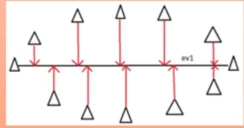
Focus for this lecture

Acquire Raw Data → Prepare / Clean Data, Visualization → Feature Engineering → Pick Model and Hyper-params for Task → Model Training / Optimization → Evaluate Model Performance → Deploy Model

- Data Visualization
- Non-linear Dimensionality Reduction
  - ISOMAP
  - LLE
  - t-SNE

Acquire Raw Data → Prepare / Clean Data, Visualization → Feature Engineering → Pick Model and Hyper-params for Task → Model Training / Optimization → Evaluate Model Performance → Deploy Model

Data Visualization



PCA

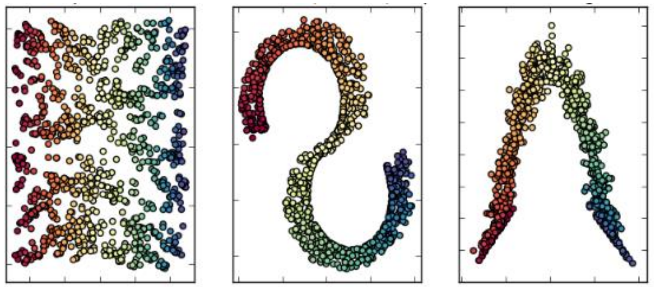
Data Visualization

When Data is High-Dimensional

Motivation

Two sides to data visualization:

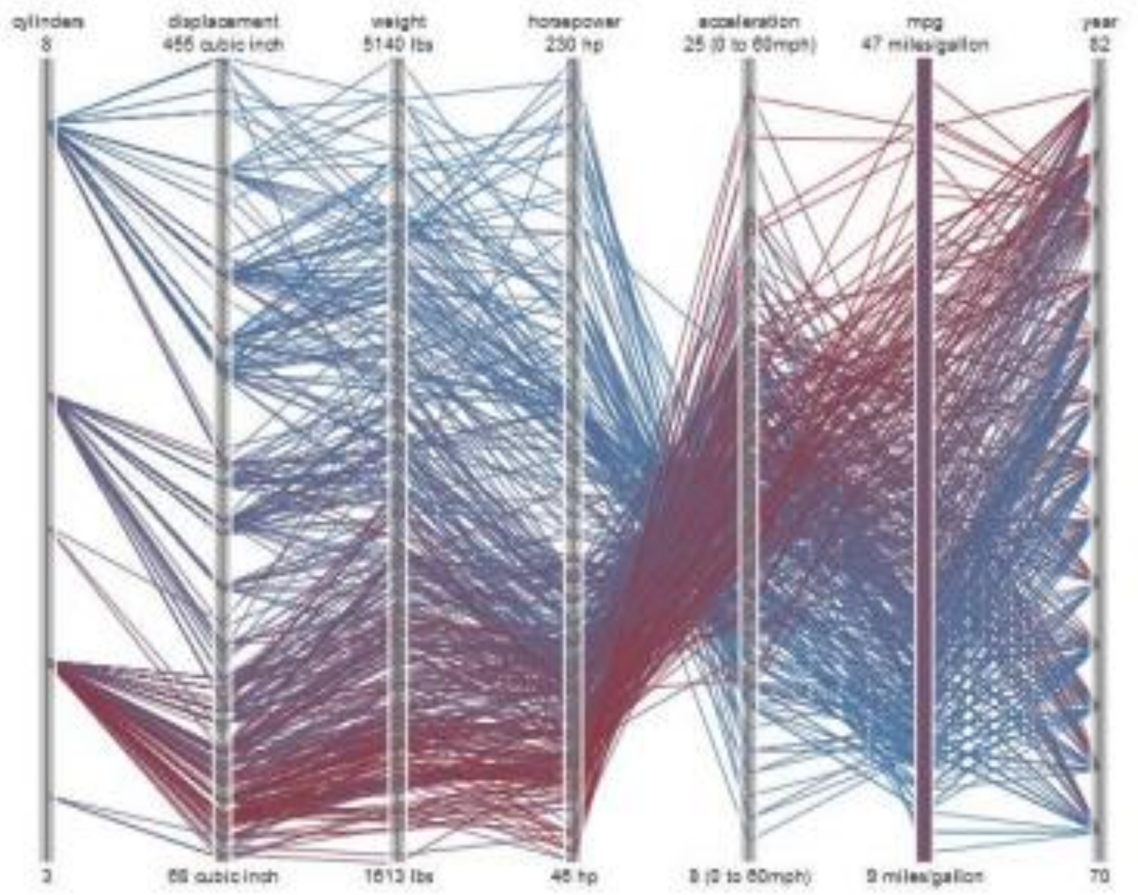
- **Data Exploration:** Making sure **you** understand your data
- **Data Communication:** Making sure **others** understand your insights and/or can use your data easily



Why Data Visualization

How do we look at HD data?

- Dimensionality Reduction
- Other methods
  - Have limitations



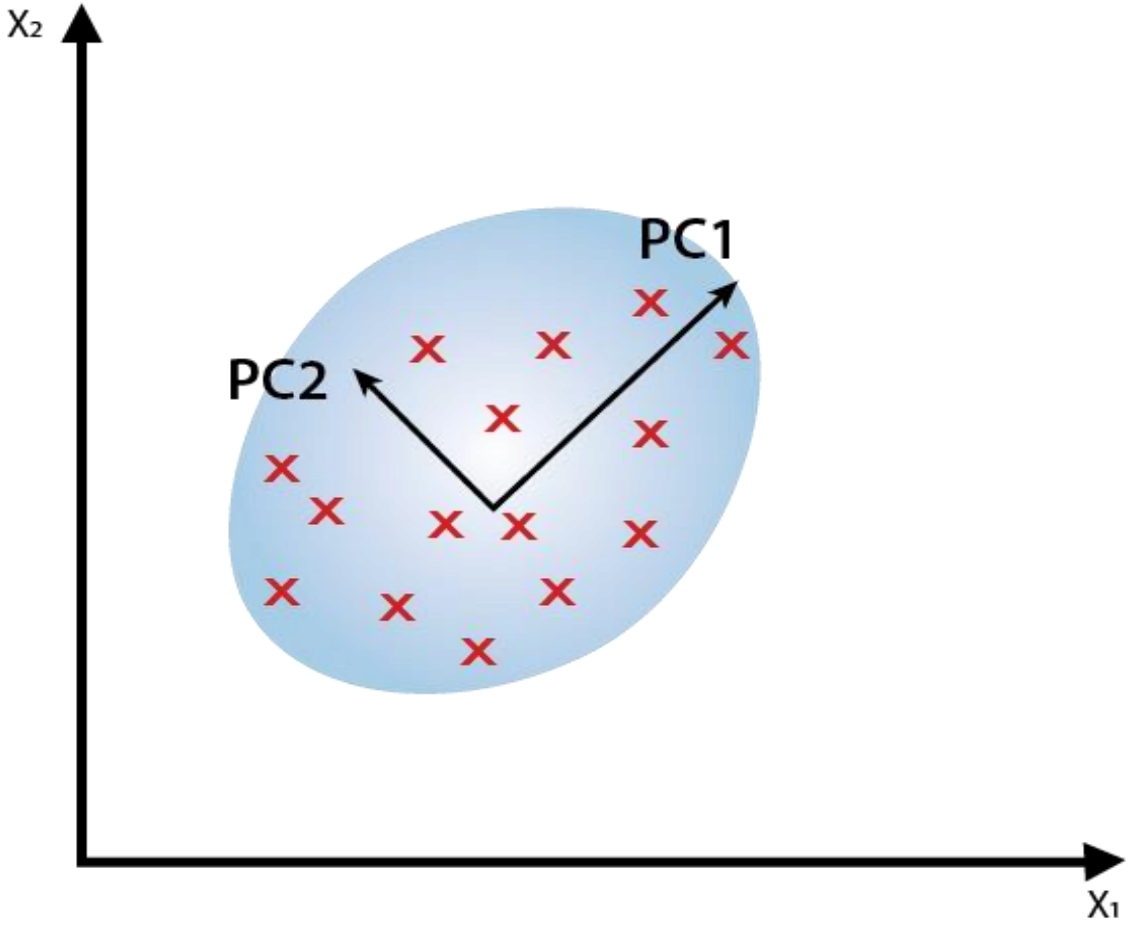
PLOT INDEPENDENT DIMENSIONS

Dimensionality Reduction: Assumptions

- High-dimensional data often lives in a lower dimensional manifold (sub-space)
- We can represent the data points well by using just their lower dimensional co-ordinates
- The lower dimensional data will capture the distribution of (pair-wise distances between) points in high dimensions
- If the manifold is a linear sub-space, we can use PCA

Principal Component Analysis

PCA is used to reduce the dimensionality of data



Dimensionality Reduction: Approaches

- Global Approach: All distances in HD are equally important and should be captured in the LD representation
  - Like PCA?

- Local Approach: Only smaller distances in HD are meaningful /reliable/ interesting to us
  - Could weigh smaller and larger distances differently?

Learning Problem

high-dimensional data set

$$\mathcal{X} = \{x_1, x_2, \ldots, x_n\} \longrightarrow \mathcal{Y} = \{y_1, y_2, \ldots, y_n\}$$

two or three-dimensional data

Formal Framework

Minimize an objective function that measures the discrepancy between similarities in the data and similarities in the map

MDS (Multidimensional scaling)

- Minimize an objective function that measures the discrepancy between similarities in the data and similarities in the map.

- Distance between samples in “high” dimension and “low’’ dimension is same (or D-d) is minimized.

MDS: Multidimensional Scaling

- Find a representation that best preserves pair-wise distances
- Mathematically:
- Start with random vectors in LD
- Update the vectors so as to minimize the cost function
  - Gradient descent

$$Cost = \sum_{i<j} (d_{ij} - \hat{d}_{ij})^2$$

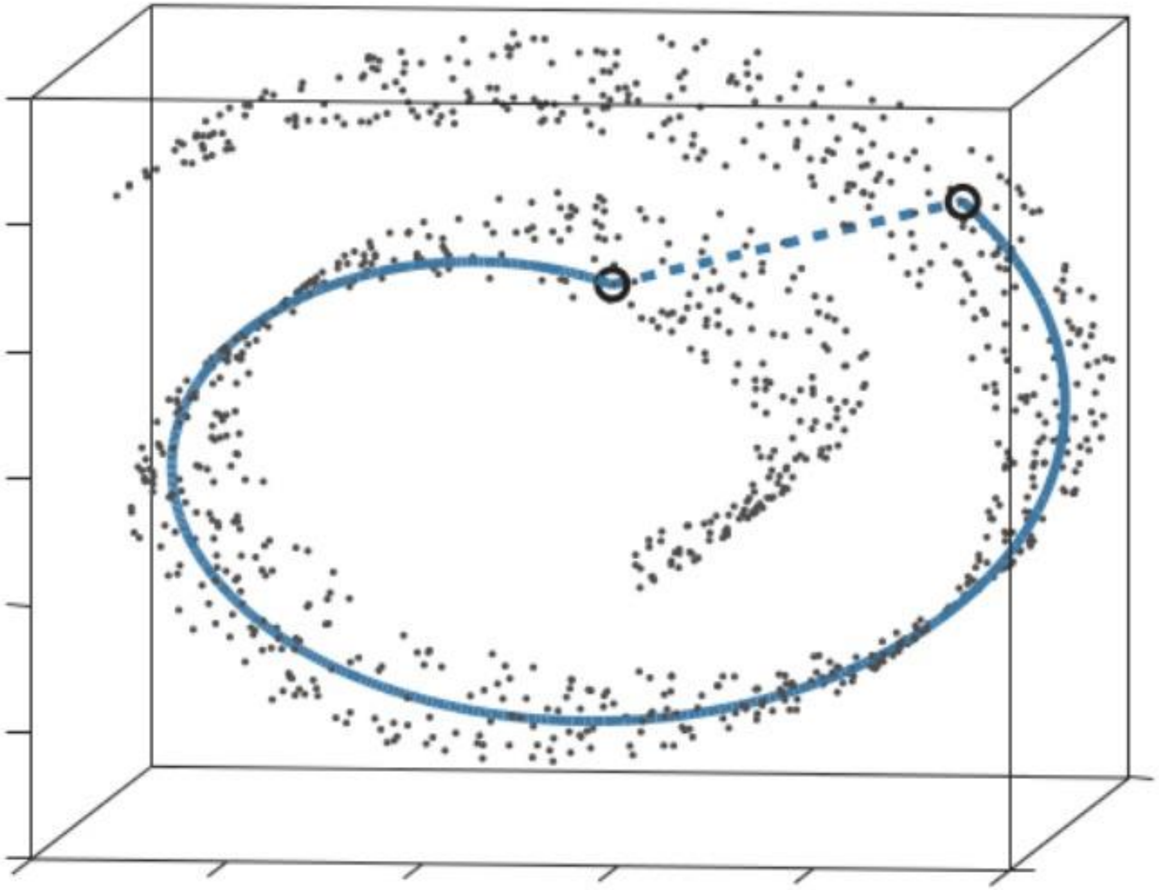
$$d_{ij} = \| x_i - x_j \|^2$$

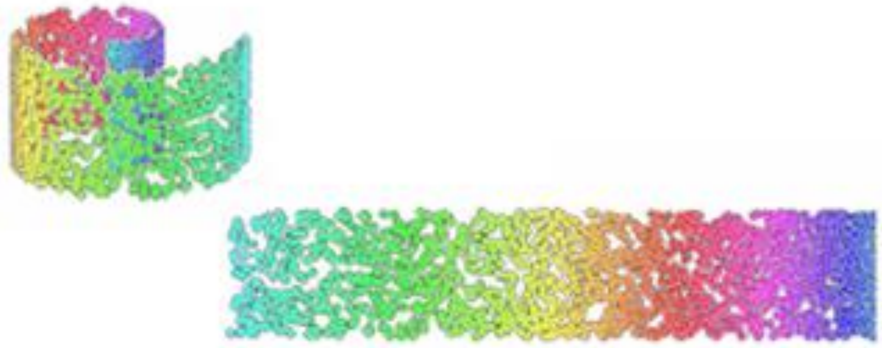
$$\hat{d}_{ij} = \| y_i - y_j \|^2$$

Can get stuck in local minima

Still Linear
Can be PCA

In complex datasets, large distances are usually *less* indicative

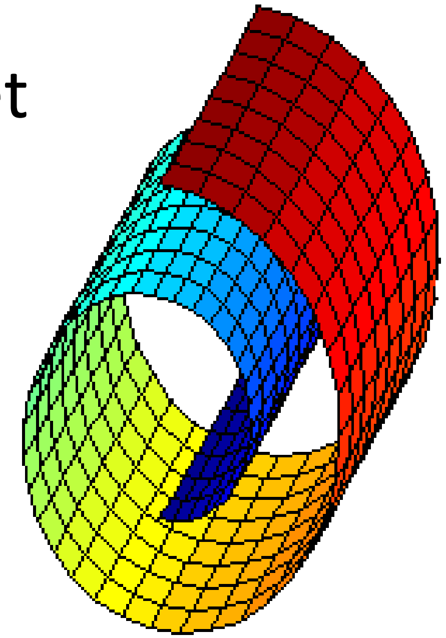




PCA: Fundamental Shortcoming

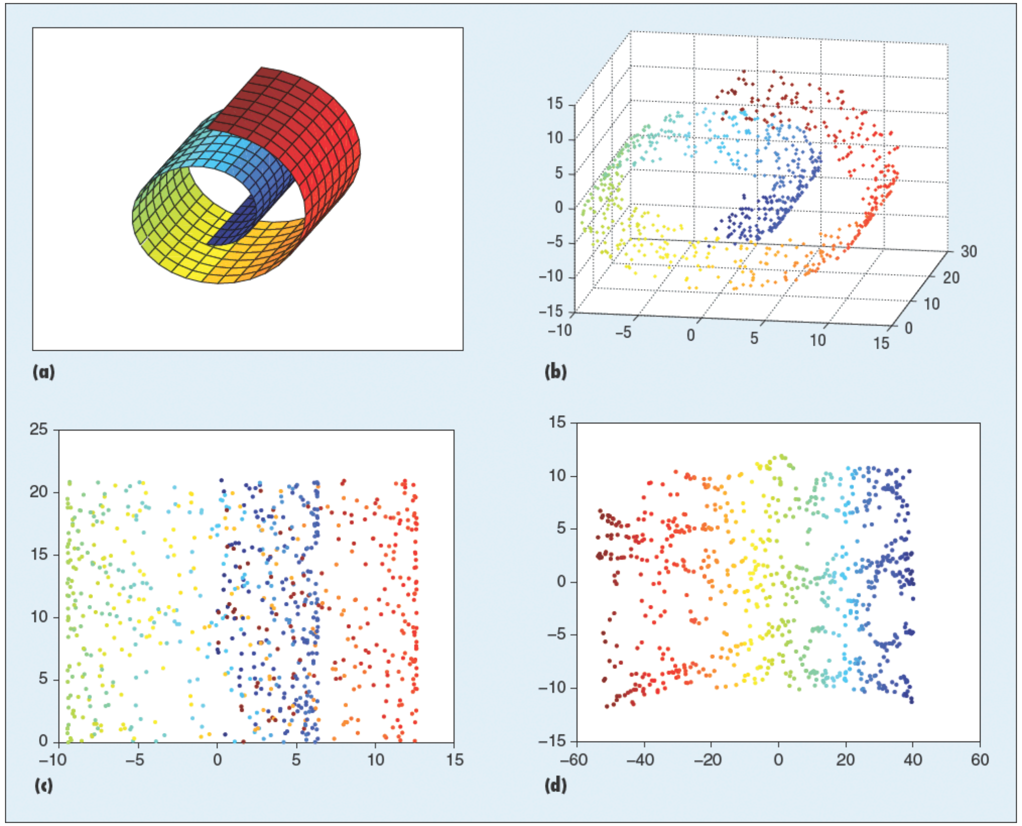
- World is often non-linear

- Consider the Swiss-Roll dataset
  - What would PCA give?
  - What do we want?

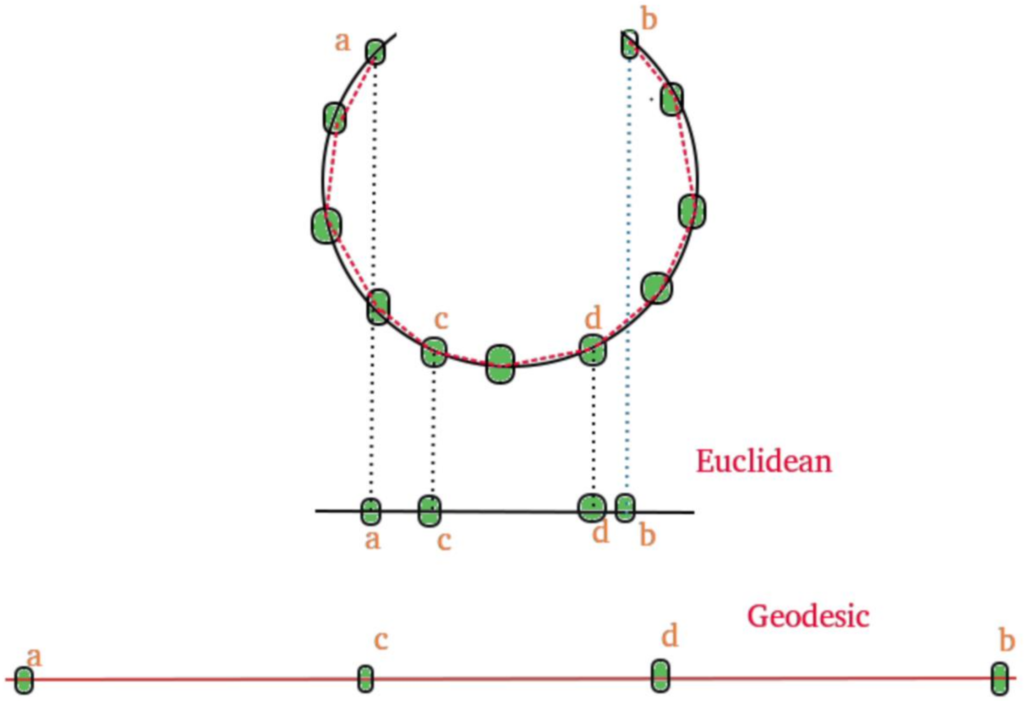


What do we really want?

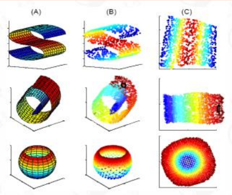
- Find a lower-dimensional representation such that:
  - Distances in LD ≅ Distances in HD
  - Closer distances are more important
- Unrolling the Swiss roll
- Do not insist on being able to get HD back from LD
  - Using for visualization



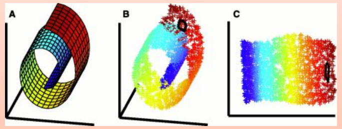
In some cases, geodesic distances are better than Euclidean distances



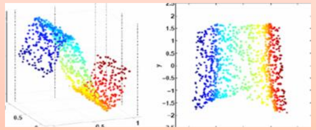
Acquire Raw Data → Prepare / Clean Data, Visualization → Feature Engineering → Pick Model and Hyper-params for Task → Model Training / Optimization → Evaluate Model Performance → Deploy Model



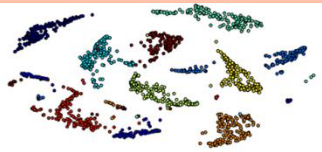
Non-linear Dimensionality Reduction



ISOMAP



LLE



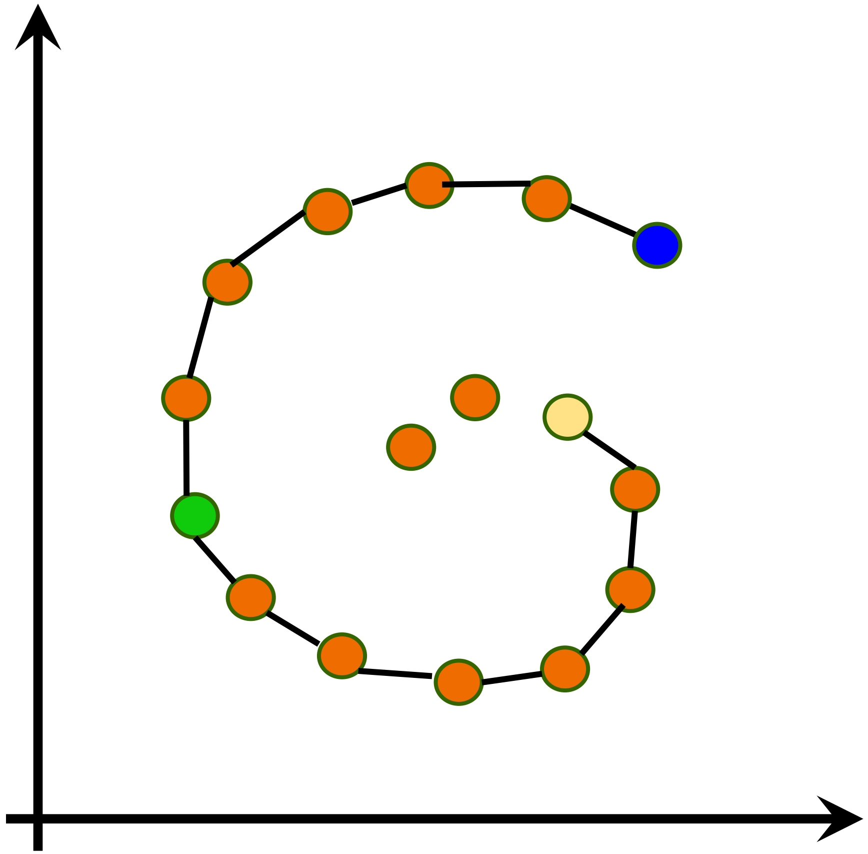
t- SNE

Non-Linear Dimensionality Reduction

ISOMAP and LLE

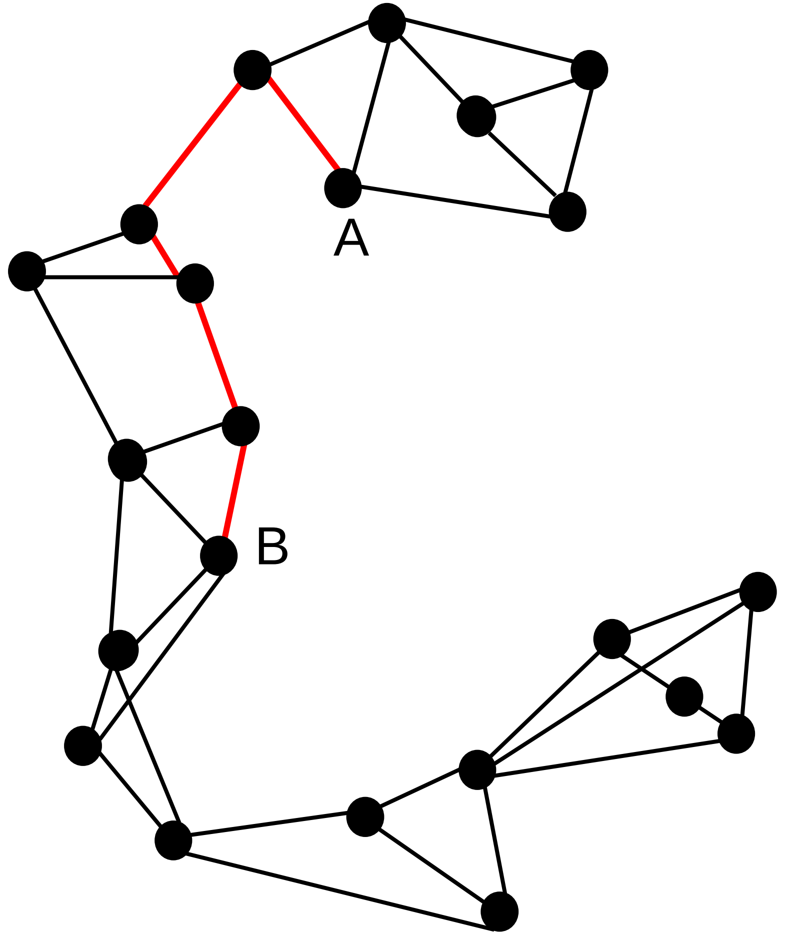
ISOMAP (Isometric Mapping)

- d(●,●) > d(●,●)
- Is Euclidean metric the right distance metric?
- How to robustly measure distances along the manifold?

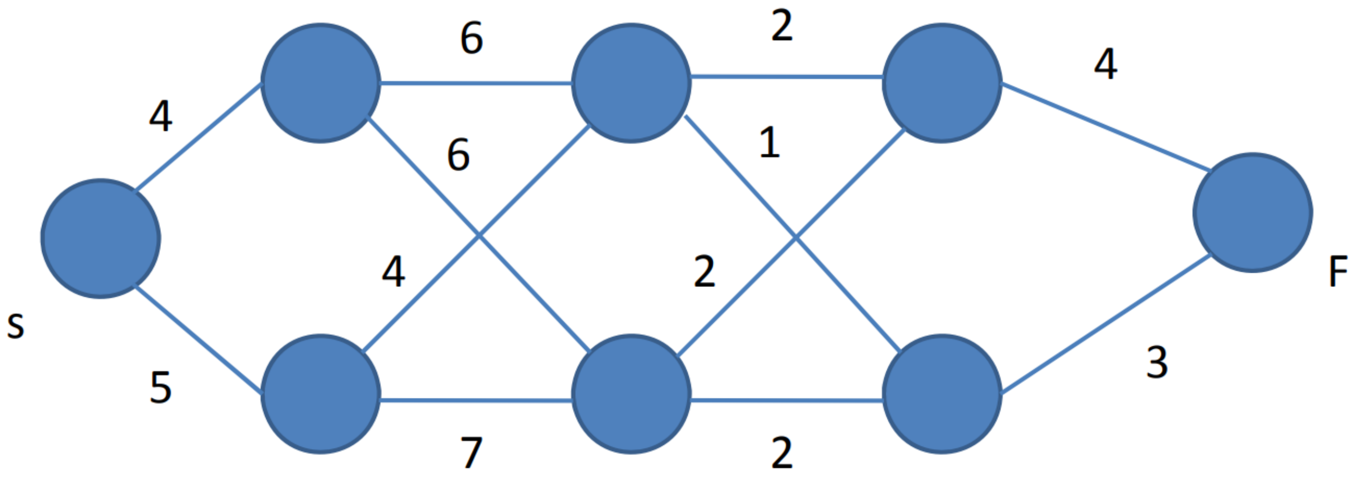


ISOMAP

- How does ISOMAP measure the MD?
- Connect each data point to its K nearest neighbors in the high-dimensional space.
- Link weights: True Euclidean distances.
- $MD(A, B) = ShortestPath(A, B)$ in this ***neighborhood graph***.
- Compute the low-dimensional embedding as in Metric MDS.



Dijkstra’s algorithm for shortest path

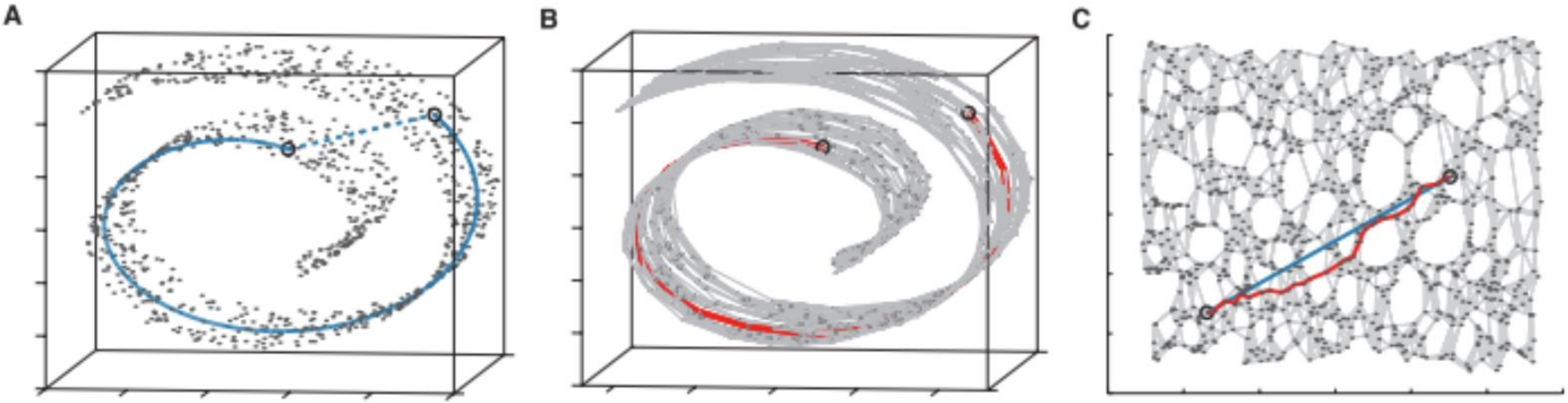


Dynamic Programming

Issues with ISOMAP

- The connectivity of each data point in the neighborhood graph is defined as its nearest *k* Euclidean neighbors in the high-dimensional space.
- This step is vulnerable to "short-circuit errors" if *k* is too large with respect to the manifold structure or if noise in the data moves the points slightly off the manifold.
- Even a single short-circuit error can alter many entries in the geodesic distance matrix, which in turn can lead to a drastically different (and incorrect) low-dimensional embedding.
- Conversely, if *k* is too small, the neighborhood graph may become too sparse to approximate geodesic paths accurately.

ISOMAP on Swiss Roll data



LLE: Locally Linear Embedding

- Idea: Preserve the structure of local neighbourhood

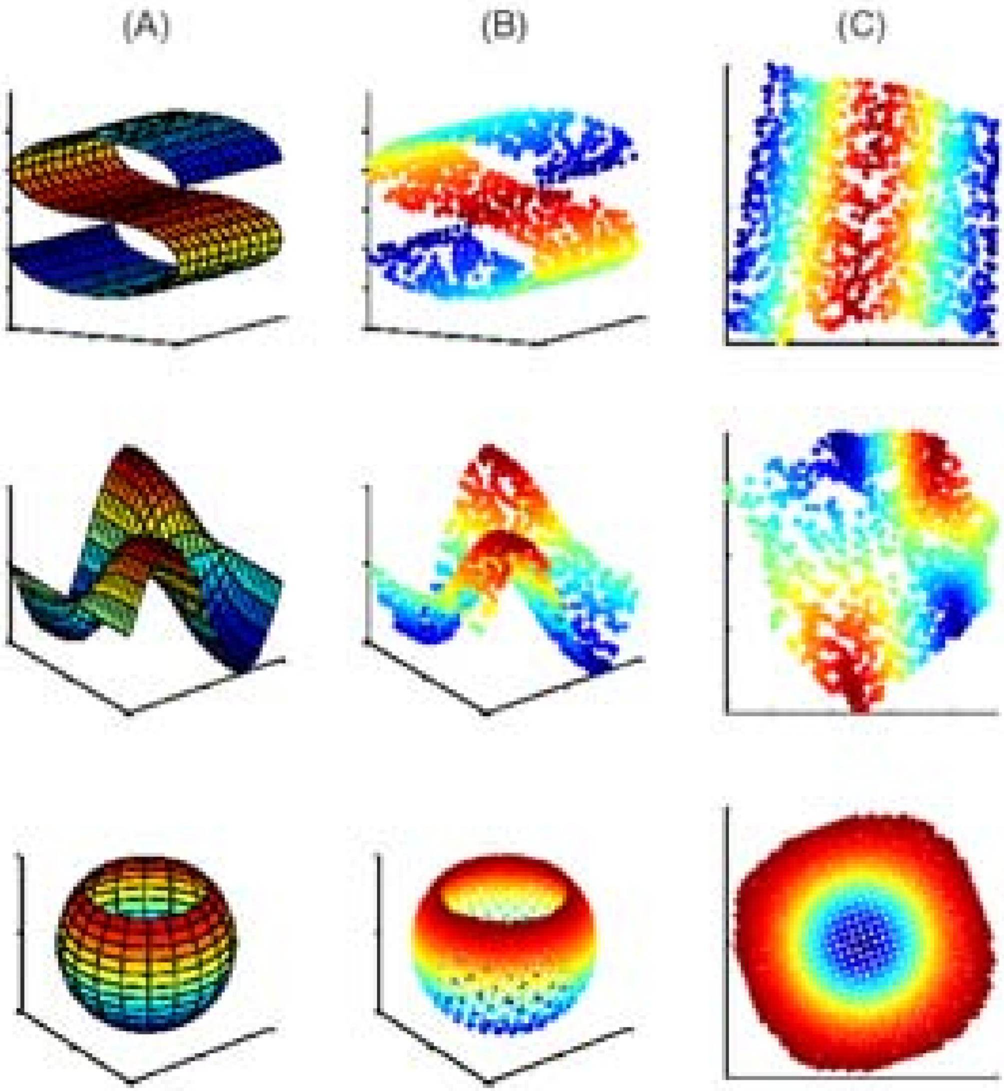
$$\mathbf{x}_i \approx \sum_j w_{ij} \mathbf{x}_j$$

- Approach:
  - Represent each point as a weighted combination of its Neighbours in HD. Remember the $w_{ij}s$.
  - Find a LD representation that minimize the representation error:

$$Cost = \sum_i \| \mathbf{y}_i - \sum_{j \varepsilon N(i)} w_{ij} \mathbf{y}_j \|^2$$

  - The weights w_{ij} refer to the amount of contribution the point x_i has while reconstructing the point x_i. The cost function is minimized under two constraints: (a) Each data point x_i is reconstructed only from its neighbors, thus enforcing w_{ij} to be zero if point x_j is not a neighbor of the point x_i and (b) The sum of every row of the weight matrix equals 1.
  - Also ys should have unit variance across each dimension.

LLE Examples



t-SNE
(T-distributed_stochastic_neighbor_embedding)

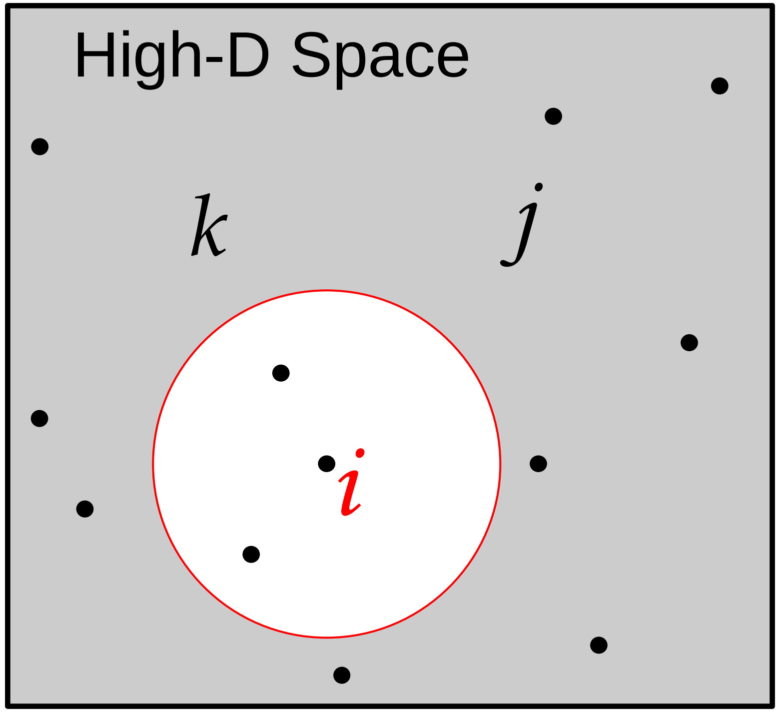
Improving Visualization

SNE and t-SNE

- Idea is simple: Instead of distance think about probabilities. P_{ij} as the probability of j in the neighborhood of i.
- For each point, we have now a probability vector (of size N).
  - SNE uses Gaussian. T-SNE uses another t-distribution (with 1 degree of freedom).
- We want these prob vectors to be the same in low dimensional.
- Optimize using gradient descent.

SNE: A Probabilistic Embedding

- For point j, there is a probability of it being called a neighbour of i.
- The probability is a function of the distance between i and j in HD.
- We end up with a matrix of probabilities.
- Each point is then represented as a probability distribution over all other points: A row of the above matrix

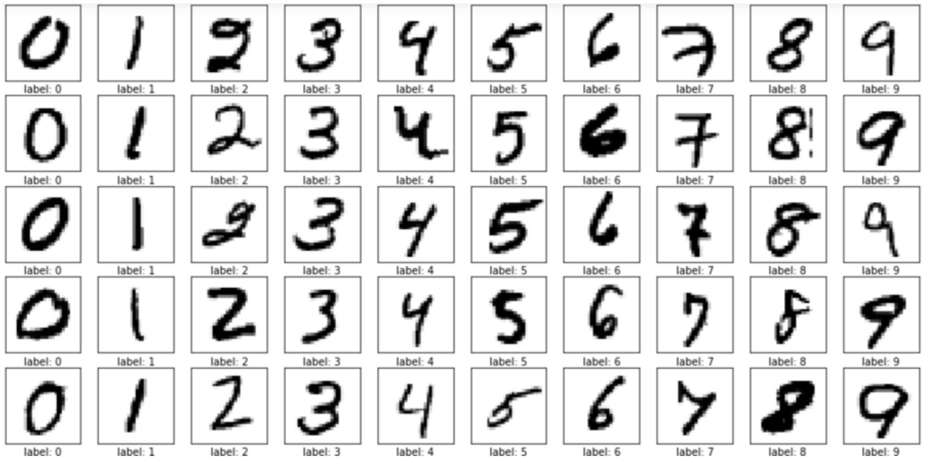


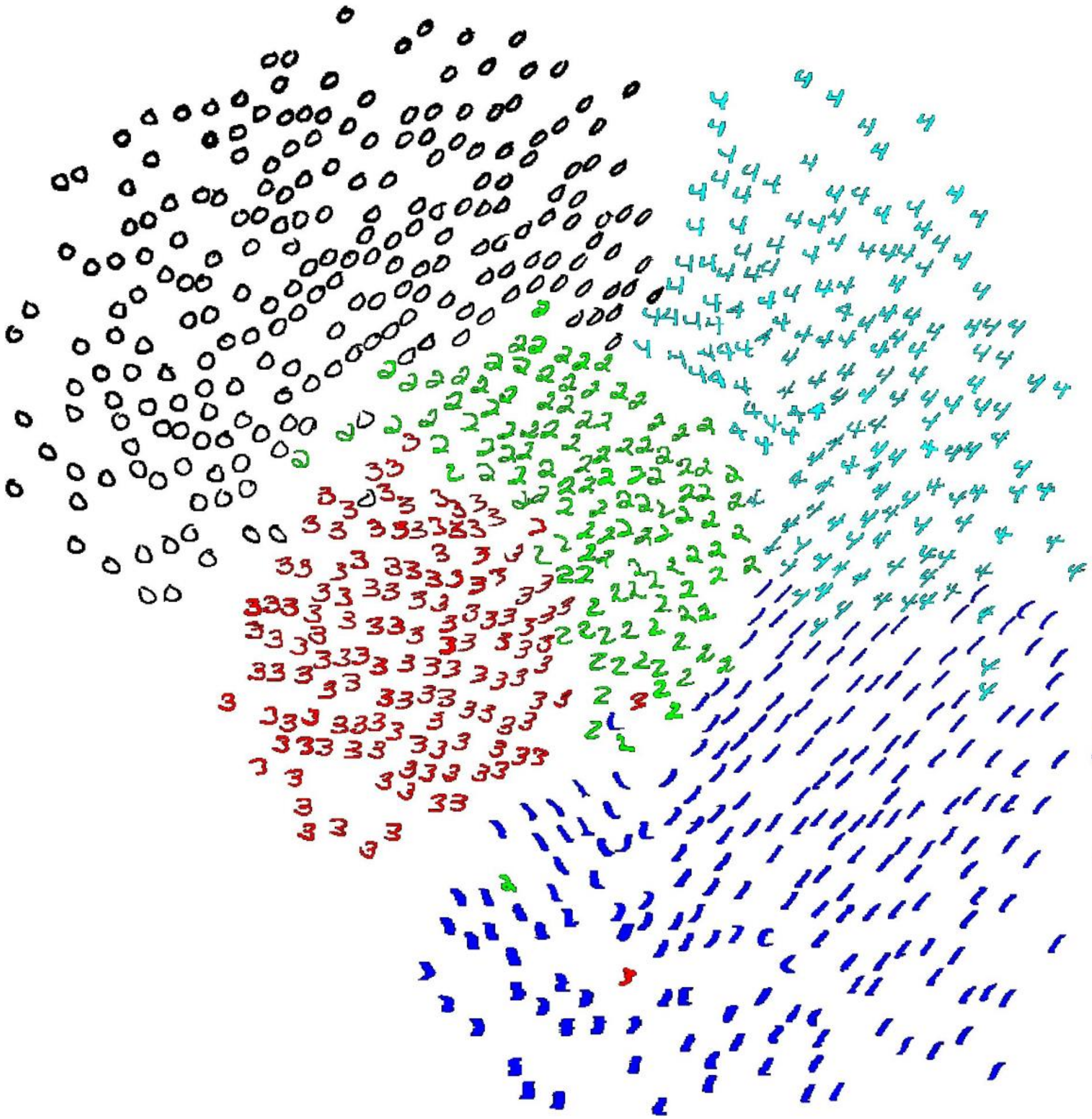
$$p_{j|i} = \frac{e^{-d_{ij}^2 / 2\sigma_i^2}}{\sum_k e^{-d_{ik}^2 / 2\sigma_i^2}}$$

SNE on MNIST

MNIST Handwritten digits dataset

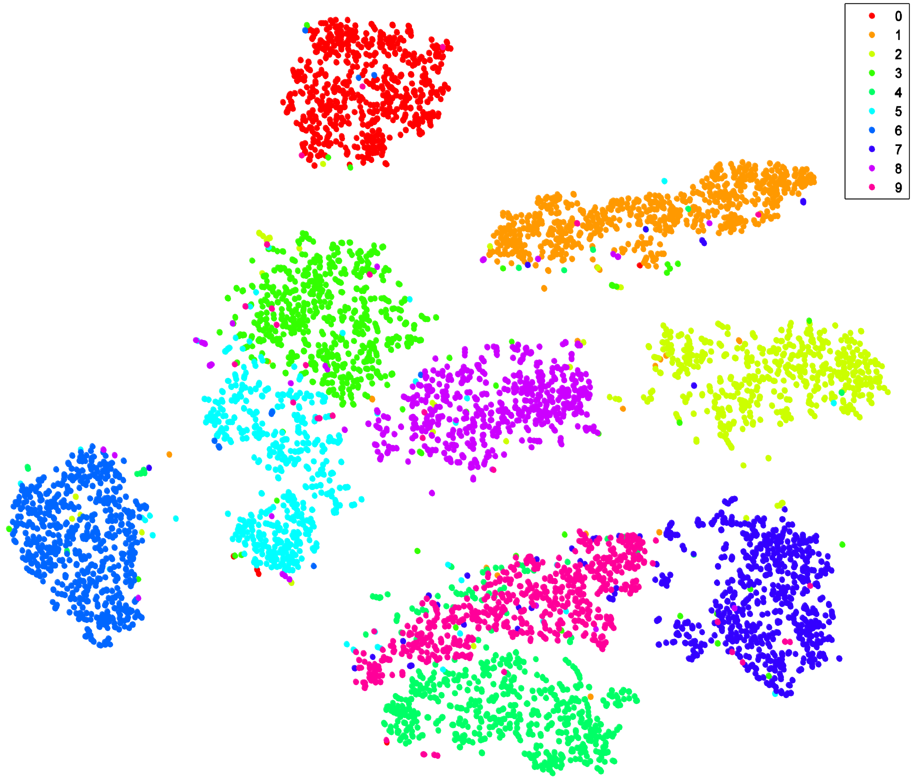
- 28x28 binary images
- Large variations in writing



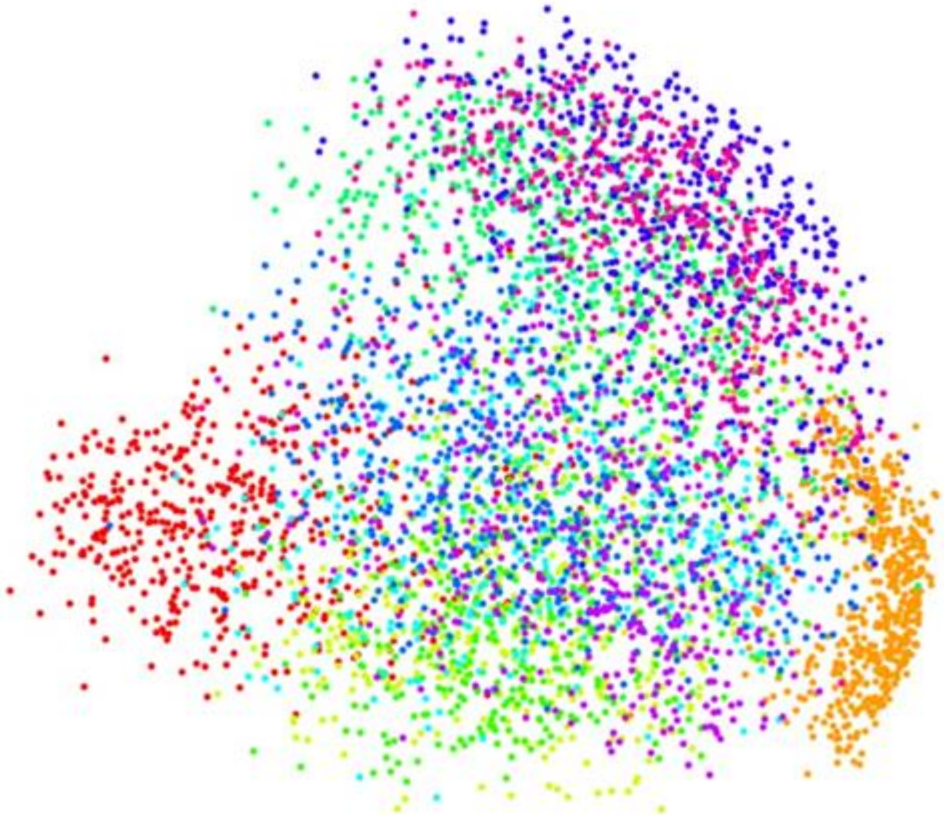


t-SNE on MNIST; Summary

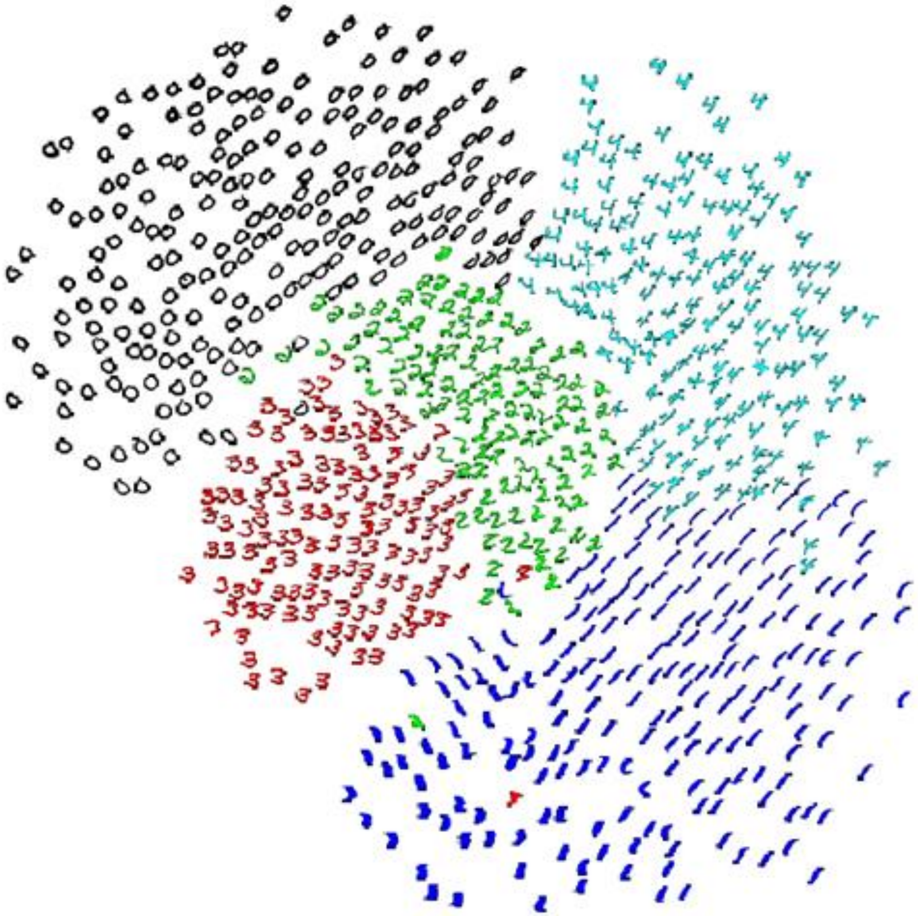
- Classes are much better separated
- Note that the method is unsupervised!!!
- Efficient approximations exist
- Most popular LD visualization at the moment.



PCA on MNIST *(0-9)*



SNE on MNIST *(0-5)*



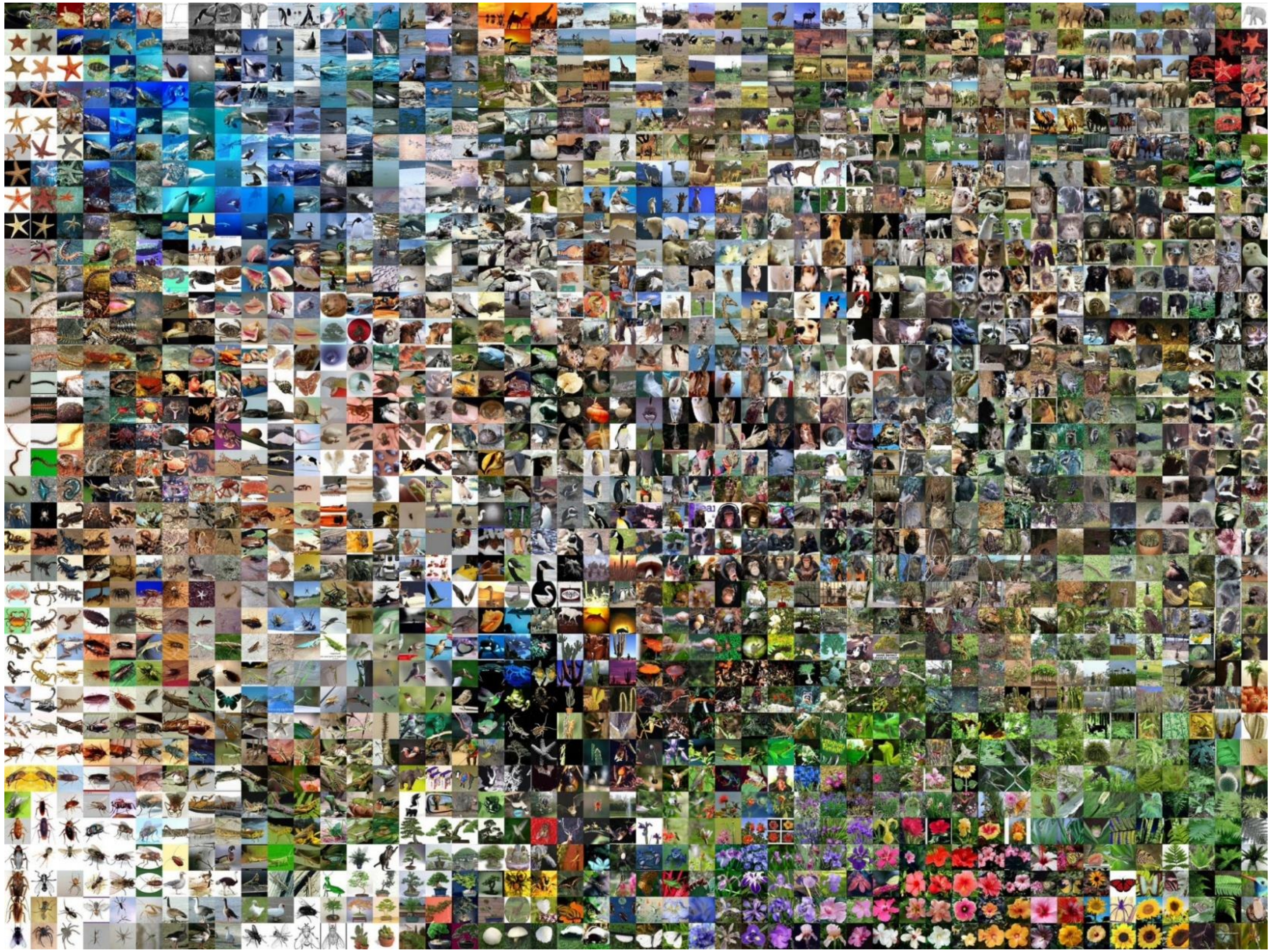
Are we overfitting?

"classic" Machine Learning	vs	Visualization
Goal: Generalization		Goal: Visualization
Given a Training set, Do well on a Test set.		We just want to "do well" on our data ("training set")
Overfitting is undesirable		**"Overfitting" is desirable**

t-SNE

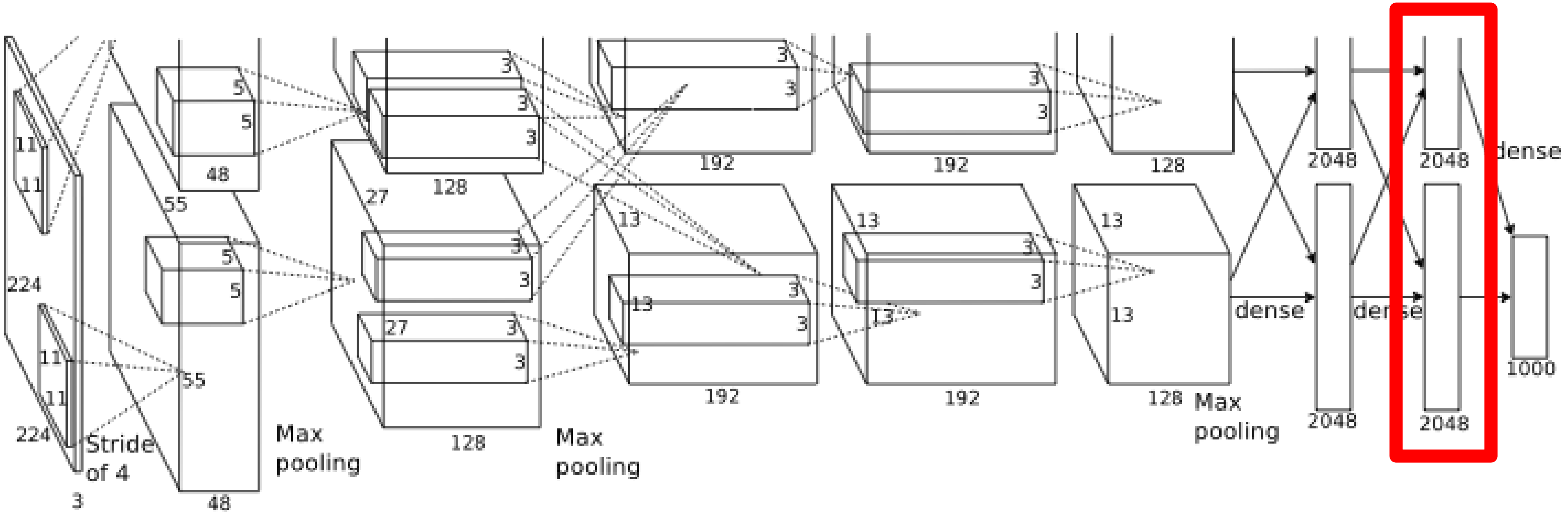
https://github.com/oreillymedia/t-SNE-tutorial

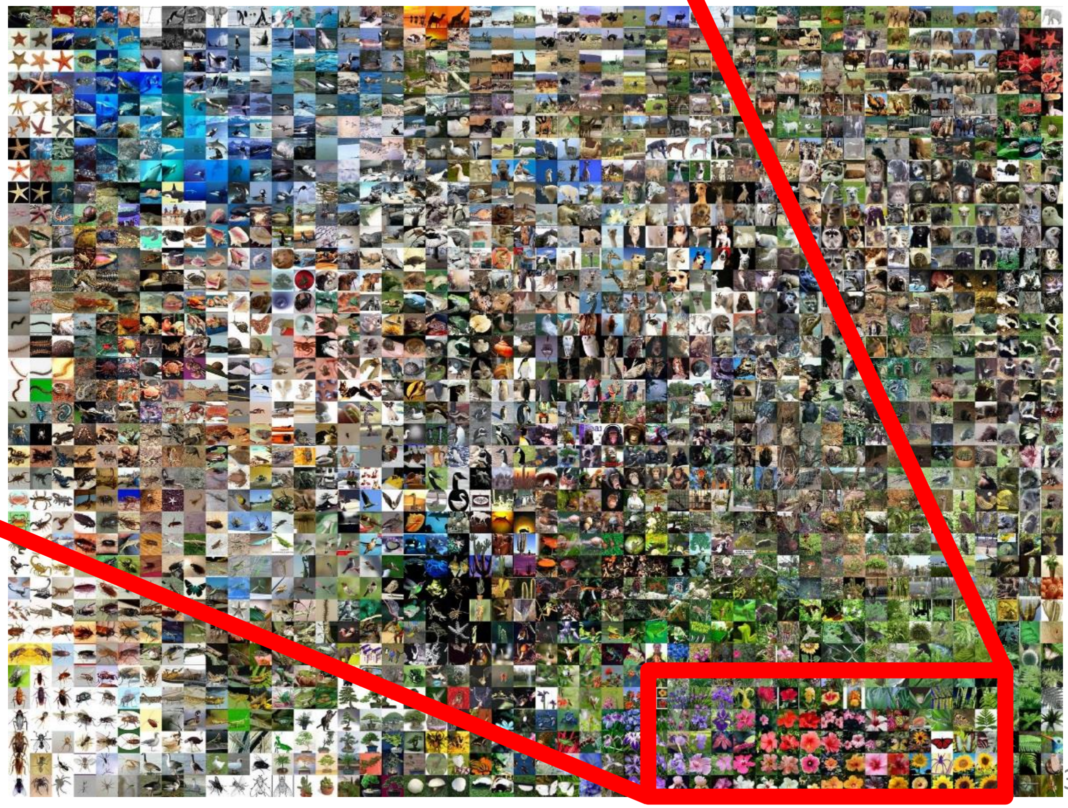
ImageNet



ImageNet

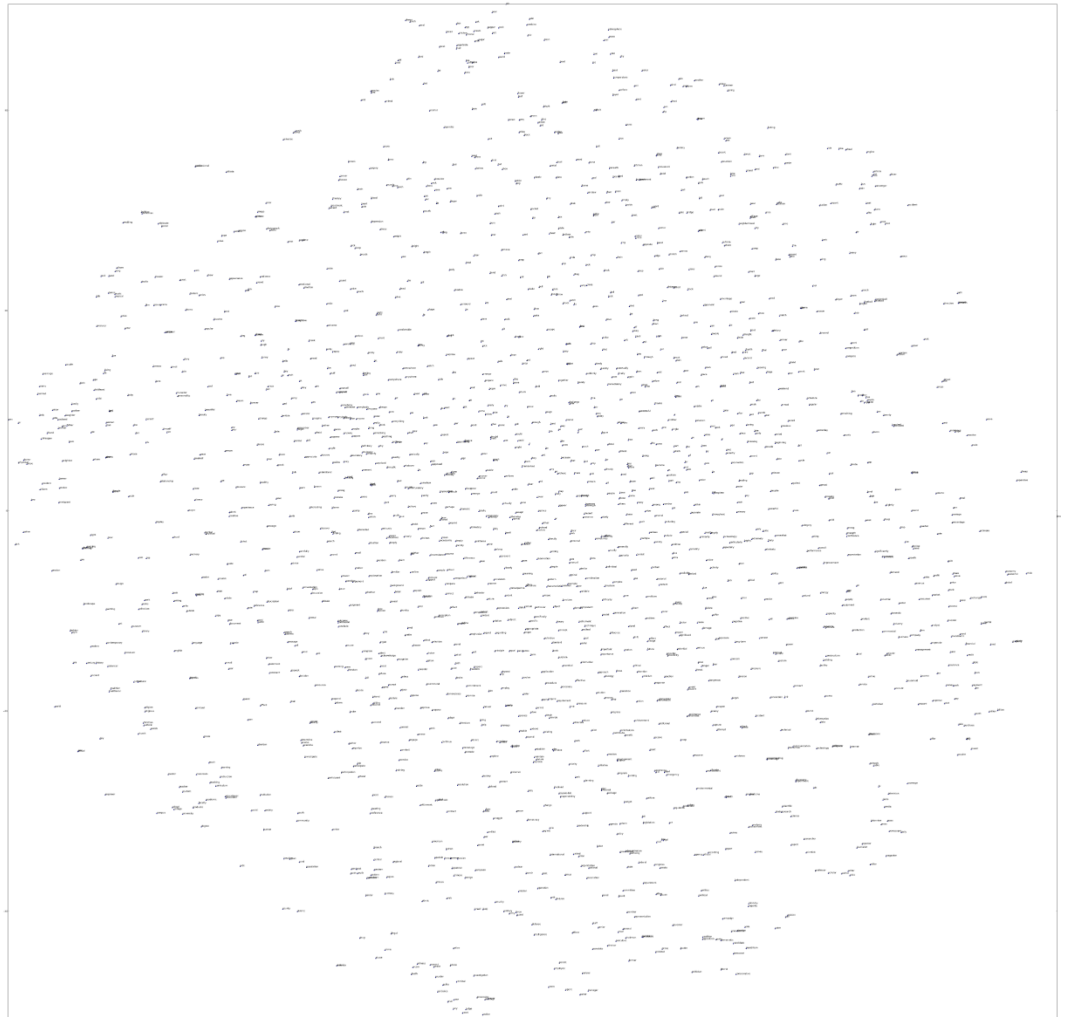
AlexNet



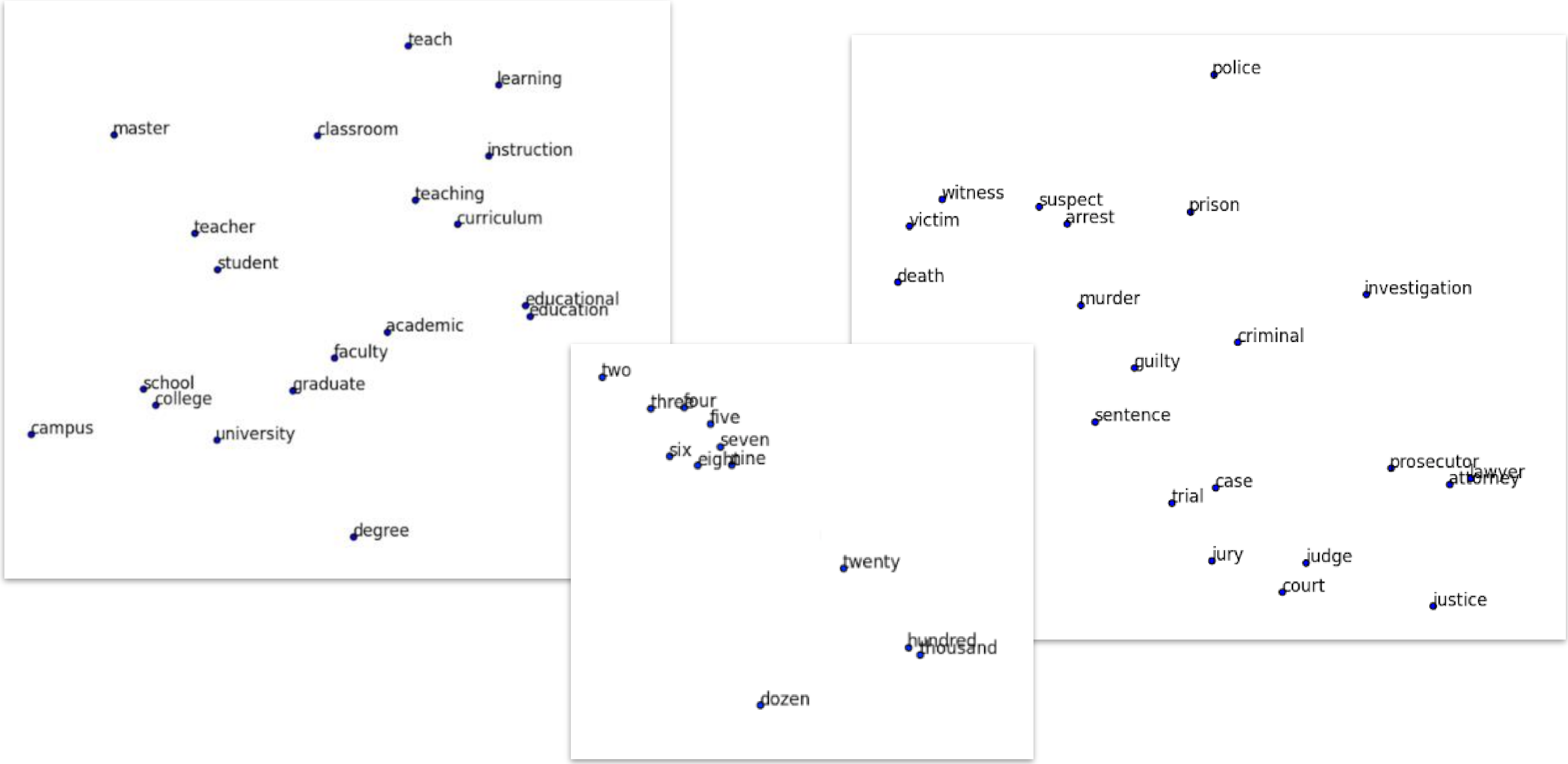


Word2vec

- Input: large corpus of text
- Embed words in to a high-dim space
  - Words with common contexts in the corpus are close in the space



Word2vec



http://nlp.yvespeirsman.be/blog/visualizing-word-embeddings-with-tsne/

Summary

Acquire Raw Data → Prepare / Clean Data, Visualization → Feature Engineering → Pick Model and Hyper-params for Task → Model Training / Optimization → Evaluate Model Performance → Deploy Model

- ✓ Data Visualization
- ✓ Non-linear Dimensionality Reduction
  - ✓ ISOMAP
  - ✓ LLE
  - ✓ t-SNE

Thanks!!

Questions?