Principal-Component Analysis

COEN140

Santa Clara University

Rank of a matrix

- rank($\mathbf{A}$): the largest number of linearly independent columns (or rows) of matrix $\mathbf{A}$.
- If $\mathbf{A} \in \mathbb{R}^{m \times n}$, then

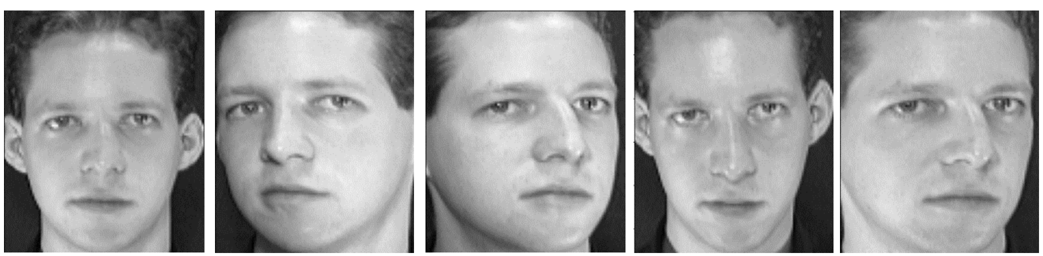
  $\text{rank}(\mathbf{A}) \leq \min(m, n)$;

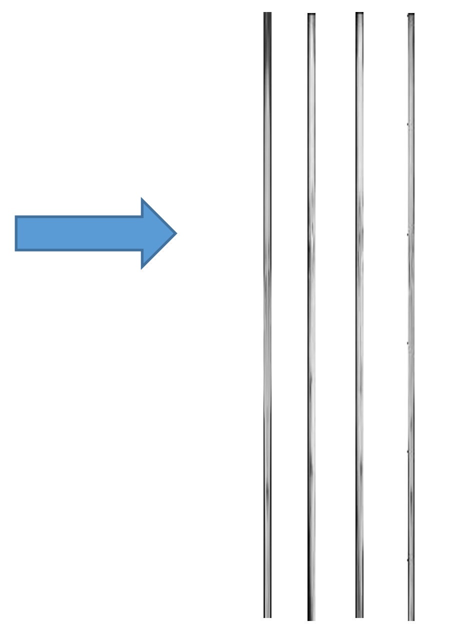
  $\mathbf{A}$ has full rank if $\text{rank}(\mathbf{A}) = \min(m, n)$
- Example: This matrix has rank of 3 because the 4th column can be written as a combination of the first 3 columns

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Rank of a matrix

- I have 5 different face images of the same person. Vectorize each image as a long vector. Form a matrix of size $D \times 5$. D is the number of pixels in each image, $D \gg 5$.
- What is the rank of matrix $\mathbf{X}$? Answer: 5

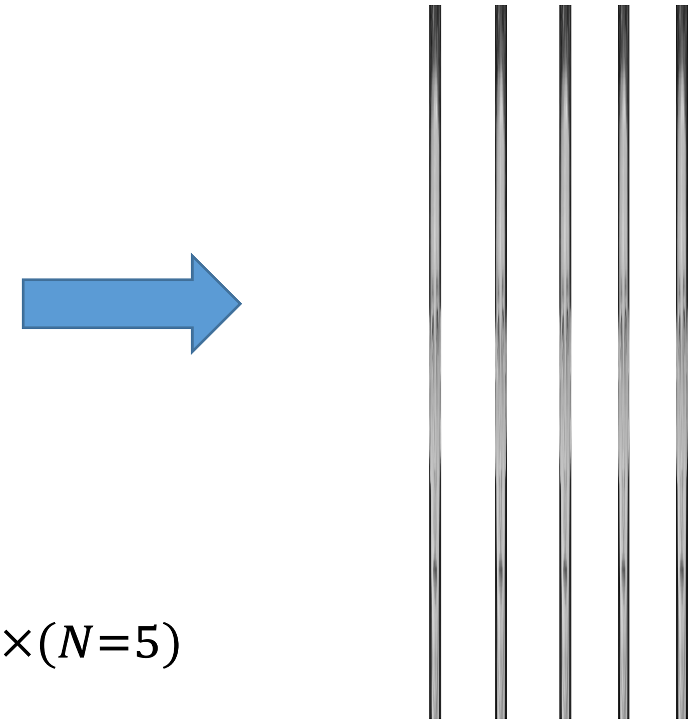




$$\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2\ \mathbf{x}_3\ \mathbf{x}_4\ \mathbf{x}_5] \in \mathbb{R}^{D\times(N=5)}$$

Rank of a matrix

- I take one face image of a person, repeat it 5 times to form a matrix
- What is the rank of the matrix? Answer: 1



$$\mathbf{X} = [\mathbf{x}\ \mathbf{x}\ \mathbf{x}\ \mathbf{x}\ \mathbf{x}] \in \mathbb{R}^{D\times(N=5)}$$

Eigenvalues and Eigenvectors

- For a square matrix $\mathbf{A}_{D \times D}$, which vectors get mapped to a scalar multiple of themselves?

- More precisely, which vectors $\mathbf{u}$ satisfy the following
$$\mathbf{A}\mathbf{u} = \lambda \mathbf{u}$$

  - These $\mathbf{u}$'s: eigenvectors of $\mathbf{A}$

  - $\mathbf{u} \in \mathbb{R}^D$

  - The scalar λ's: the associated eigenvalues

Eigenvalues and Eigenvectors

- For a full-rank matrix $\mathbf{A}_{D\times D}$, there are D eigenvectors
- $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_D$, and the corresponding eigenvalues are $\lambda_1, \lambda_2, \dots, \lambda_D$
- These eigenvectors are orthogonal to each other.

$$\mathbf{u}_i^T \mathbf{u}_j = 0, \forall i \neq j$$

- Each eigenvector has unit-norm: $\|\mathbf{u}_i\|_2 = 1, i = 1, \dots, D$.

- Usually we sort the eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D$

Eigenvalues and Eigenvectors

- Let $\mathbf{U} = [\mathbf{u}_1\ \mathbf{u}_2\ \ \dots \mathbf{u}_D\ ]$

- Property: $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}_D$: $D \times D$ identity matrix

- Hence, $\mathbf{U}^{-1} = \mathbf{U}^T$

Eigen-Value Decomposition (EVD)

- Let $\mathbf{U} = [\mathbf{u}_1 \ \mathbf{u}_2 \ \dots \mathbf{u}_D \ ]$
- Let $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_D) = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_D \end{bmatrix}$
- EVD: $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$

- Recall
  - $\mathbf{A}_{D \times D}$ is a square matrix
  - $\mathbf{A}\mathbf{u} = \lambda\mathbf{u}$

  - $\Rightarrow \mathbf{A}\mathbf{U} = \mathbf{U}\boldsymbol{\Lambda}$
  - $\Rightarrow \mathbf{A}\mathbf{U}\mathbf{U}^T = \mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$

Singular-Value Decomposition (SVD)

- Consider a full-rank matrix

$$\mathbf{X}_{D\times N} = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in \mathbb{R}^{D\times N}$$

- SVD: $\mathbf{X}_{D\times N} = \mathbf{U}_{D\times D}\mathbf{\Sigma}_{D\times N}\mathbf{V}_{N\times N}^{\mathrm{T}}$

- $\mathbf{U}_{D\times D} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_D]$: left singular vectors
  - $\mathbf{u}_i \in \mathbb{R}^{D\times 1}, i = 1, \dots, D$: same length as one column of $\mathbf{X}$
  - Unitary matrix: $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$

Singular-Value Decomposition (SVD)

- Consider a full-rank matrix

$$\mathbf{X}_{D\times N} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D\times N}$$

- SVD: $\mathbf{X}_{D\times N} = \mathbf{U}_{D\times D}\mathbf{\Sigma}_{D\times N}\mathbf{V}_{N\times N}^{\mathrm{T}}$

- $\mathbf{V}_{N\times N} = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_N]$: right singular vectors
  - $\mathbf{v}_i \in \mathbb{R}^{N\times 1}, i = 1, \ldots, N$: same length as one row of $\mathbf{X}$
  - Unitary matrix: $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}$

Singular-Value Decomposition (SVD)

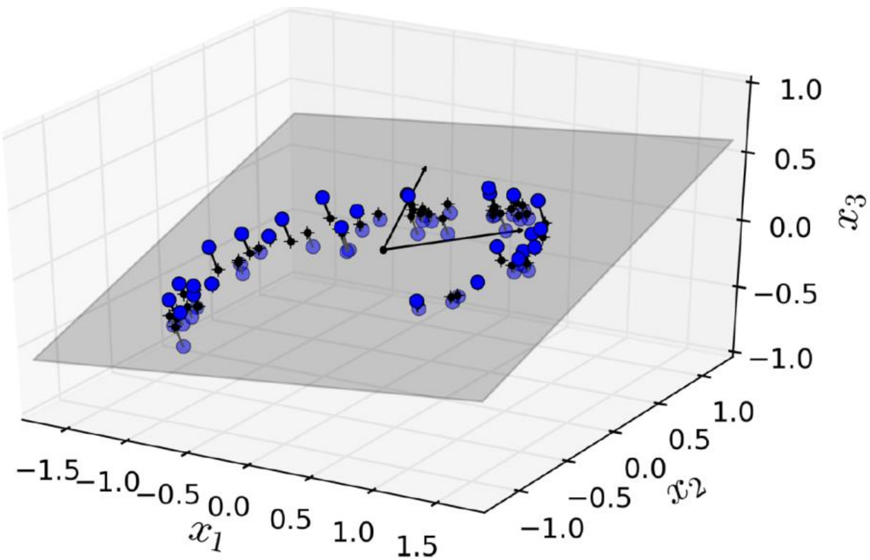
- Consider a full-rank matrix
$$\mathbf{X}_{D\times N} = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in \mathbb{R}^{D\times N}$$
- SVD: $\mathbf{X} = \mathbf{U}_{D\times D}\boldsymbol{\Sigma}_{D\times N}\mathbf{V}_{N\times N}^{\mathrm{T}}$
- If $D > N$, then
$$\boldsymbol{\Sigma}_{D\times N} = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ & & & \sigma_N \\ & & & \vdots \\ 0 & \cdots & & 0 \end{bmatrix}$$
- The singular values (assume sorted):
$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_N$$

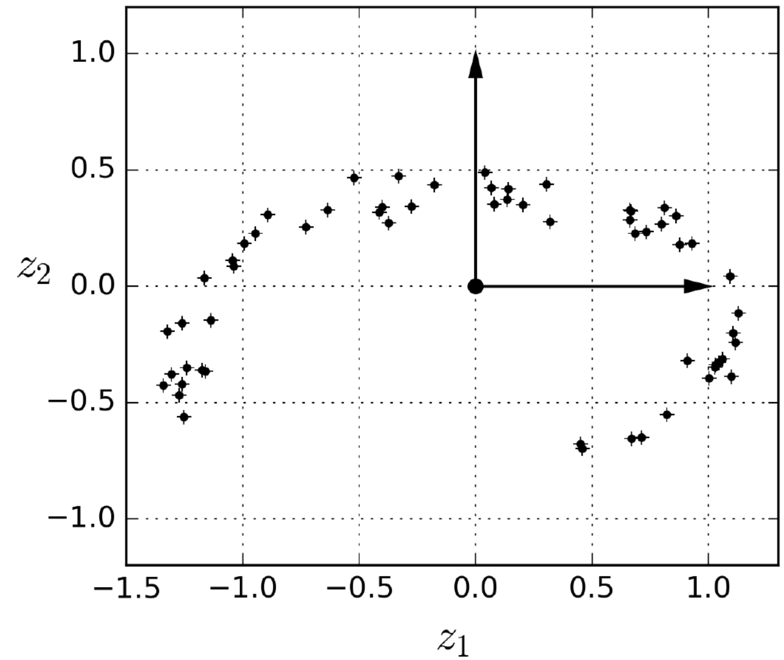
Dimensionality Reduction

- High-dimensional data may be approximately lying in a low-dimensional subspace.
- Most information in the data would be retained if we project the data onto this subspace
- Advantages: visualization, extracting meaningful attributes, computational efficiency

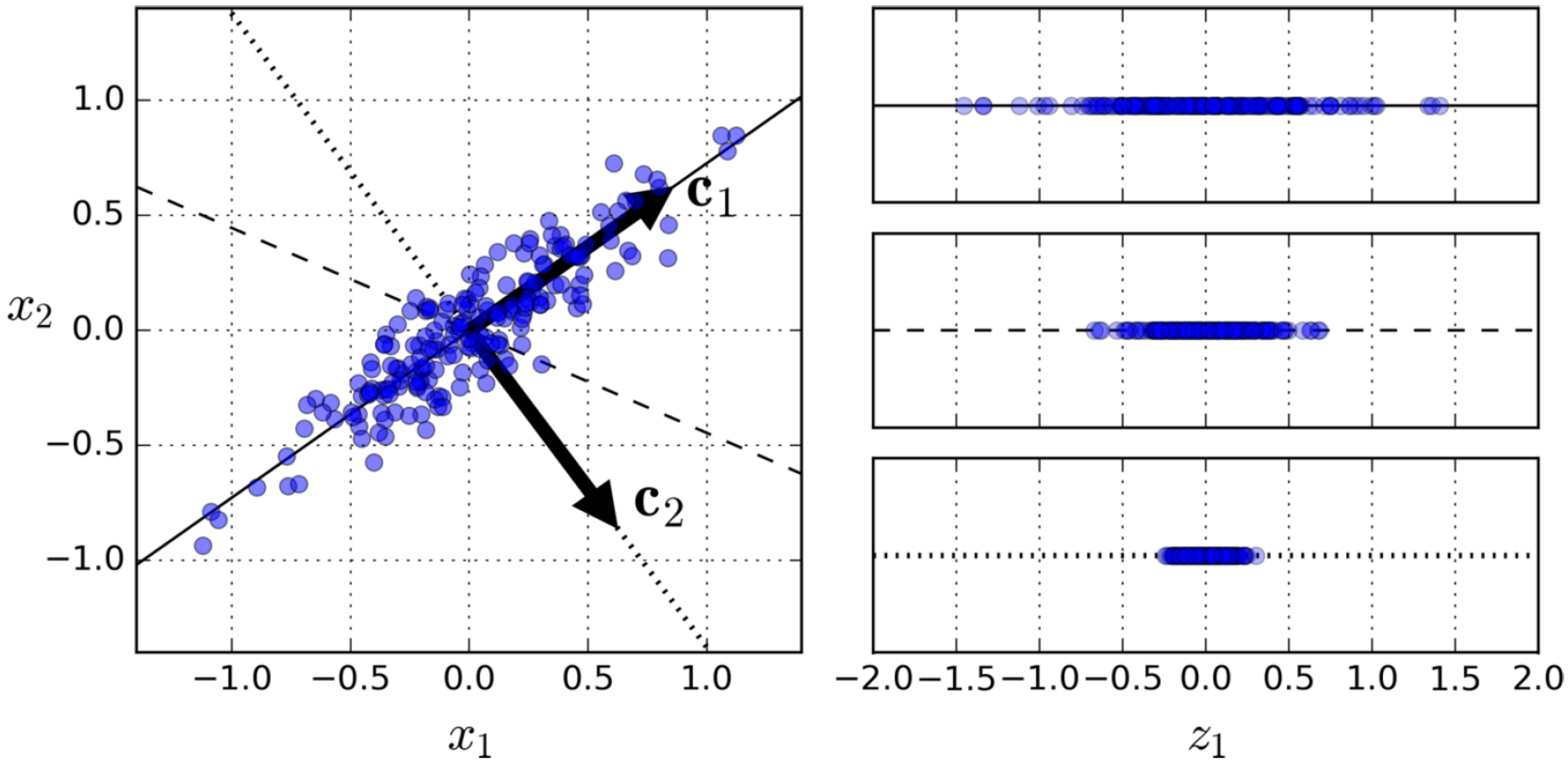
- Principal-component analysis (PCA):
    - A dimensionality reduction method

Dimensionality Reduction



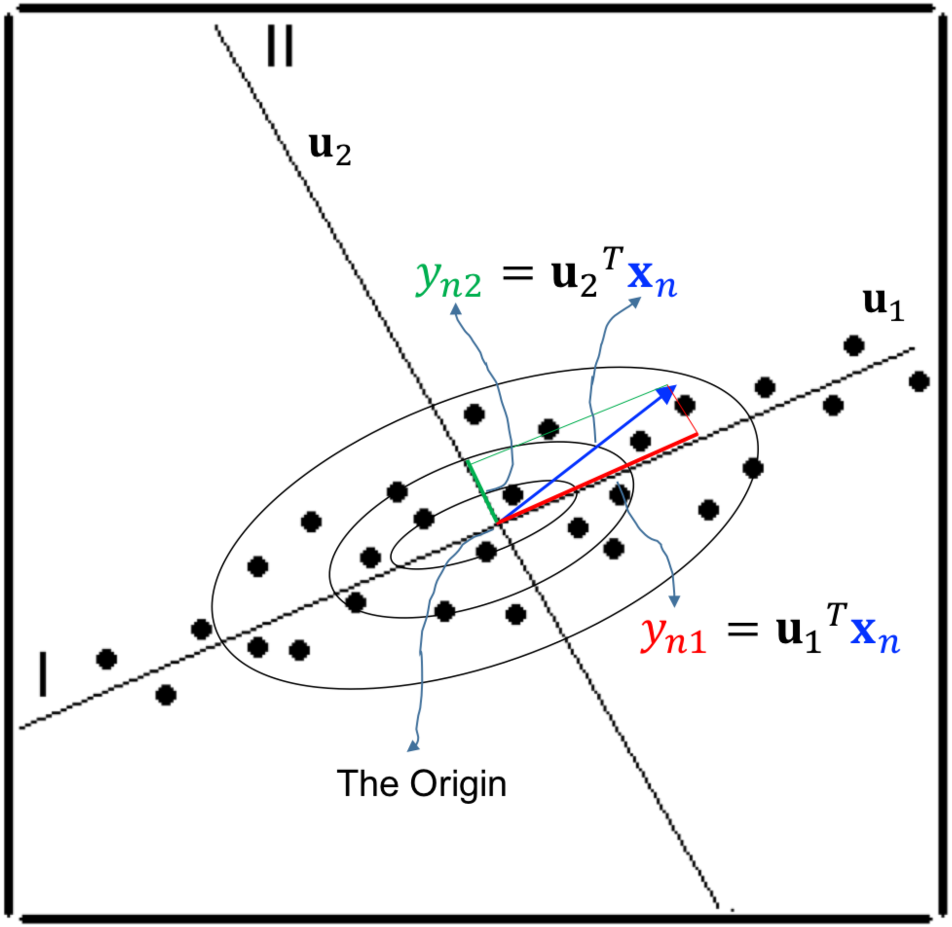


How to select the projection direction?



PCA

- We search for the direction in which the data projection has the largest energy.



N data points:
$\mathbf{x}_n, n = 1, 2, \ldots, N$
Assume $\mathbf{x}_n \in \mathbb{R}^2$

Look for the direction
$\mathbf{u} \in \mathbb{R}^2$ and $\|\mathbf{u}\|_2 = 1$ such that

$$\mathbf{u} = \arg \max_{\substack{\mathbf{u}^T\mathbf{u}=1 \\ \mathbf{u}\in\mathbb{R}^2}} \sum_{n=1}^{N} |\mathbf{x}_n{}^T\mathbf{u}|^2$$

PCA

- In general
  - if the data samples are $\mathbf{x}_n \in \mathbb{R}^D, n = 1, 2, \ldots, N$.
- Then we want to find $\mathbf{u} \in \mathbb{R}^D$ such that

$$\mathbf{u} = \arg \max_{\substack{\mathbf{u}^T\mathbf{u}=1 \\ \mathbf{u}\in\mathbb{R}^D}} \sum_{n=1}^{N} |\mathbf{x}_n{}^T \mathbf{u}|^2$$

- That is, we find the direction that maximizes the sum of the squared data projection.

PCA

- The solution:
- Form a data matrix

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$$

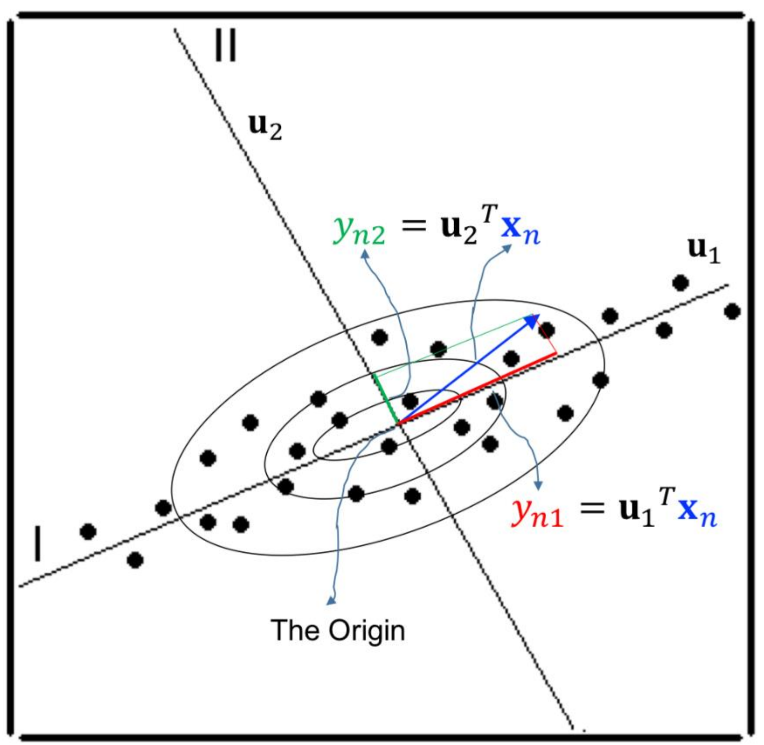
- Obtain the matrix $\mathbf{X}\mathbf{X}^T \in \mathbb{R}^{D \times D}$
- EVD: $\mathbf{X}\mathbf{X}^T = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$
  - $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_D] \in \mathbb{R}^{D \times D}$
- The solution $\mathbf{u}$ is
- $\mathbf{u} = \mathbf{u}_1$: the eigenvector of matrix $\mathbf{X}\mathbf{X}^T$ that corresponds to the largest eigenvalue λ_1
- We call $\mathbf{u}_1$: the top eigenvector

PCA

- $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_D]$

- y_i: the projection (or the coefficient) of the data sample $\mathbf{x}$ along direction $\mathbf{u}_i$
- Data projection

$$y_i = \mathbf{u}_i{}^T\mathbf{x}, i = 1, 2, \ldots, D$$



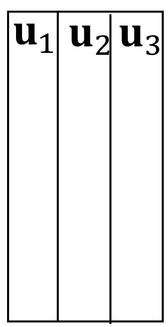
PCA

- Use d top eigenvectors of $\mathbf{X}\mathbf{X}^T$ for projection (those that correspond to the d largest eigenvalues)

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_d \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T\mathbf{x} \\ \vdots \\ \mathbf{u}_d^T\mathbf{x} \end{bmatrix}$$

- For example, $d = 3$

$\mathbf{U}_{:,[1:3]} =$



$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T\mathbf{x} \\ \mathbf{u}_2^T\mathbf{x} \\ \mathbf{u}_3^T\mathbf{x} \end{bmatrix}$$

- The D-dimensional data sample $\mathbf{x}$ is reduced to 3-dimensional data point $\mathbf{y}$

Singular-Value Decomposition (SVD)

- The principal components $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_D]$ can also be found by singular-value decomposition (SVD).
- Let the data matrix be $\mathbf{X}_{D\times N} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$

- SVD: $\mathbf{X}_{D\times N} = \mathbf{U}_{D\times D}\mathbf{\Sigma}_{D\times N}\mathbf{V}^{\mathrm{T}}_{N\times N}$
- $\mathbf{X}\mathbf{X}^T = \mathbf{U}_{D\times D}\mathbf{\Sigma}_{D\times N}\mathbf{V}^{\mathrm{T}}_{N\times N}\ \mathbf{V}_{N\times N}\mathbf{\Sigma}^{\mathrm{T}}_{N\times D}\ \mathbf{U}^{\mathrm{T}}_{D\times D}$

  $= \mathbf{U}_{D\times D}\mathbf{\Sigma}_{D\times N}\mathbf{\Sigma}^{\mathrm{T}}_{N\times D}\ \mathbf{U}^{\mathrm{T}}_{D\times D}$
- EVD: $\mathbf{X}\mathbf{X}^T = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$

  – $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_D]$: the left singular vectors of $\mathbf{X}_{D\times N}$ are the same as the eigenvectors of $\mathbf{X}\mathbf{X}^T$

PCA

- Use d top left singular vectors of $\mathbf{X}_{D\times N}$ for projection (those that correspond to the d largest singular values)

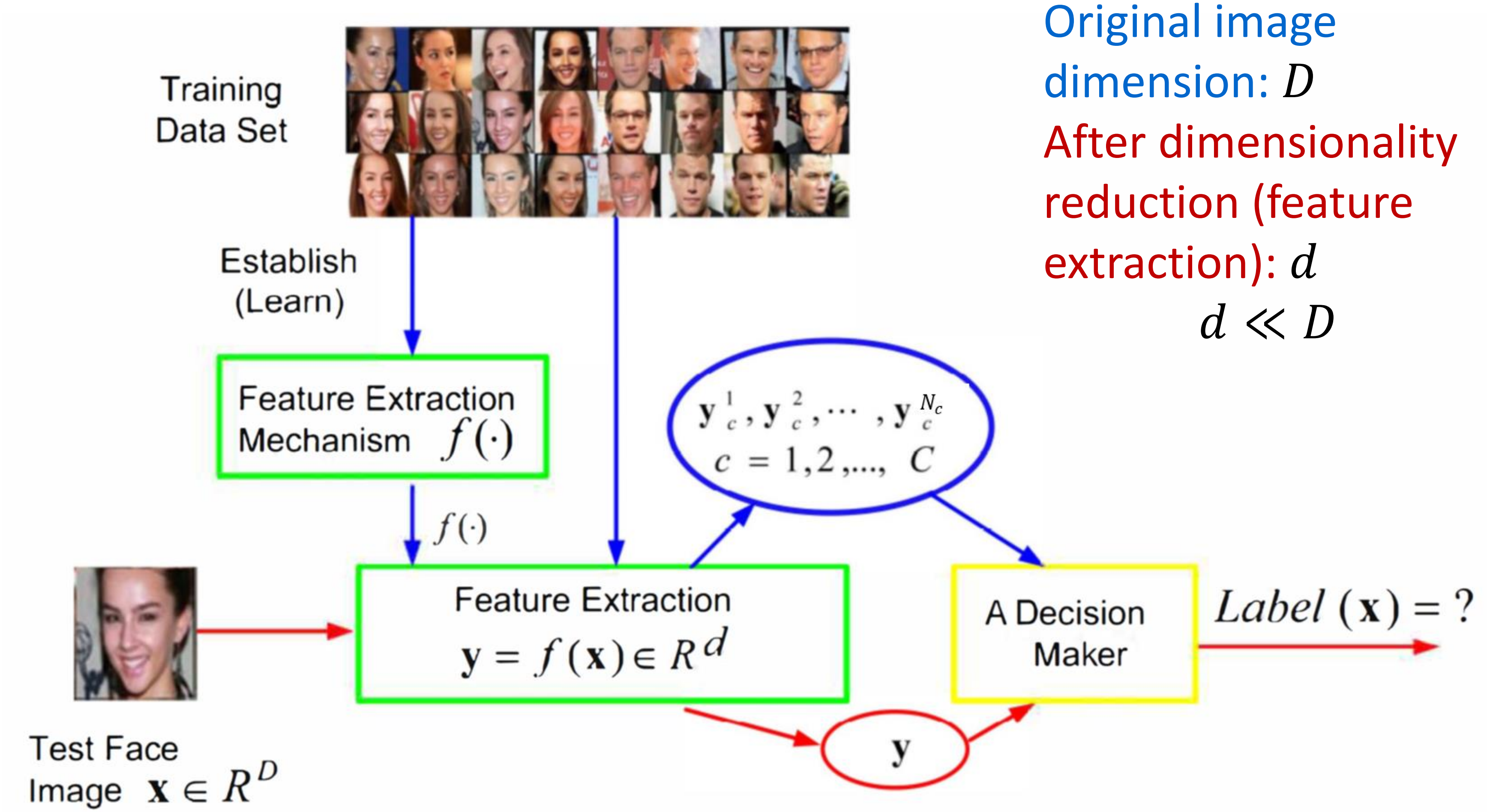
$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_d \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T \mathbf{x} \\ \vdots \\ \mathbf{u}_d^T \mathbf{x} \end{bmatrix}$$

- For example, $d = 3$

$$\mathbf{U}_{:,[1:3]} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T \mathbf{x} \\ \mathbf{u}_2^T \mathbf{x} \\ \mathbf{u}_3^T \mathbf{x} \end{bmatrix}$$

- The D-dimensional data sample $\mathbf{x}$ is reduced to 3-dimensional data point $\mathbf{y}$

Face Recognition



Assume:

Original image dimension: D

After dimensionality reduction (feature extraction): d

$$d \ll D$$

Face Recognition

- If we apply SVD for face feature extraction,
- $f(\mathbf{x}) = ?$
- $f(\mathbf{x}) = \mathbf{y}_{d\times 1} = {\mathbf{U}_{:,[1:d]}}^T \mathbf{x}_{D\times 1}, d \ll D$
- That is: $\mathbf{y}_{d\times 1} = \begin{bmatrix} y_1 \\ \vdots \\ y_d \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T\mathbf{x} \\ \vdots \\ \mathbf{u}_d^T\mathbf{x} \end{bmatrix}$

Face Recognition

- C subjects/Classes
- Training set (each subject has N training images)
  - Subject 1: $\mathbf{x}_{11}, \ldots, \mathbf{x}_{1N}$ (i.e. $\mathbf{x}_{1n}, n = 1, \cdots, N$)
  - Subject 2: $\mathbf{x}_{21}, \ldots, \mathbf{x}_{2N}$
  - …
  - Subject C: $\mathbf{x}_{C1}, \ldots, \mathbf{x}_{CN}$

Face Recognition

- SVD for dimensionality reduction
  - $\mathbf{X}$=$[\mathbf{x}_{11}, \dots, \mathbf{x}_{1N}, \mathbf{x}_{21}, \dots, \mathbf{x}_{2N}, \dots, \mathbf{x}_{C1}, \dots, \mathbf{x}_{CN}]$
    $\mathbf{X} = \mathbf{U\Sigma V}^{\mathrm{T}}$
  - Find the top-d left singular vectors
    $$\mathbf{U}_{:,[1:d]} = [\mathbf{u}_1, \cdots, \mathbf{u}_d]$$
  - Project each training image onto these vectors
    - The nth training image of the cth subject: $\mathbf{x}_{cn}$

$$\mathbf{y}_{cn} = \begin{bmatrix} y_{cn,1} \\ \vdots \\ y_{cn,d} \end{bmatrix} = \mathbf{U}_{:,[1:d]}{}^{T}\mathbf{x}_{cn} = \begin{bmatrix} \mathbf{u}_1^T\mathbf{x}_{cn} \\ \vdots \\ \mathbf{u}_d^T\mathbf{x}_{cn} \end{bmatrix}$$

Face Recognition

- Nearest-Neighbor Classifier
  - For a test image $\mathbf{x}$
  - Project the test image onto vectors

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_d \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T \mathbf{x} \\ \vdots \\ \mathbf{u}_d^T \mathbf{x} \end{bmatrix} = \mathbf{U}_{:,[1:d]}{}^T \ \mathbf{x}$$

  - Determine its class label by

$$\hat{c} = \arg \min_{\substack{c=1,2,\ldots,C \\ n=1,2,\ldots,N}} \|\mathbf{y} - \mathbf{y}_{cn}\|_2$$