CSE/ISyE 6740
Computational Data Analysis

Dimensionality Reduction

08/25/2025

Kai Wang, Assistant Professor in Computational Science and Engineering
kwang692@gatech.edu

Georgia Tech

Outline

- **Unsupervised Learning**
  - **Linear dimensionality reduction**
    - Principal component analysis
    - Eigenvalue decomposition
    - Reconstruction
  - **Non-linear dimensionality reduction**
    - Isomap
    - How Isomap works?
    - Other non-linear dimensionality reduction techniques

Matrix and Vector Convention

- A data point (feature) is always a column vector in $\mathbb{R}^d$, with dimensionality d

$$x^i = \begin{bmatrix} x_1^i \\ x_2^i \\ \dots \\ x_d^i \end{bmatrix} \in \mathbb{R}^d$$

- Feature matrix is a $n \times d$ matrix, concatenating n data points (transposed)

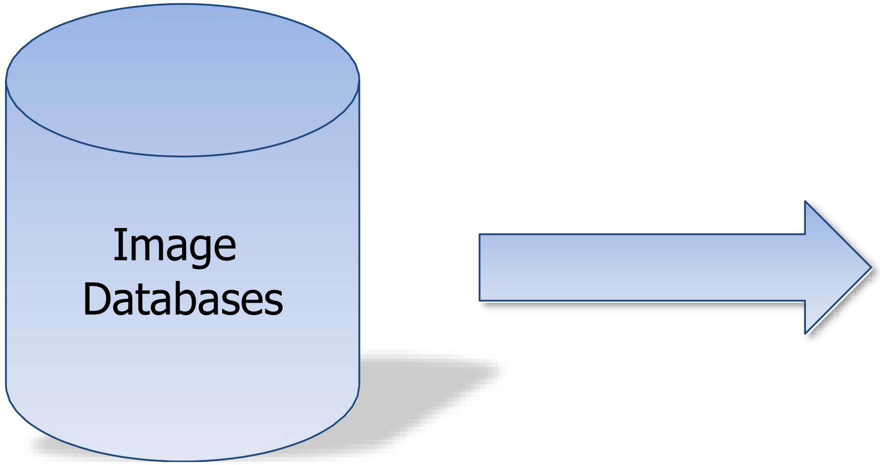
$$X = [x^1, x^2, \dots, x^n]^\top = \begin{bmatrix} {x^1}^\top \\ {x^2}^\top \\ \dots \\ {x^n}^\top \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_d^1 \\ x_1^2 & x_2^2 & \dots & x_d^2 \\ \dots & \dots & & \dots \\ x_1^n & x_2^n & \dots & x_d^n \end{bmatrix}$$

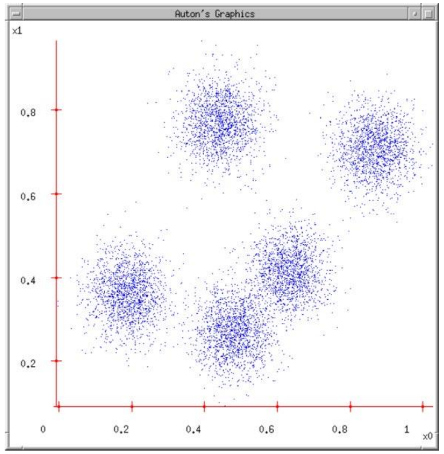
Data point #1

Feature #2

Dimensionality Reduction

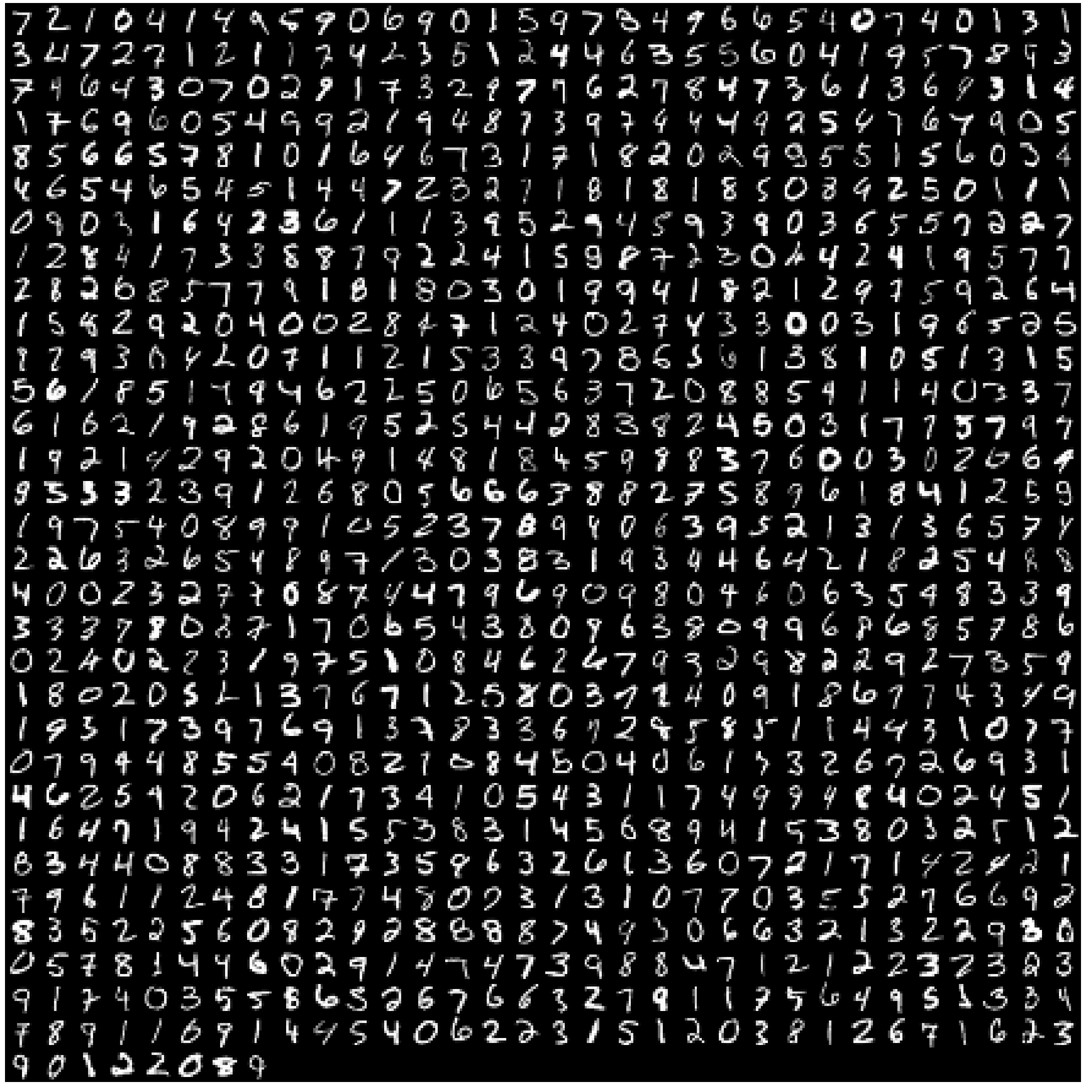
Image Databases



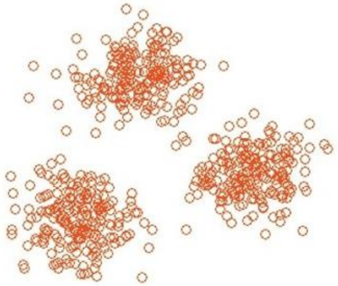


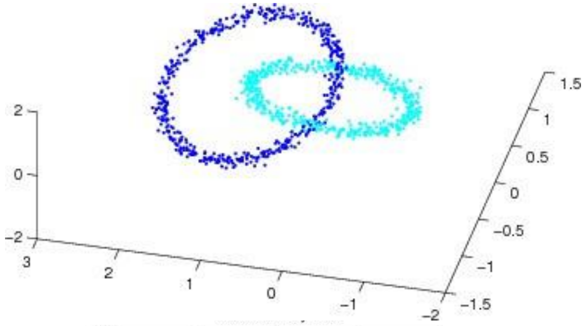
- What are the desired outcome?
- What are the input (data)?
- What are the learning paradigms?

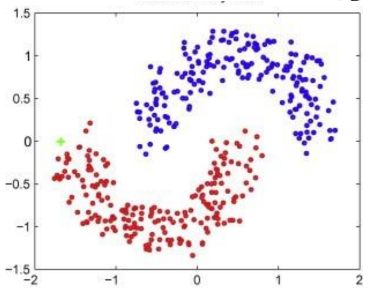
Handwritten Digits



What are the relations between data points?

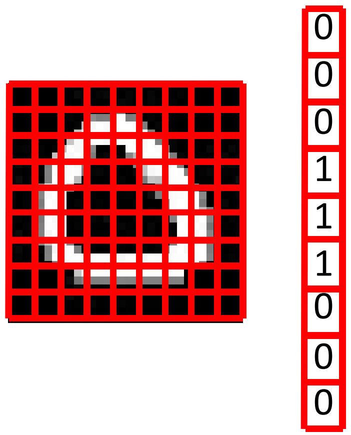






What is Dimensionality Reduction?

- The process of reducing the number of random variables under consideration
  - One can combine, transform or select variables
  - One can use linear or nonlinear operations



Vector in $\mathbb{R}^d$

Original data point

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_d \end{pmatrix}$$

$$f(x): \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$$

Where $d' \ll d$

Reduced representation

$$x = \begin{pmatrix} z_1 \\ z_2 \\ \dots \\ z_{d'} \end{pmatrix}$$

Why Dimensionality Reduction and How to Think?

- The dimension-reduced data can be used for
  - Visualization
  - Aggregating weak signals in the data
  - Cleaning the data
  - Speeding up subsequent learning task
  - Simplify model

- Key questions of a dimensionality reduction algorithm
  - What is the criterion for carrying out the reduction process?
  - What are the algorithm steps?

Principal Component Analysis

- Given n data points, $\{x^1, x^2, \dots, x^n\} \in \mathbb{R}^d$

- **Step 1**: estimate the mean and covariance matrix from the data:
  - $\mu = \frac{1}{n}\sum_{i=1}^{n} x^i$ and $C = \frac{1}{n}\sum_{i=1}^{n}(x^i - \mu)(x^i - \mu)^\top$

- **Step 2**: take the eigenvectors $w^1, w^2, \dots$of C corresponding to the largest eigenvalue λ_1, the second largest eigenvalue λ_2,…

- **Step 3**: compute reduced representation

$$z^i = \begin{pmatrix} {w^1}^\top (x^i - \mu)/\sqrt{\lambda_1} \\ {w^2}^\top (x^i - \mu)/\sqrt{\lambda_2} \\ \dots \\ {w^{d'}}^\top (x^i - \mu)/\sqrt{\lambda_{d'}} \end{pmatrix}$$

Use What Criterion for Reduction?

- There are many criteria (geometric based, information theory based, etc.)

- **One criterion**: want to capture **variation** in data
  - Variations are “signals” or information in the data
  - Need to normalize each variables first

- In the process, also discover variables or dimensions highly **correlated**
  - Represent highly related phenomena
  - Combine them to form a stronger signal
  - Lead to simpler presentation

How to Formulate the Problem?

- Given n data points, $\{x^1, x^2, \ldots, x^n\} \in \mathbb{R}^d$, with their mean $\mu = \frac{1}{n}\sum_{i=1}^{n} x^i$

- Find a direction $w \in \mathbb{R}^d$, where $||w|| = 1$

- Such that the variance (or variation) of the data along direction w is maximized

$$\max_{w:||w||=1} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left(w^\top x^i - w^\top \mu\right)^2}_{\text{variance}}$$

Georgia Tech

Is it an Easy Optimization Problem?

- Manipulate the objective with linear algebra

$$
\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\left(w^\top x^i - w^\top \mu\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(w^\top (x^i - \mu)\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n} w^\top (x^i - \mu)(x^i - \mu)^\top w \\
&= w^\top \underbrace{\left(\frac{1}{n}\sum_{i=1}^{n}(x^i - \mu)(x^i - \mu)^\top\right)}_{\text{Covariance matrix}} w
\end{aligned}
$$

Landscape of the Optimization Problem

- Suppose the data has two dimension ($d = 2$)

- C is a diagonal matrix

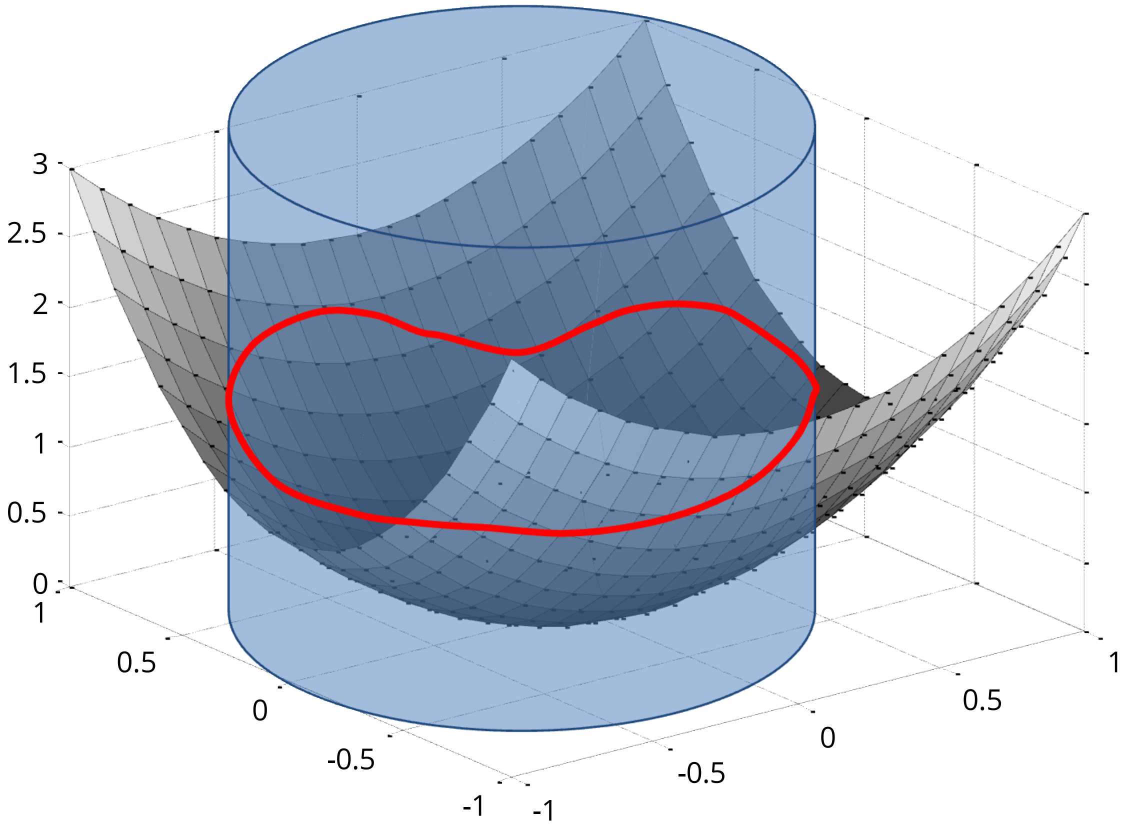
$$C = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

- The optimization problem becomes

$$\begin{aligned} &\max_{w:\, \|w\|=1} w^\top C w \\ &= \max_{w:\, \|w\|=1} (w_1, w_2) \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \\ &= \max_{w:\, \|w\|=1} w_1^2 + 2w_2^2 \end{aligned}$$

Landscape of the Optimization Problem

$f(w_1, w_2) = w_1^2 + 2w_2^2$



Eigenvalue Problem

- **Eigenvalue problem**
  - Given a symmetric matrix $C \in \mathbb{R}^{d \times d}$
  - Find a vector $w \in \mathbb{R}^d$ and $||w|| = 1$
  - Such that

$$Cw = \lambda w$$

- There will be multiple solution of $w^1, w^2, \dots, w^d$ with different $\lambda_1, \lambda_2, \dots \lambda_d$
  - They are orthonormal: ${w^i}^\top w^i = 1, {w^i}^\top w^j = 0$

Equivalent to Eigenvalue Problem

- **Claim**:

$$\max_{w:\|w\|=1} w^{\top} C w \Rightarrow C w = \lambda w$$

- **Proof**: Form the Lagrangian function of the optimization problem

$$L(w, \lambda) = \mathrm{w}^{\top} C w + \lambda(1 - \|w\|^2)$$

Necessary condition

- If w is a maximum of the original optimization problem, then there exists a λ, where (w, λ) is a **stationary point** of $L(w, \lambda)$.

- This implies that:

$$0 = \frac{\partial L}{\partial w} = 2Cw - 2\lambda w$$

Variance in the Principal Direction

- Principal direction w satisfies

$$Cw = \lambda w$$

- Variance in principal direction is

$$w^\top Cw$$
$$= \lambda w^\top w$$
$$= \lambda$$

Eigenvalue

Multiple Principal Directions

- Direction $w^1, w^2, \dots, w^d$, which has
  - the largest variances
  - but are also **orthogonal** to each other
- Take the eigenvectors $w^1, w^2, \dots, w^d$ of C corresponding to
  - The largest eigenvalue λ_1, the second largest eigenvalue λ_2, …

Principal Component Analysis (Revisit)

- Given m data points, $\{x^1, x^2, \dots, x^n\} \in \mathbb{R}^d$

- **Step 1**: estimate the mean and covariance matrix from the data:
  - $\mu = \frac{1}{n}\sum_{i=1}^{n} x^i$ and $C = \frac{1}{n}\sum_{i=1}^{n}(x^i - \mu)(x^i - \mu)^\top$

Principal directions

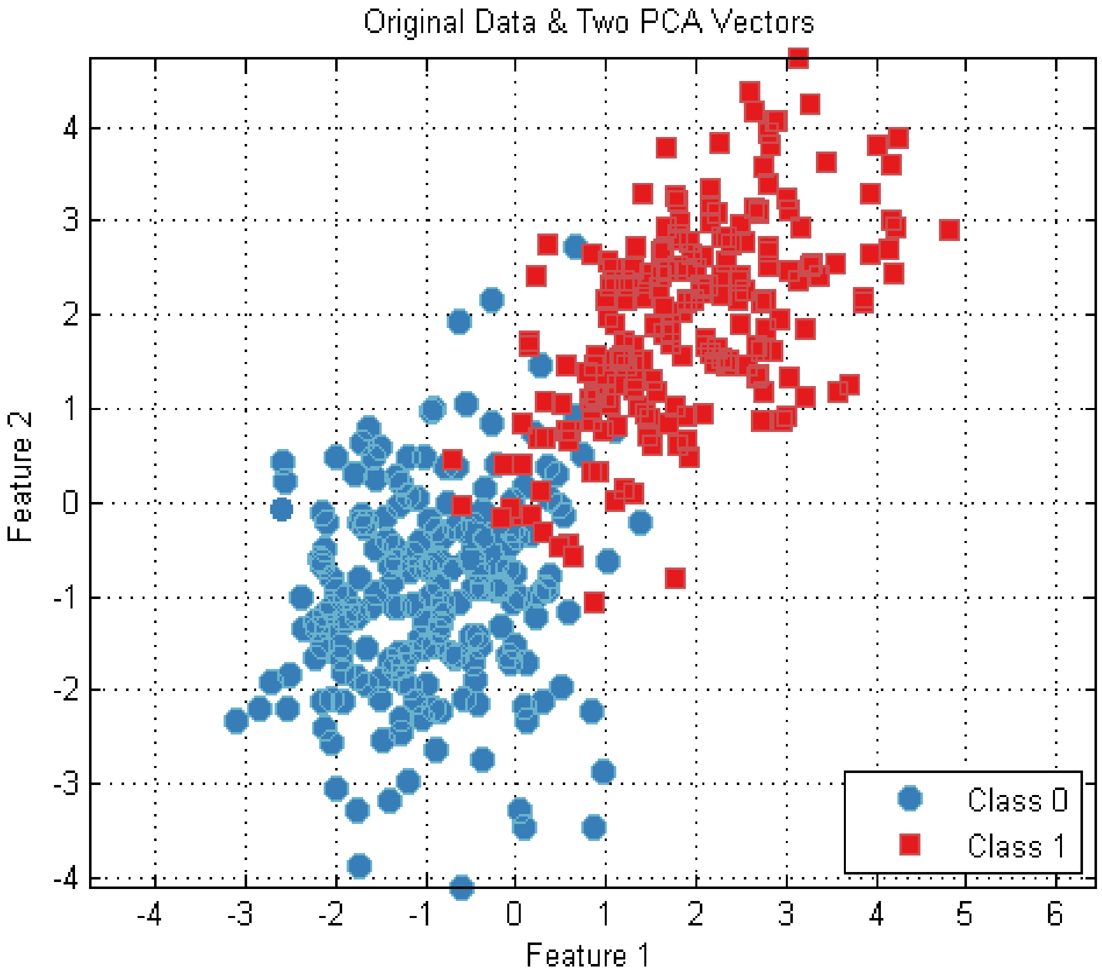
- **Step 2**: take the eigenvectors $w^1, w^2, \dots, w^{d'}$ of C corresponding to the largest eigenvalue λ_1, the second largest eigenvalue λ_2,…

- **Step 3**: compute reduced representation

$$z^i = \begin{pmatrix} {w^1}^\top (x^i - \mu)/\sqrt{\lambda_1} \\ {w^2}^\top (x^i - \mu)/\sqrt{\lambda_2} \\ \dots \\ {w^{d'}}^\top (x^i - \mu)/\sqrt{\lambda_{d'}} \end{pmatrix}$$

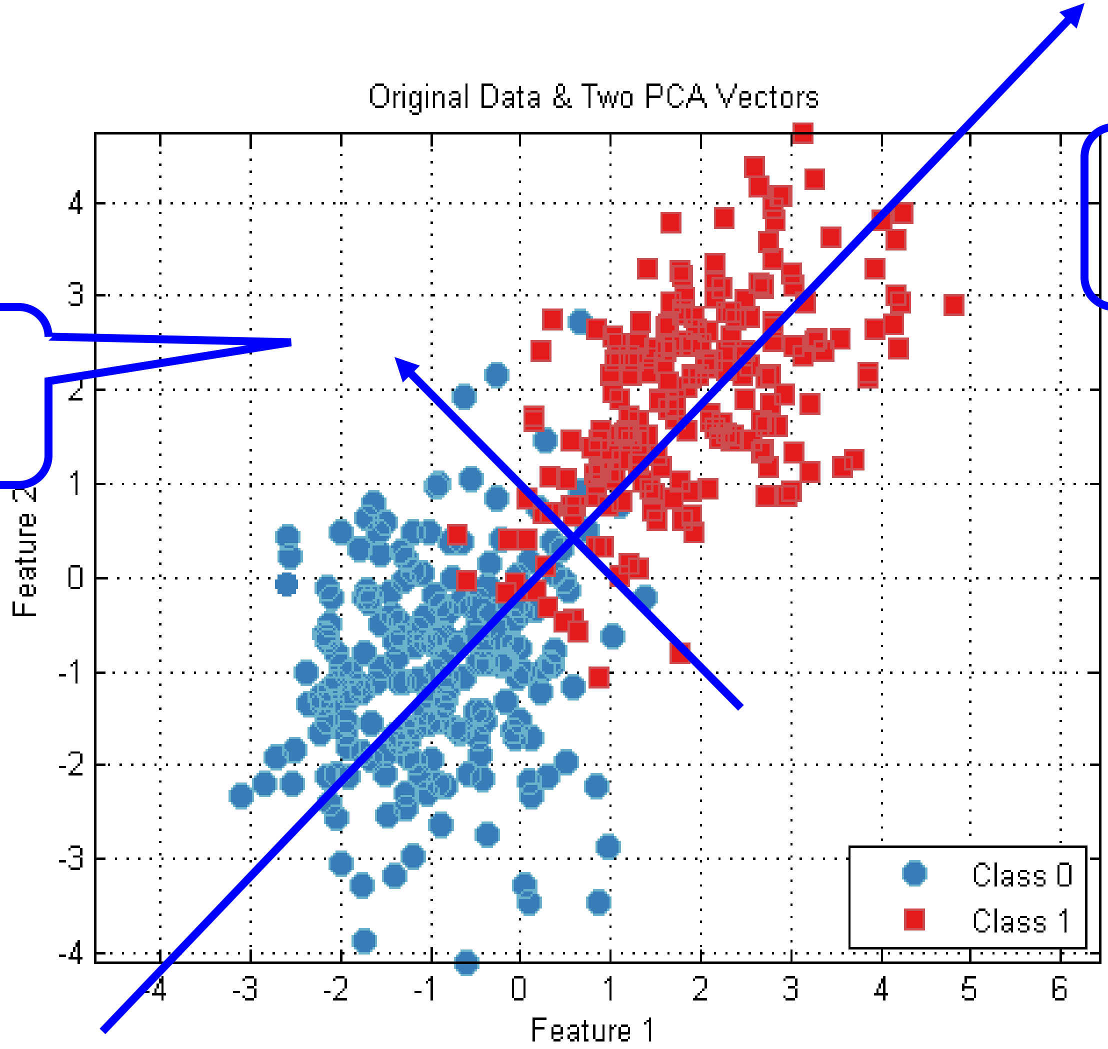
Normalize by standard deviation

An Example


Original Data & Two PCA Vectors
Feature 2
Feature 1
4
3
2
1
0
-1
-2
-3
-4
-4
-3
-2
-1
0
1
2
3
4
5
6
Class 0
Class 1

An Example

Data varies less in this direction

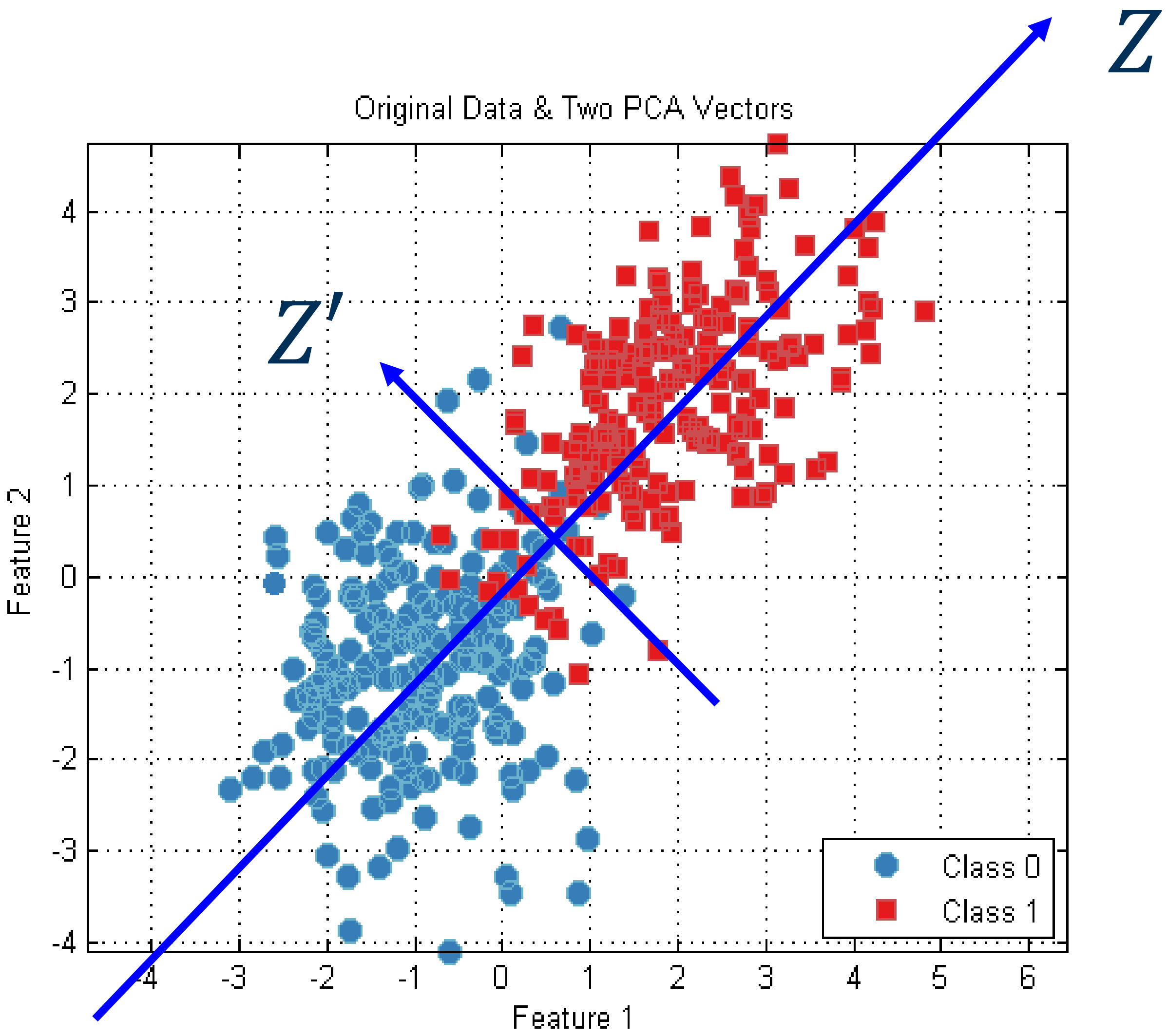


Data varies more in this direction

Two features are correlated

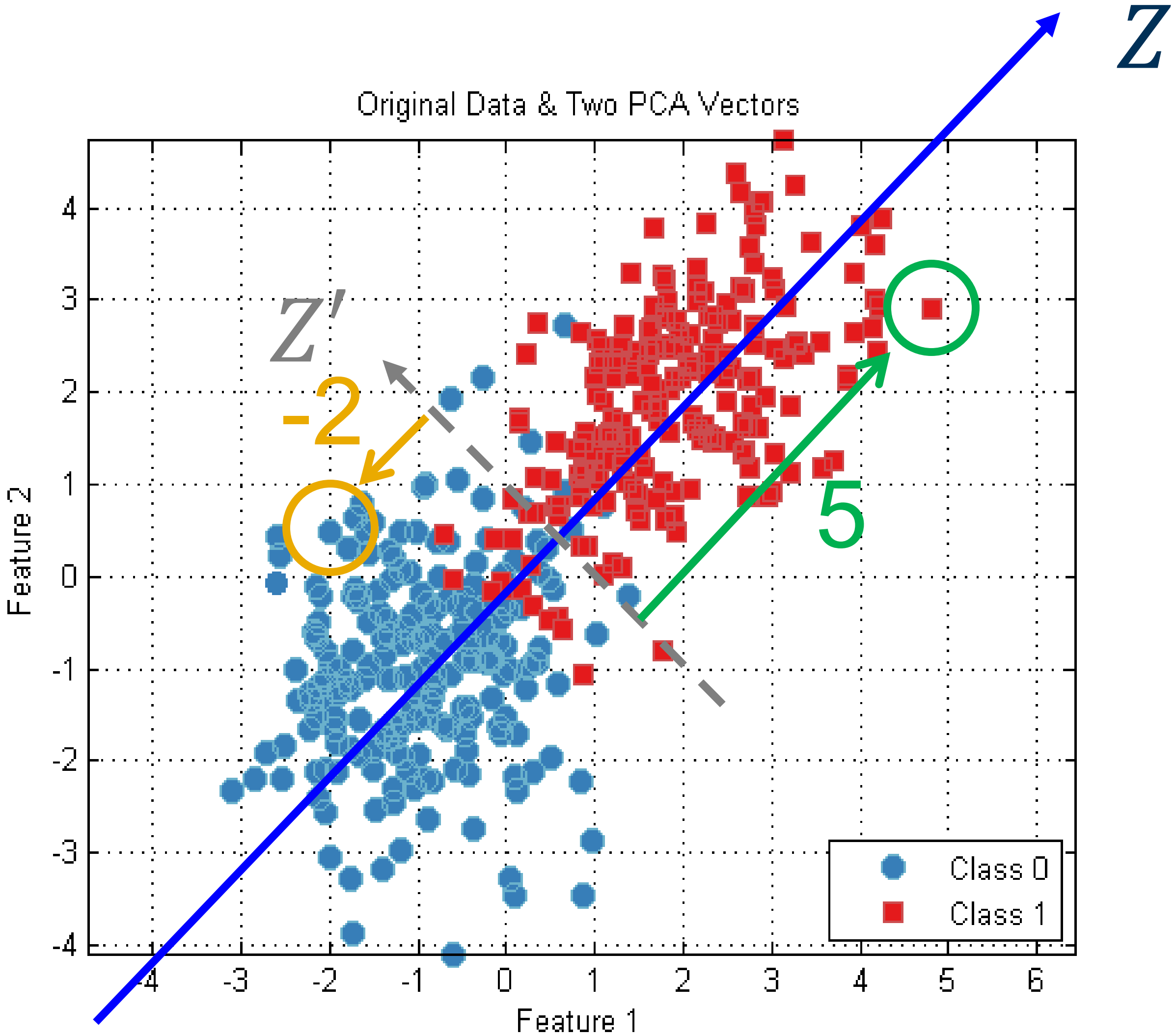
Georgia Tech

Principal Directions of the Data

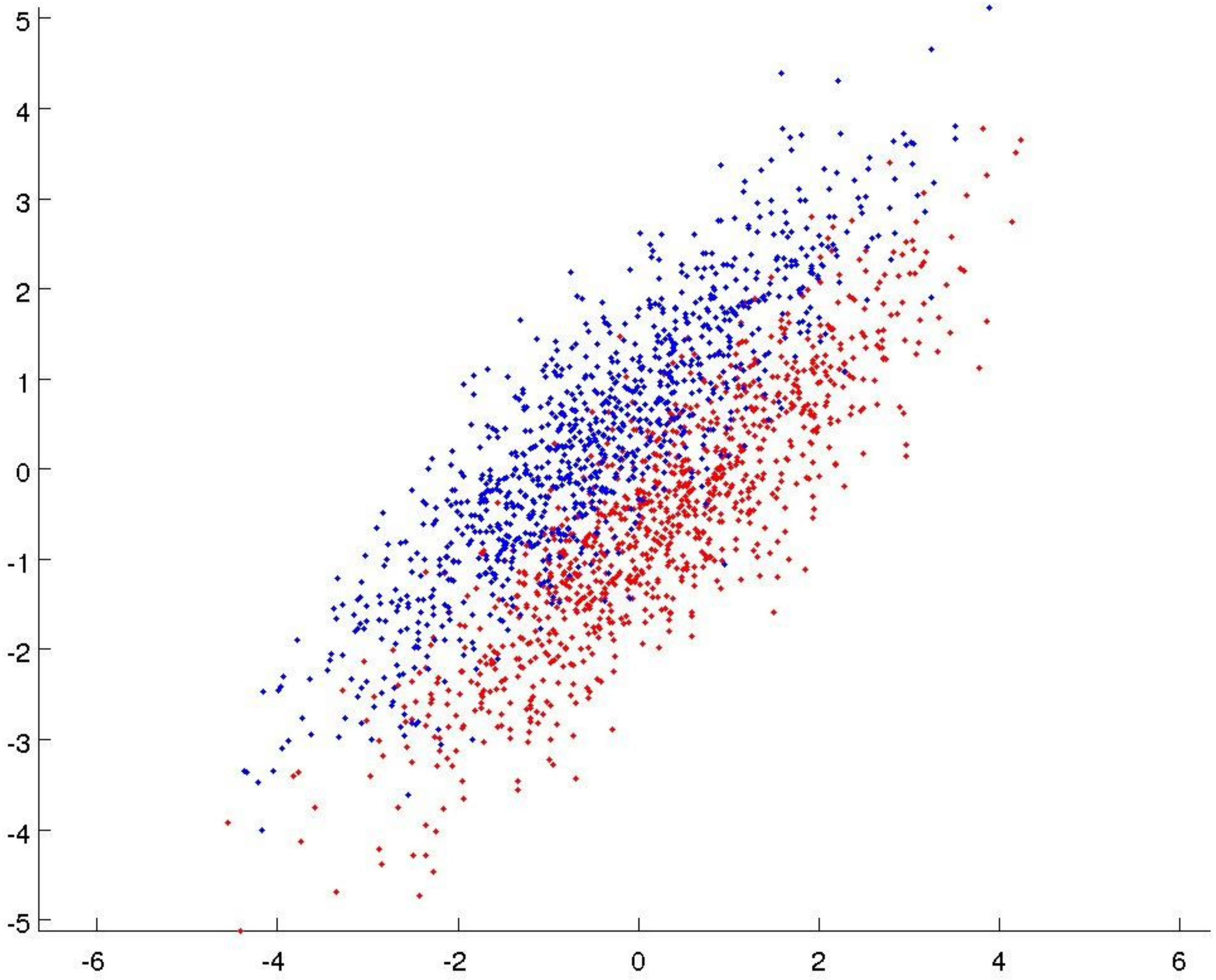


Georgia Tech

Reduce to 1 Dimension



Are Principal Components Good for Classification?



When to Use PCA?

- **Visualization**: reduce dimension to 2 or 3 dimensions so that you can plot
- **Feature distribution**: analyze variance, mean, distribution, etc.
- **Feature engineering**: identify independent principal directions, reduce # of features (e.g., # features >> # data)
- **Data compression**
- **Drawbacks**:
  - Lose interpretability
  - Larger variance $\neq$ more information/predictability
  - Label agnostic, i.e., may not fit labels well

PCA on Leaves

Input Features (representation)

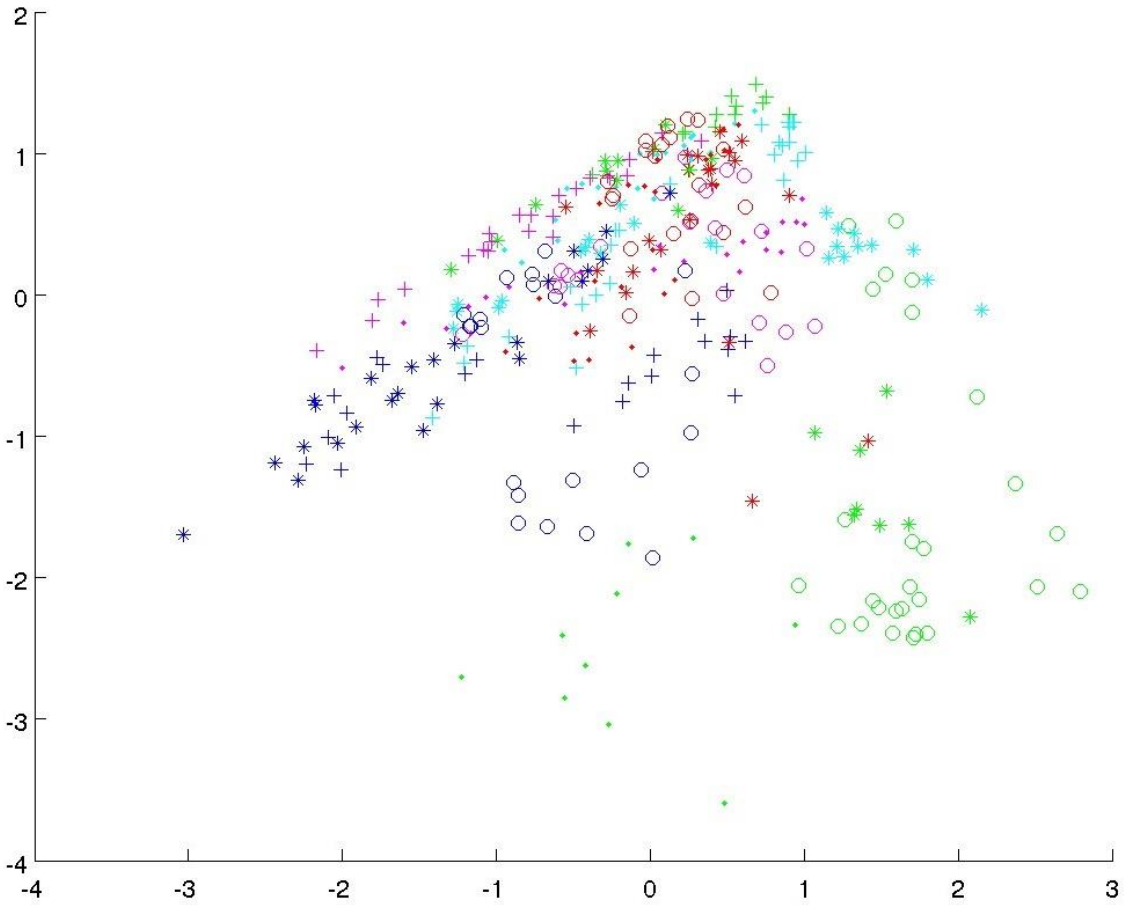
Shape feature	Description
Eccentricity	Eccentricity of the ellipse with identical second moments to I. This value ranges from 0 to 1.
Aspect Ratio	Consider any $X, Y \in \partial I$. Choose X and Y such that $d(X,Y) = D(I)$. Find $Z, W \in \partial I$ maximizing $D^{\perp} = d(Z,W)$ on the set of all pairs of ∂I that define a segment orthogonal to $[XY]$. The aspect ratio is defined as the quotient $D(I)/D^{\perp}$. Values close to 0 indicate an elongated shape.
Elongation	Compute the maximum escape distance $d_{\max} = \max_{X \in I} d(X, \partial I)$. Elongation is obtained as $1 - 2d_{\max}/D(I)$ and ranges from 0 to 1. The minimum is achieved for a circular region. Note that the ratio $2d_{\max}/D(I)$ is the quotient between the diameter of the largest inscribed circle and the diameter of the smallest circumscribed circle.
Solidity	The ratio $A(I)/A(H(I))$ is computed, which can be understood as a certain measure of convexity. It measures how well I fits a convex shape.
Stochastic Convexity	This variable extends the usual notion of convexity in topological sense, using sampling to perform the calculation. The aim is to estimate the probability of a random segment $[XY]$, $X, Y \in I$, to be fully contained in I.
Isoperimetric Factor	The ratio $4\pi A(I)/L(\partial I)^2$ is calculated. The maximum value of 1 is reached for a circular region. Curvy intertwined contours yield low values.
Maximal Indentation Depth	Let $C_{H(I)}$ and $L(H(I))$ denote the centroid and arclength of $H(I)$. The distances $d(X, C_{H(I)})$ and $d(Y, C_{H(I)})$ are computed $\forall X \in H(I)$ and $\forall Y \in \partial I$. The indentation function can then be defined as $[d(X, C_{H(I)}) - d(Y, C_{H(I)})]/L(H(I))$, which is sampled at one degree intervals. The maximal indentation depth $\mathfrak{D}$ is the maximum of this function.
Lobedness	The Fourier Transform of the indentation function above is computed after mean removal. The resulting spectrum is normalized by the total energy. Calculate lobedness as $F \times \mathfrak{D}^2$, where F stands for the smallest frequency at which the cumulated energy exceeds 80%. This feature characterizes how lobed a leaf is.

Texture feature	Description
Average Intensity	Average intensity is defined as the mean of the intensity i
Average Contrast	Average contrast is the the standard deviation of the inte age, $\sigma = \sqrt{\mu_2(z)}$.
Smoothness	Smoothness is defined as $R = 1 - 1/(1+\sigma^2)$ and mea relative smoothness of the intensities in a given region. Fo of constant intensity, R takes the value 0 and R approa regions exhibit larger disparities in intensity values. σ^2 is normalized by $(L-1)^2$ to ensure that $R \in [0,1]$.
Third moment	μ_3 is a measure of the intensity histogram's skewness. This is generally normalized by $(L-1)^2$ like smoothness.
Uniformity	Defined as $U = \sum_{i=0}^{L-1} p^2(z_i)$, uniformity's maximum reached when all intensity levels are equal.
Entropy	A measure of intensity randomness.

8 shape features
6 texture features

Georgia Tech

Reduce Representation



Principal directions: w_1 w_2

w_1	w_2
0.0938	0.1924
0.1902	0.0253
0.2266	-0.1800
-0.1850	0.4084
-0.1600	0.3825
-0.2063	0.3488
0.1940	-0.4037
0.2150	-0.3566
-0.3723	-0.2001
-0.3657	-0.1974
-0.3602	-0.2037
-0.3175	-0.1886
-0.3056	-0.1243
-0.3482	-0.1829

Shape features

Texture features

Georgia Tech

How to Recover the Original Data Point

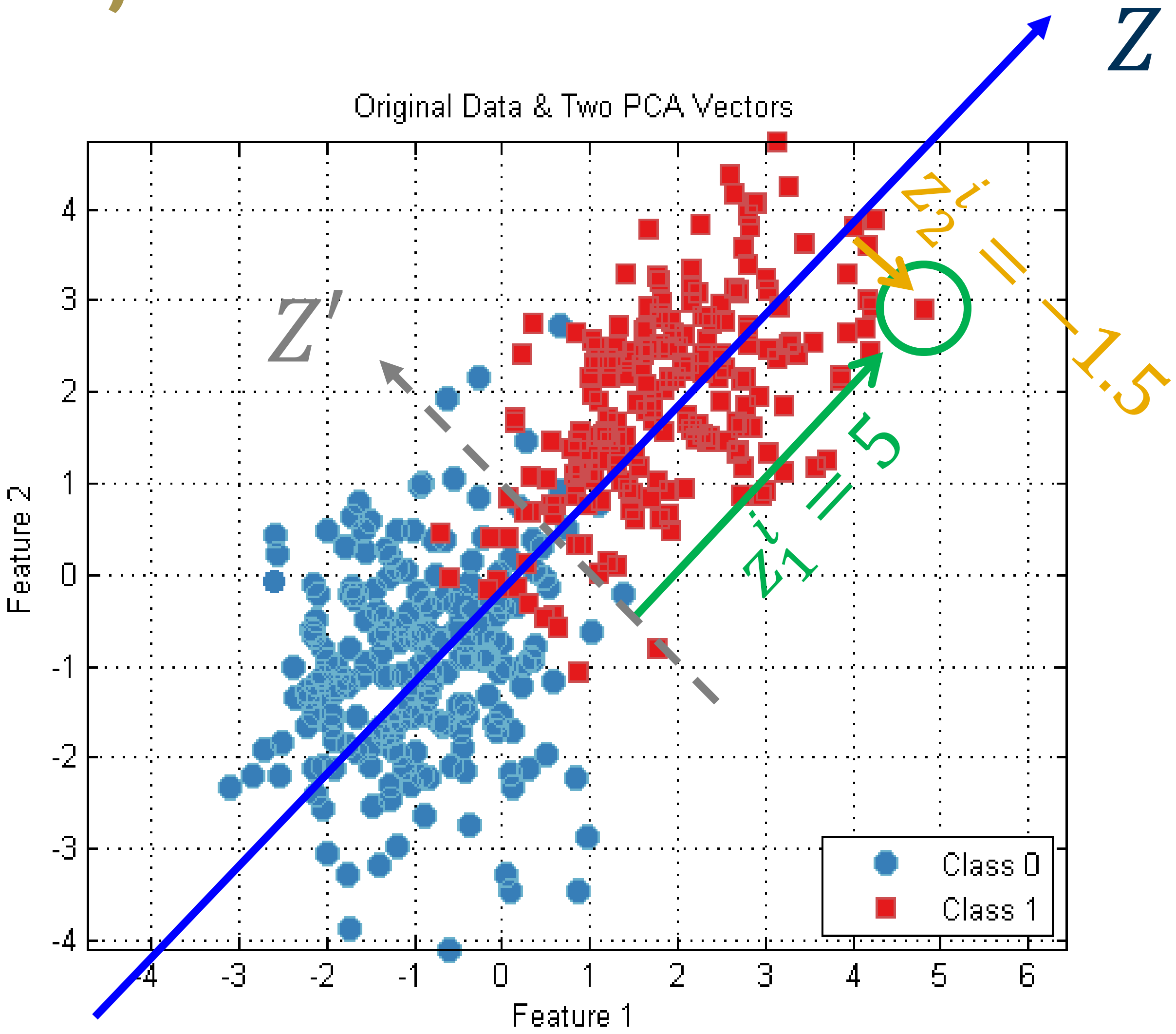
- Given data mean μ, principal directions $w^1, w^2, \dots, w^d$, and the corresponding eigenvalues $\lambda_1, \lambda_2, \dots$

- Can we recover x^i from the reduced representation z^i approximately?

$$z^i = \begin{pmatrix} z_1^i \\ z_2^i \\ \dots \\ z_{d'}^i \end{pmatrix} = \begin{pmatrix} {w^1}^\top (x^i - \mu)/\sqrt{\lambda_1} \\ {w^2}^\top (x^i - \mu)/\sqrt{\lambda_2} \\ \dots \\ {w^{d'}}^\top (x^i - \mu)/\sqrt{\lambda_{d'}} \end{pmatrix}$$

- $x^i \approx \hat{x}^i = \mu + z_1^i \cdot \sqrt{\lambda_1} \cdot w^1 + z_2^i \cdot \sqrt{\lambda_2} \cdot w^2 + \cdots$

Reduce to 1-Dim, Reconstruct 2-Dim

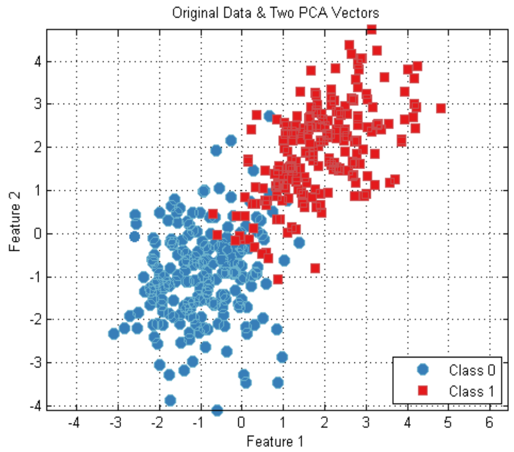


$$x^i \approx \hat{x}^i = \mu + \boxed{z_1^i \cdot \sqrt{\lambda_1} \cdot w^1} + \boxed{z_2^i \cdot \sqrt{\lambda_2} \cdot w^2} + \cdots$$

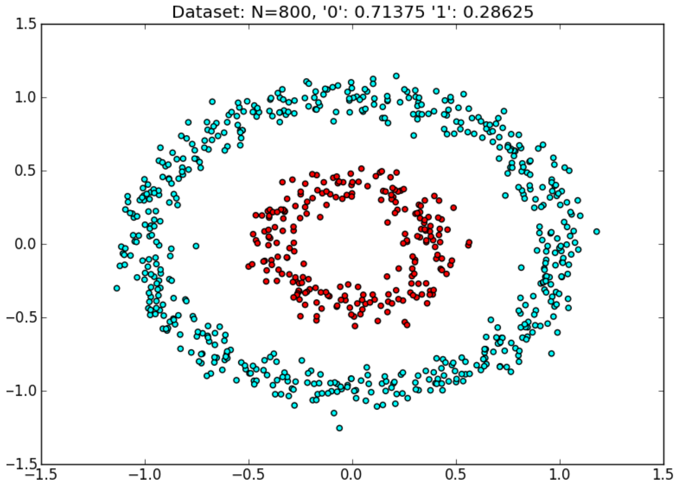
Non-linear Dimensionality Reduction

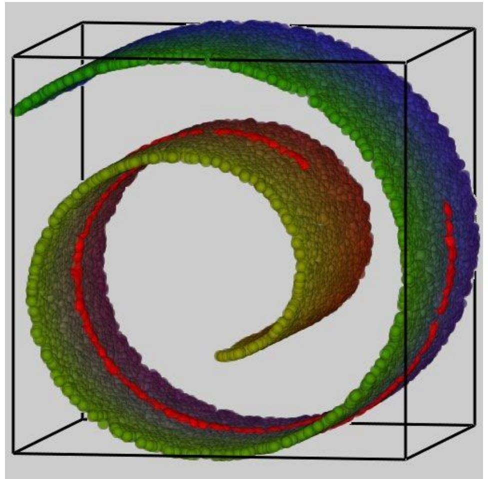
Limitation of PCA

- Suitable when variables are linearly correlated

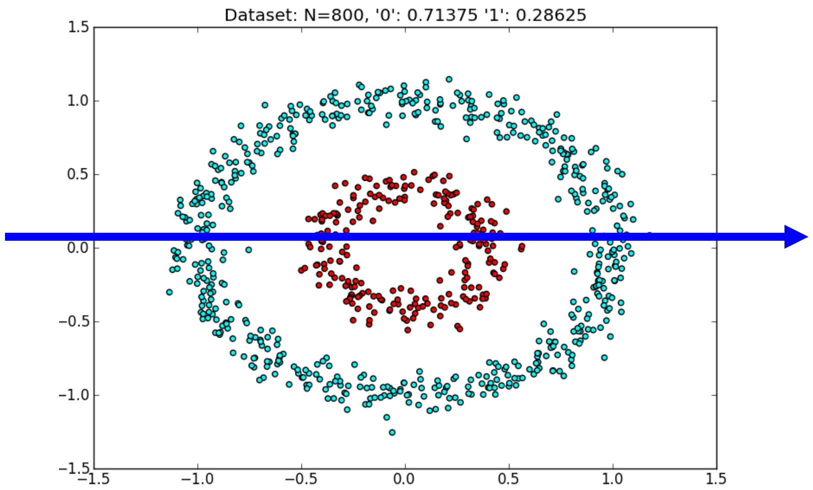


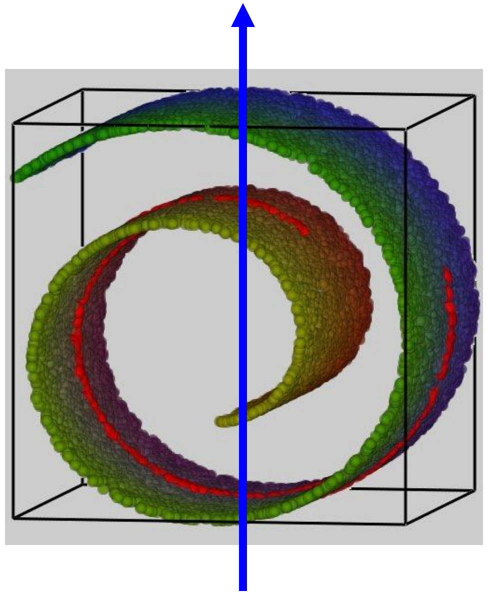
- Not suitable when nonlinear structures are present





What’s Wrong with PCA?

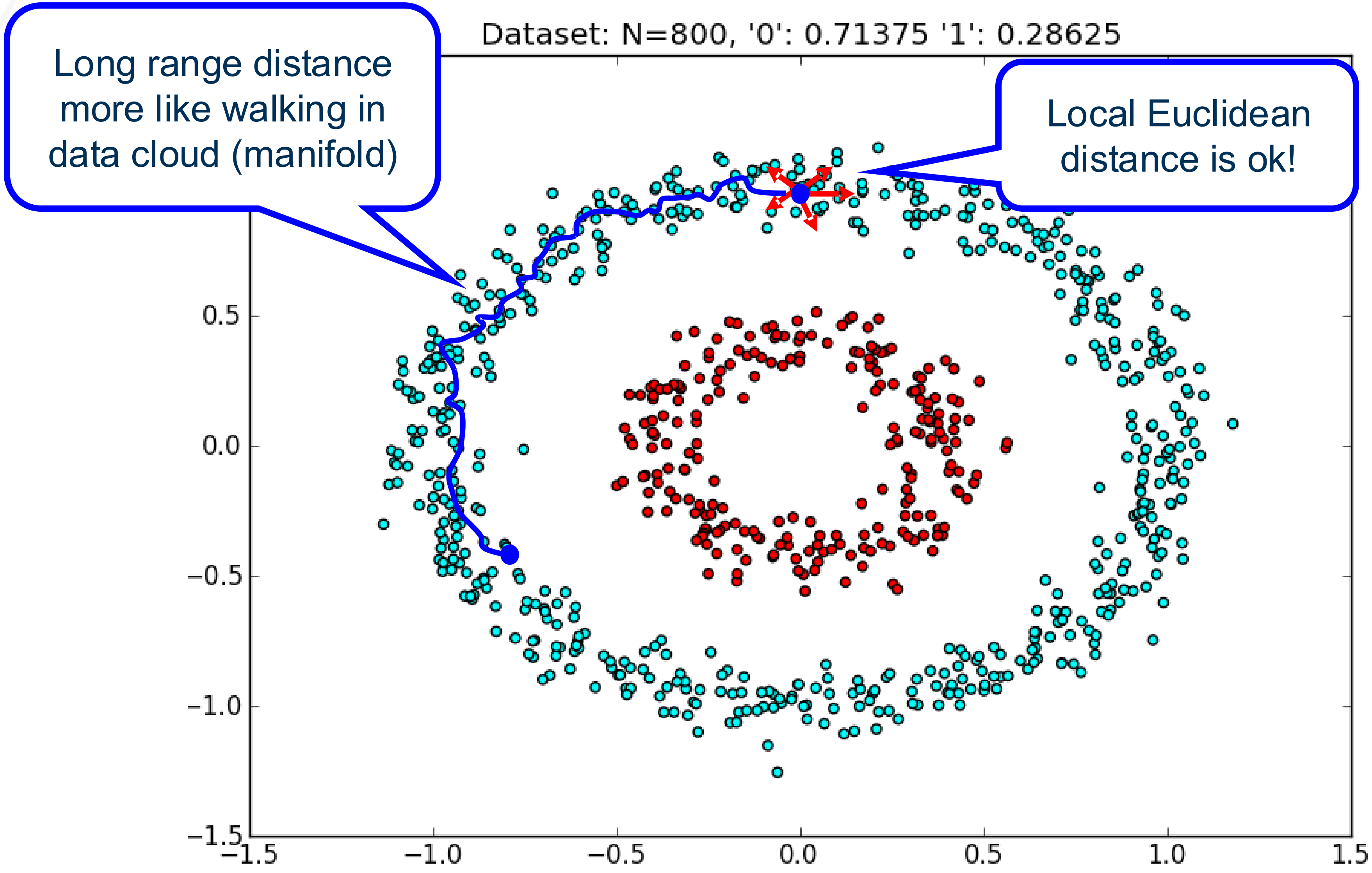




- PCA uses linear projection $w^{\top}x$, implicitly assuming Euclidean distance is the dissimilarity (distance) measure

- When there are nonlinear structure, Euclidean distance is **not** the right distance measure **globally**.

What’s a Reasonable Distance Measure?



Isomap

- Given n data points, $\{x^1, x^2, \dots, x^n\} \in \mathbb{R}^d$

- **Step 1**: build an adjacency matrix A using nearest neighbors, and compute pairwise shortest distance matrix D
- **Step 2**: use a centering matrix $H = I - \frac{1}{n} 1 1^\top$ to define covariance matrix

$$C = -\frac{1}{2} H (D)^2 H$$

Where $(D)^2 = \left(D_{ij}^2\right)_{i,j \in [1,2,\dots,n]}$

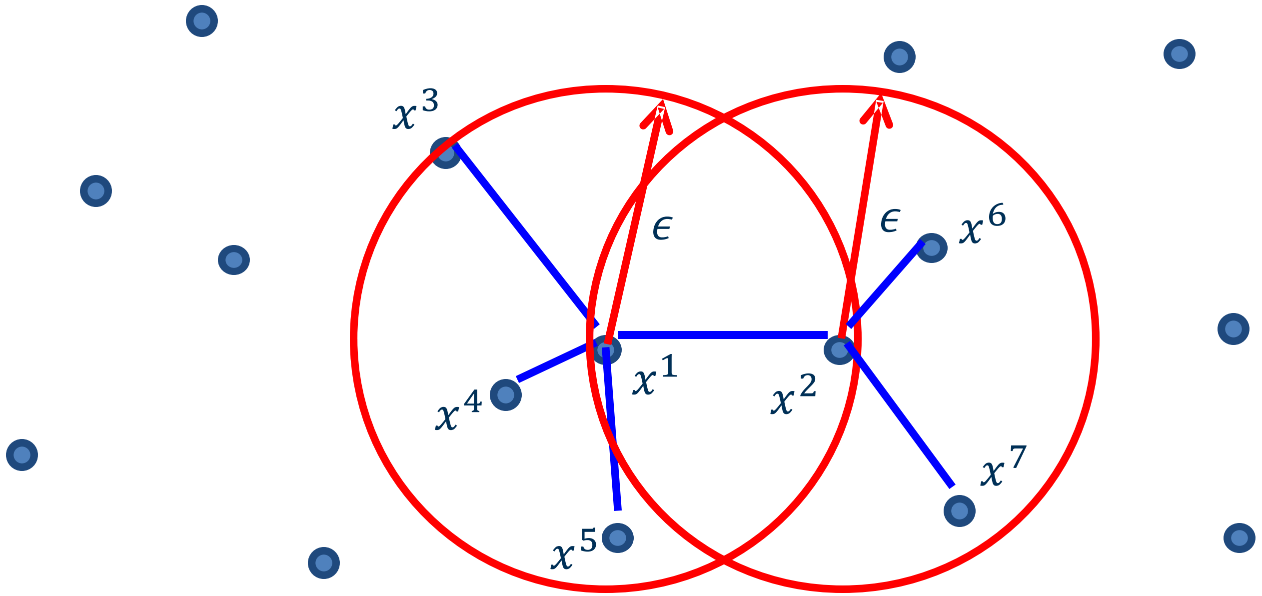
- **Step 3**: compute leading eigenvectors $w^1, w^2, \dots, w^{d'}$ and eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_{d'}$ of C

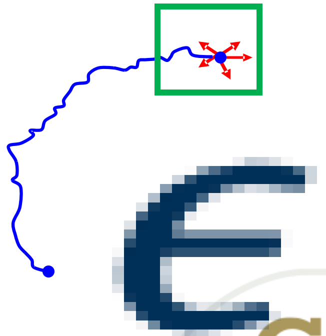
$$\tilde{Z} = \left(w^1, w^2, \dots, w^{d'}\right) \begin{pmatrix} \lambda_1^{1/2} & & \\ & \dots & \\ & & \lambda_{d'}^{1/2} \end{pmatrix}$$

Using Neighbor Graph to Define Distance

- Given n data points, threshold ϵ, construct adjacency matrix $A \in \mathbb{R}^{n\times n}$

$$A_{ij} = \begin{cases} 1, & \text{if } \|x^i - x^j\| \le \epsilon \\ 0, & \text{otherwise} \end{cases}$$



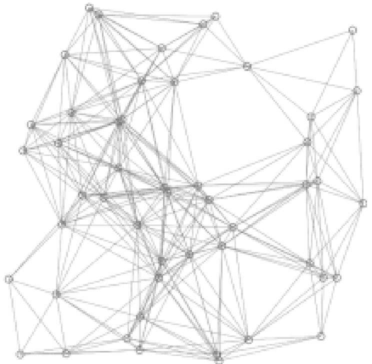


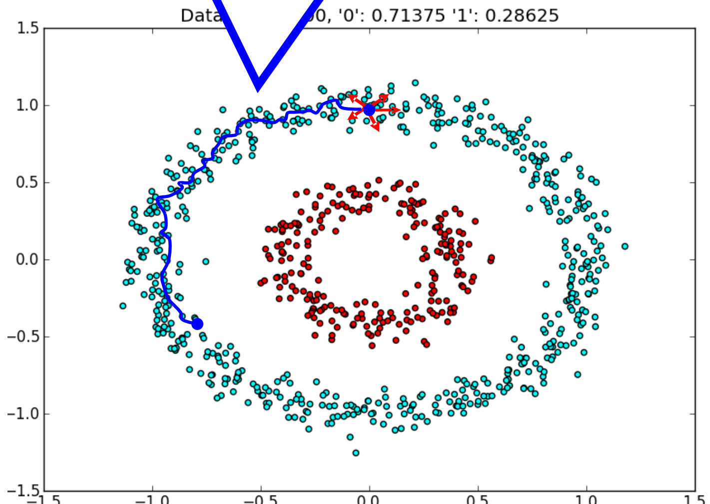
Shortest Path Distance

- With the graph defined by $A \in \mathbb{R}^{n \times n}$, find the shortest path distance matrix D between any pairs of points.
  - Aka. graph distance matrix

- The shortest path distance matrix D can be computed by:
  - Floyd-Warshall algorithm (all pair shortest path problem)
    - Cost: $O(|V|^3) = O(n^3)$
  - Dijkstra's algorithm * n
    - Cost: $O\big(n(|E| + |V|\log|V|)\big) = O(n|E| + n^2 \log n)$

How many steps to move from one point to another





Images from Wikipedia

From Distances to Reconstruct Representation

- **Goal**: Given the distance matrix D, find representation $z^i \in \mathbb{R}^{d'}$ $\forall i$ such that

$$D_{ij}^2 = \left\|z^i - z^j\right\|^2$$
$$= \left(z^i - z^j\right)^\top \left(z^i - z^j\right)$$
$$= {z^i}^\top z^i + {z^j}^\top z^j - 2{z^i}^\top z^j$$

- In matrix format, let $Z = (z^1, z^2, \dots, z^n)^\top \in \mathbb{R}^{n\times d'}$

$$(D)^2 = a1^\top + 1a^\top - 2ZZ^\top \in \mathbb{R}^{n\times n}. \quad \text{(pairwise distance)}$$

where $a = \left({z^1}^\top z^1, {z^2}^\top z^2, \dots, {z^n}^\top z^n\right)^\top$

Georgia Tech

From Distances to Reconstruct Representation

- Construct a special centering matrix $H = I - \frac{1}{n}11^\top$
  - Verify
    - $\left(I - \frac{1}{n}11^\top\right)1a^\top\left(I - \frac{1}{n}11^\top\right) = 0$
    - $\left(I - \frac{1}{n}11^\top\right)a1^\top\left(I - \frac{1}{n}11^\top\right) = 0$

- Then apply H to both side of $(D)^2$
  - $C = -\frac{1}{2}H(D)^2H = -\frac{1}{2}H(a1^\top + 1a^\top - 2ZZ^\top)H = HZZ^\top H$
  - $HZ = \left(I - \frac{1}{n}11^\top\right)Z = Z - \mu 1^\top = \tilde{Z}$
  - Ultimately we get $C = \tilde{Z}\tilde{Z}^\top$

Georgia Tech

Obtain Low-dimensional Representation

- Given $C = -\frac{1}{2}H(D)^2H = \tilde{Z}\tilde{Z}^\top$

- Perform eigenvalue decomposition on C
  - $Cw = \lambda w$
  - Take the eigenvectors w^1, w^2, … of C corresponding to
    - The largest eigenvalue λ_1, as the first coordinate
    - The second largest eigenvalue λ_2, as the second coordinate…

- Reduced representation

$$\tilde{Z} = \left(w^1, w^2, \dots, w^{d'}\right)\begin{pmatrix} \lambda_1^{1/2} & & \\ & \dots & \\ & & \lambda_{d'}^{1/2} \end{pmatrix}$$

Isomap

- Given n data points, $\{x^1, x^2, \dots, x^n\} \in \mathbb{R}^d$

- **Step 1**: build a weighted graph A using nearest neighbors, and compute pairwise shortest distance matrix D
- **Step 2**: use a centering matrix $H = I - \frac{1}{n}11^\top$ to define covariance matrix

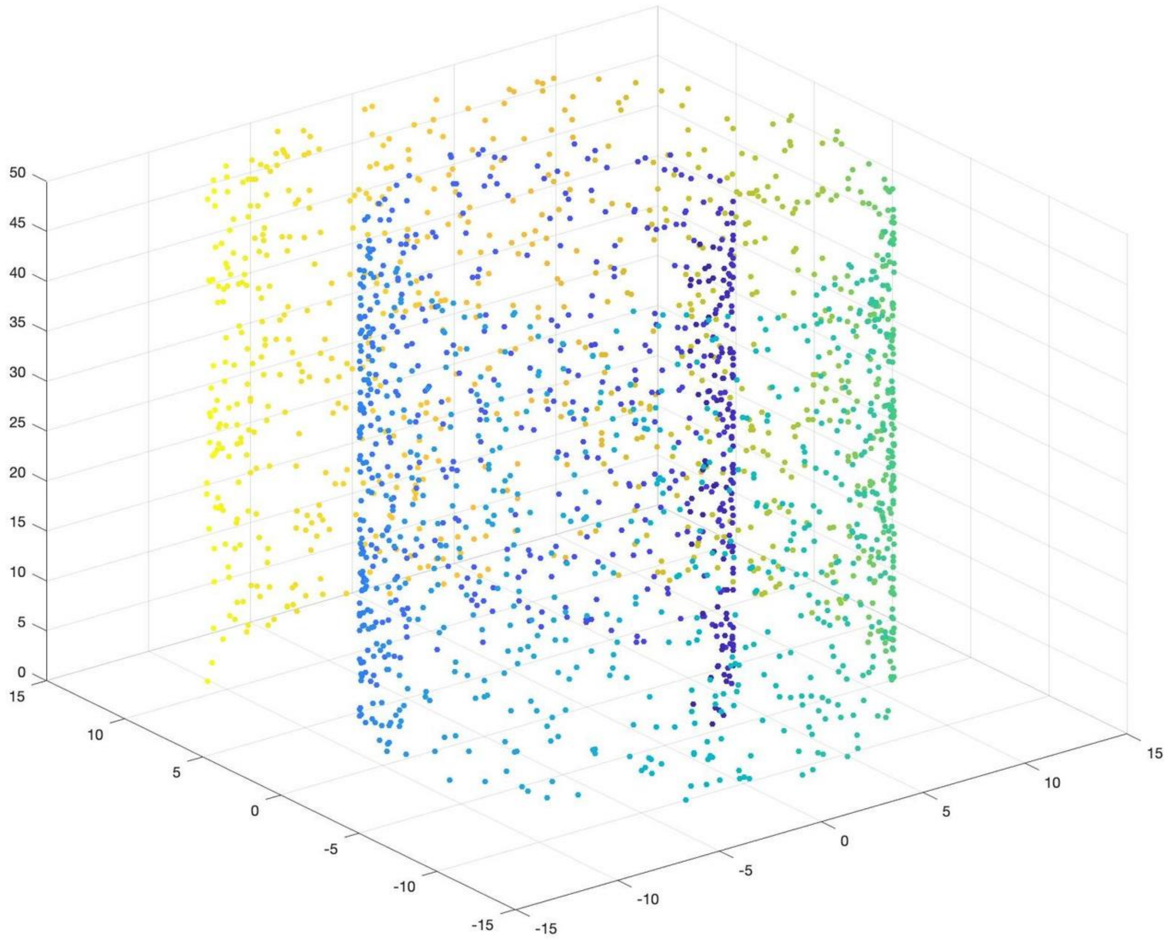
$$C = -\frac{1}{2}H(D)^2H$$

Where $(D)^2 = \left(D_{ij}^2\right)_{i,j \in [1,2,\dots,n]}$

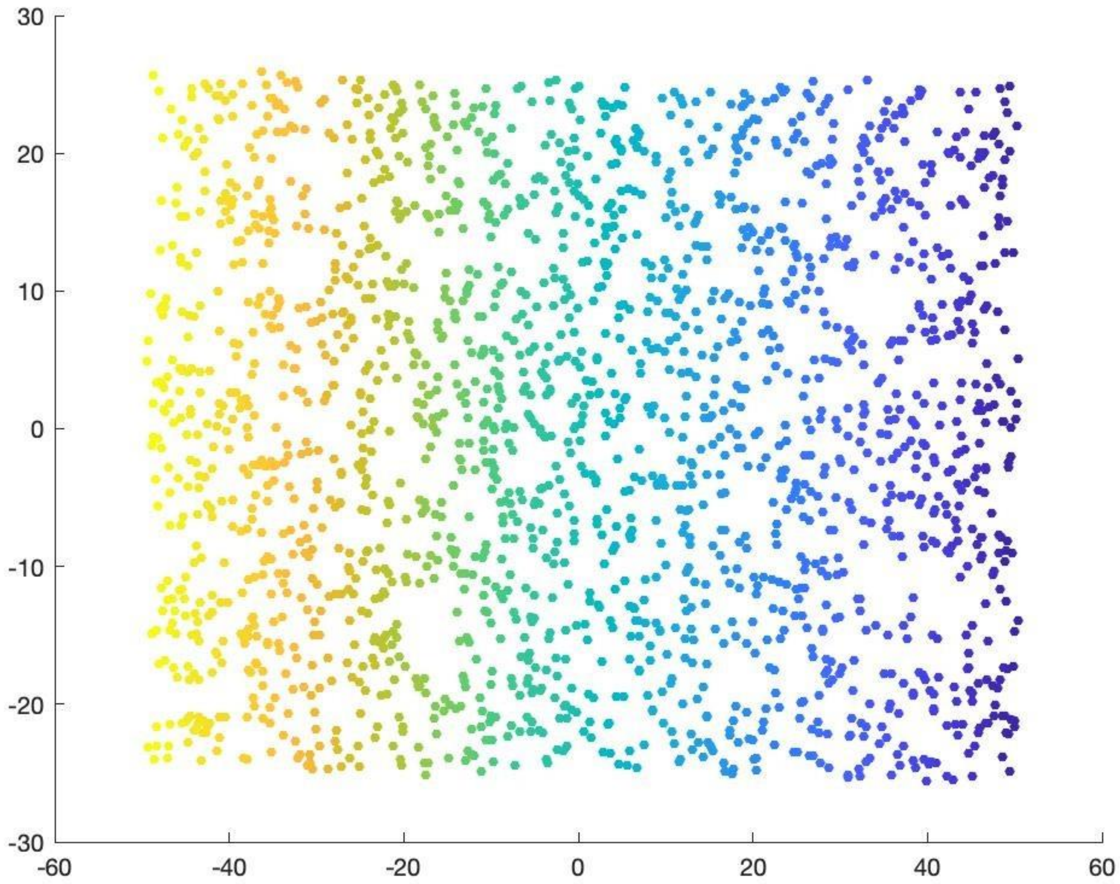
- **Step 3**: compute leading eigenvectors $w^1, w^2, \dots, w^{d'}$ and eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_{d'}$ of C

$$\tilde{Z} = \left(w^1, w^2, \dots, w^{d'}\right)\begin{pmatrix} \lambda_1^{1/2} & & \\ & \dots & \\ & & \lambda_{d'}^{1/2} \end{pmatrix}$$

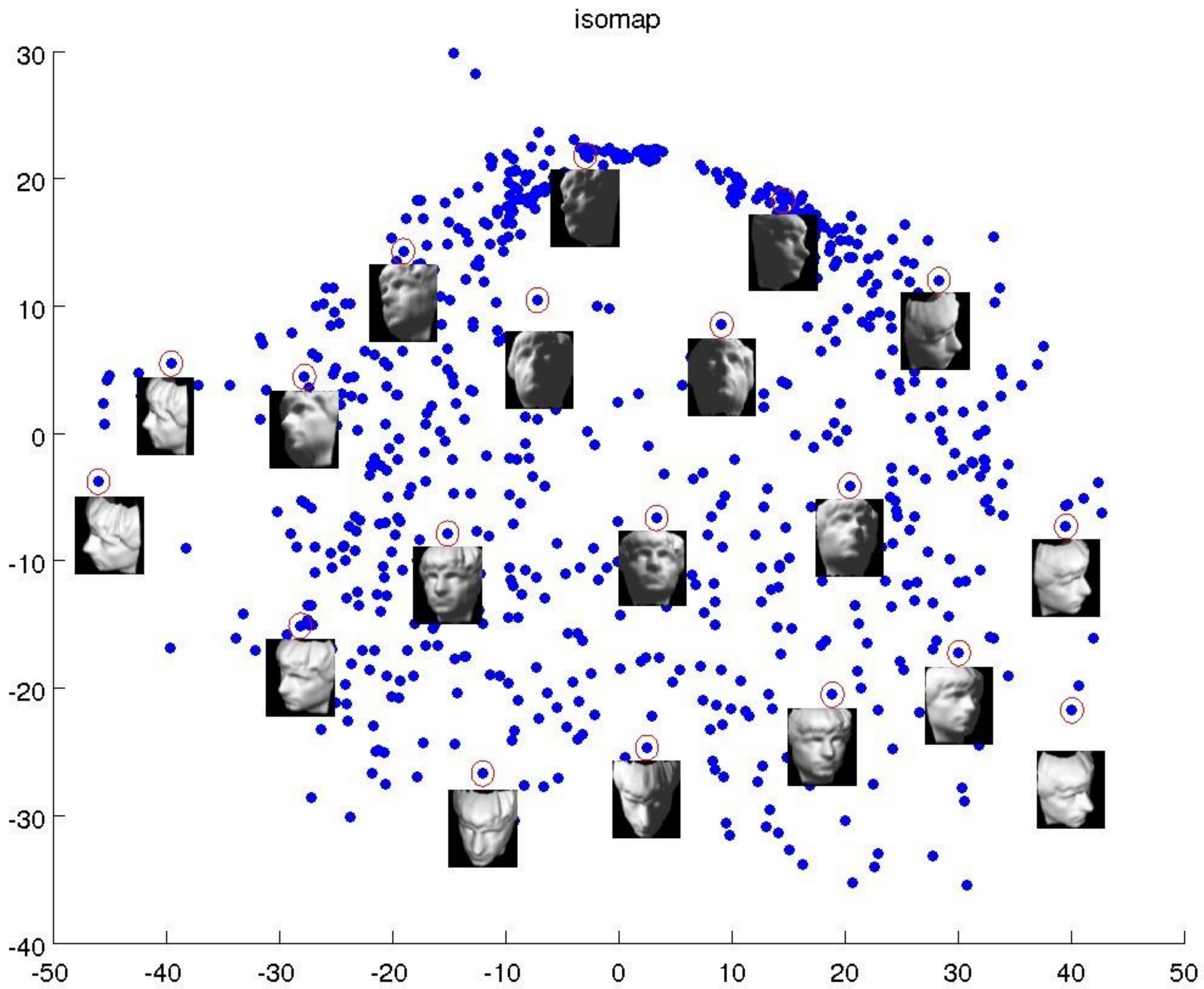
Swissroll



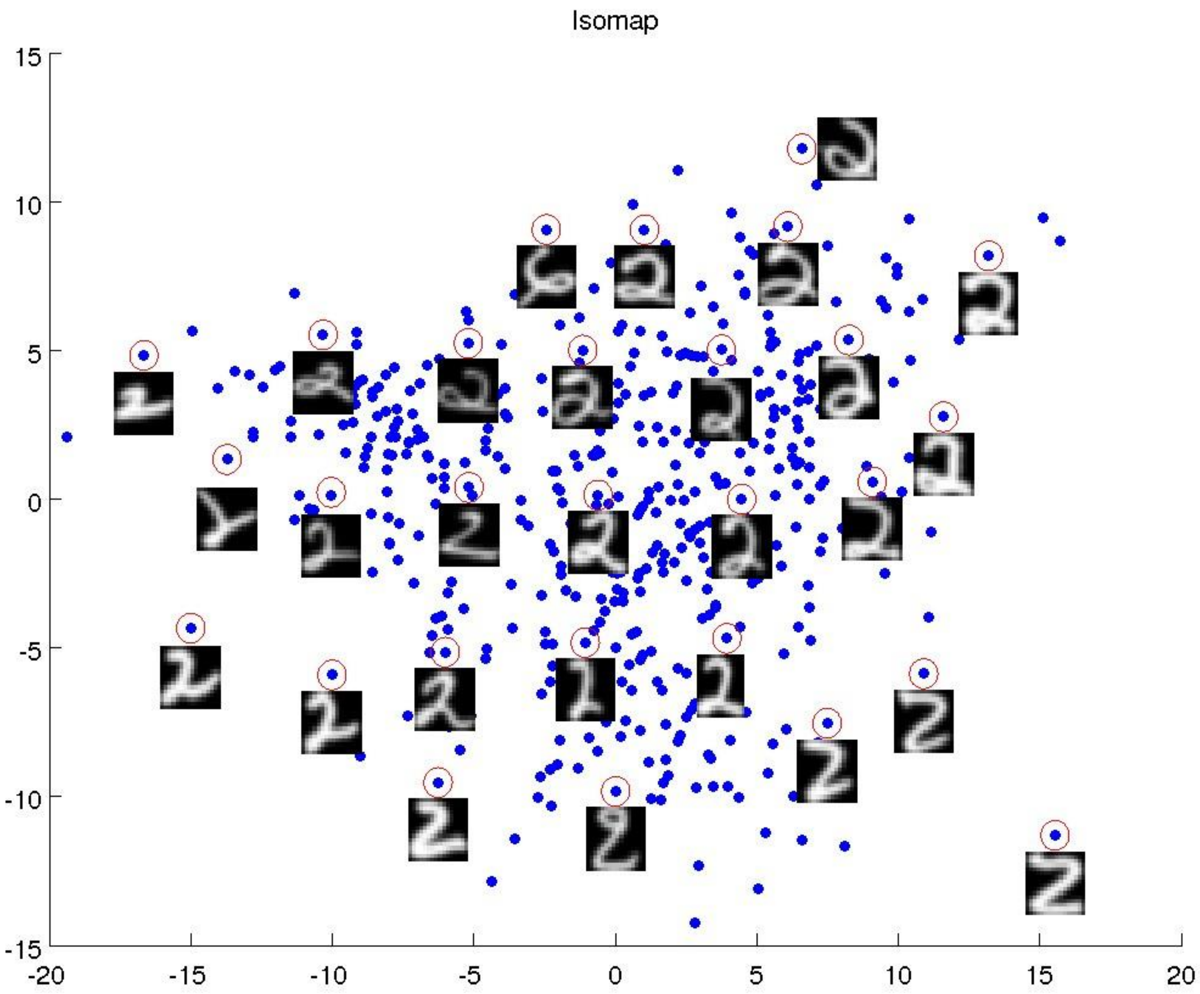
Swissroll (demo test_isomap2.py)



Faces



Handwritten Digits

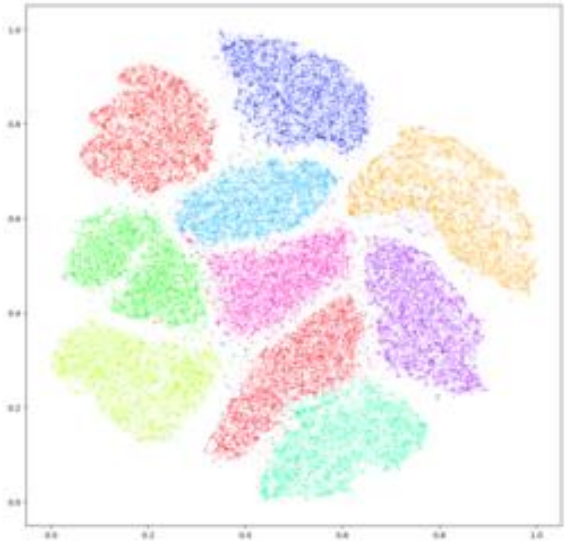


Takeaway

- **PCA reduces dimensions by finding the top-k principal directions**
  - Principal directions are equivalent to the directions with largest eigenvalues of the covariance matrix
  - It is also equivalent to directions that maximize the variance

- **When to use dimensionality reduction?**
  - Feature distribution analysis, feature engineering
  - Visualization
  - Note that dimensionality reduction finds directions maximizing variance, but not the predictive power

- **When to use linear/non-linear dimensionality reduction?**
  - Linear/non-linear similarity/distance metric
  - Non-linear dimensionality reduction: Isomap, SNE, t-SNE, kernel PCA



T-SNE embeddings of MNIST dataset