Within-Class Variability** (WCV), which measures how tightly they are grouped

$$\text{WCV} = \sum_{i=1}^{k} \sum_{\boldsymbol{x} \in \text{Class}_i} (\boldsymbol{x} - \boldsymbol{m}_i)^T (\boldsymbol{x} - \boldsymbol{m}_i)$$

| Method | PCA | LDA | ISOMAP | LLE | LAP | t-SNE |
|---|---|---|---|---|---|---|
| BCV | 134.2739 | 189.0304 | 215.1110 | 339.5823 | 328.4660 | 422.3671 |
| WCV | 86.3030 | 56.3350 | 110.0640 | 43.8792 | 53.3823 | 92.3314 |

The geometric structure of the reduced dataset can also impact clustering performance.

(a)



(b)

(c)

Figure: The clustering result of the embedded coordinates (a) LDA (b) LLE (c) LAP

# Conclusion

- Dimensionality reduction techniques are generally effective in capturing most of the information from a 28x28 matrix and representing it in a 2-dimensional space.
- Among the six methods compared, t-SNE performed the best in clustering as it maximized the within-cluster variance (WCV) and preserved the near-spherical structure, which helped the k-means algorithm to identify the clusters accurately.