So Far…

- It's time for
  - Unsupervised learning
    - We are only given inputs
    - Goal: find "interesting patterns"
    - Discovering clusters
      - Clustering
    - Discovering latent factors
      - Dimensionality reduction
      - Topic modeling
      - Matrix factorization

Dimensionality Reduction

Deng Cai (蔡登)

College of Computer Science
Zhejiang University

dengcai@gmail.com

Feature Representation

- For all the learning tasks (supervised, unsupervised), we need $\boldsymbol{x}$
- Better representation makes learning easier
- Minimum requirement:
  - $\boldsymbol{x}$ should contains relevant features
- But we don't know which features are useful.
  - As many features as possible
- Feature engineering problem:
  - Dimensionality reduction

What is Dimensionality Reduction?

- The Key:
  - Feature mapping from $\boldsymbol{x}$ to $\boldsymbol{z}$
  - The $\boldsymbol{x}$ is the original representation, usually with high dimensionality.
  - We believe the number of latent factors (degree of the freedoms) of the data is far less.
    - Handwritten digits example
  - Thus, the dimensionality of $\boldsymbol{z}$ is **usually** smaller than that of $\boldsymbol{x}$
  - This is the name DR comes from.

Linear and Nonlinear

$$\mathcal{F}(\boldsymbol{x} \in R^p) = \boldsymbol{z} \in R^d$$

$$f_1(\boldsymbol{x}) = z_1 \qquad f_i(\boldsymbol{x}) = z_i \qquad f_d(\boldsymbol{x}) = z_d$$

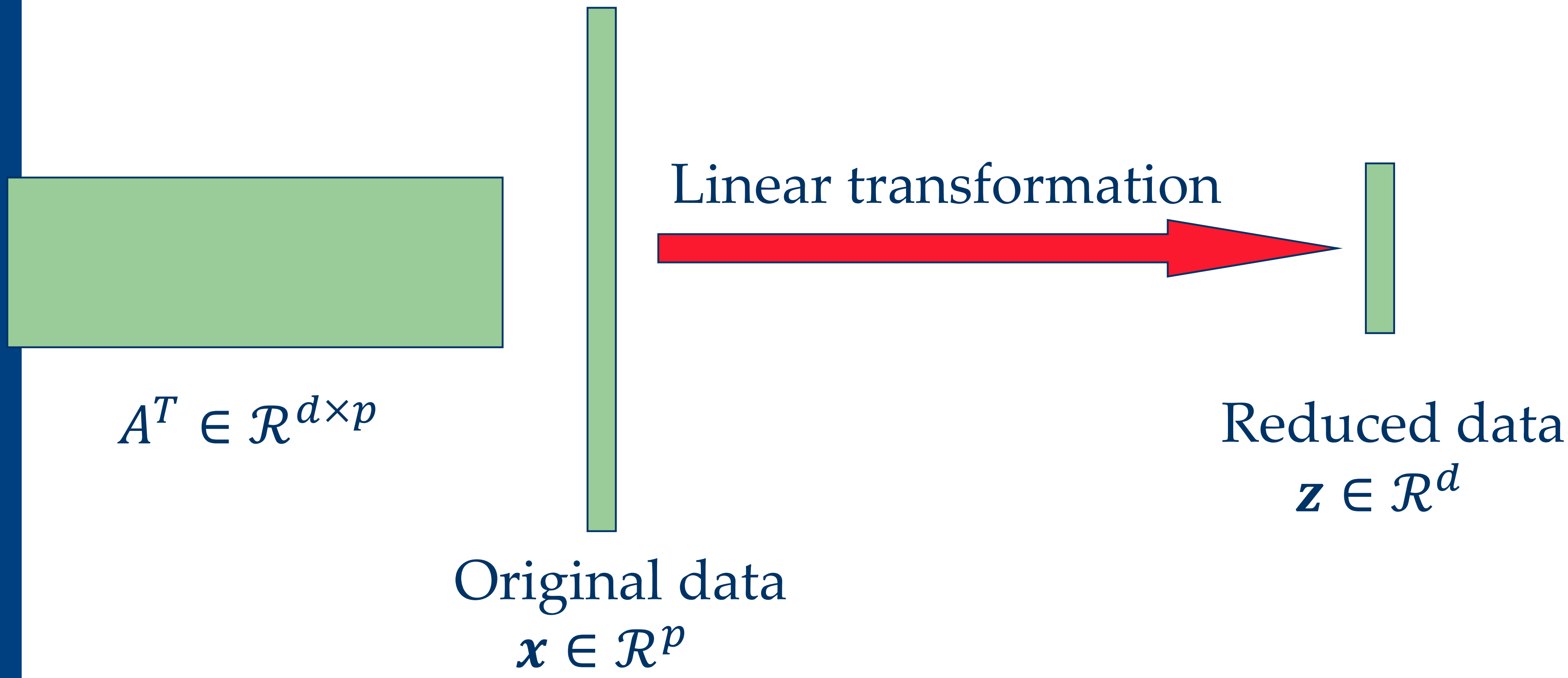
- All the methods (classification & clustering) can be seen as a DR approach (either supervised or unsupervised)
- If f is linear, linear dimensionality reduction

$$\boldsymbol{a}_1^T \boldsymbol{x} = z_1 \qquad \boldsymbol{a}_i^T \boldsymbol{x} = z_i$$

$$A \triangleq [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_d] \qquad A^T \boldsymbol{x} = \boldsymbol{z}$$

- If f is nonlinear, nonlinear dimensionality reduction
  - We know the embedding function
  - We don't know the function

Linear Transformation



$$A \in \mathcal{R}^{p \times d}: \boldsymbol{x} \in \mathcal{R}^p \rightarrow \boldsymbol{z} = A^T \boldsymbol{x} \in \mathcal{R}^d$$

Feature Extraction vs Feature Selection

- Dimensionality reduction (Feature reduction)
  - Feature extraction
  - Feature selection

- Selection: choose a best subset of size d from the available p features
- Extraction: given p features (set X), extract d new features (set Z) by linear or non-linear combination of all the p features

$$A \in \mathcal{R}^{p \times d}: \boldsymbol{x} \in \mathcal{R}^{p} \rightarrow \boldsymbol{z} = A^{T}\boldsymbol{x} \in \mathcal{R}^{d}$$

- Selection: $A \in [0,1]^{p \times d}$, every column of A has only one 1.
- Extraction: $A \in \mathcal{R}^{p \times d}$

Dimensionality Reduction Algorithms

- Unsupervised
  - Latent Semantic Indexing (LSI): truncated SVD
  - Principal Component Analysis (PCA)
  - Independent Component Analysis (ICA)
  - Canonical Correlation Analysis (CCA)
- Supervised
  - Linear Discriminant Analysis (LDA)
- Semi-supervised
  - Semi-supervised Discriminant Analysis (SDA)

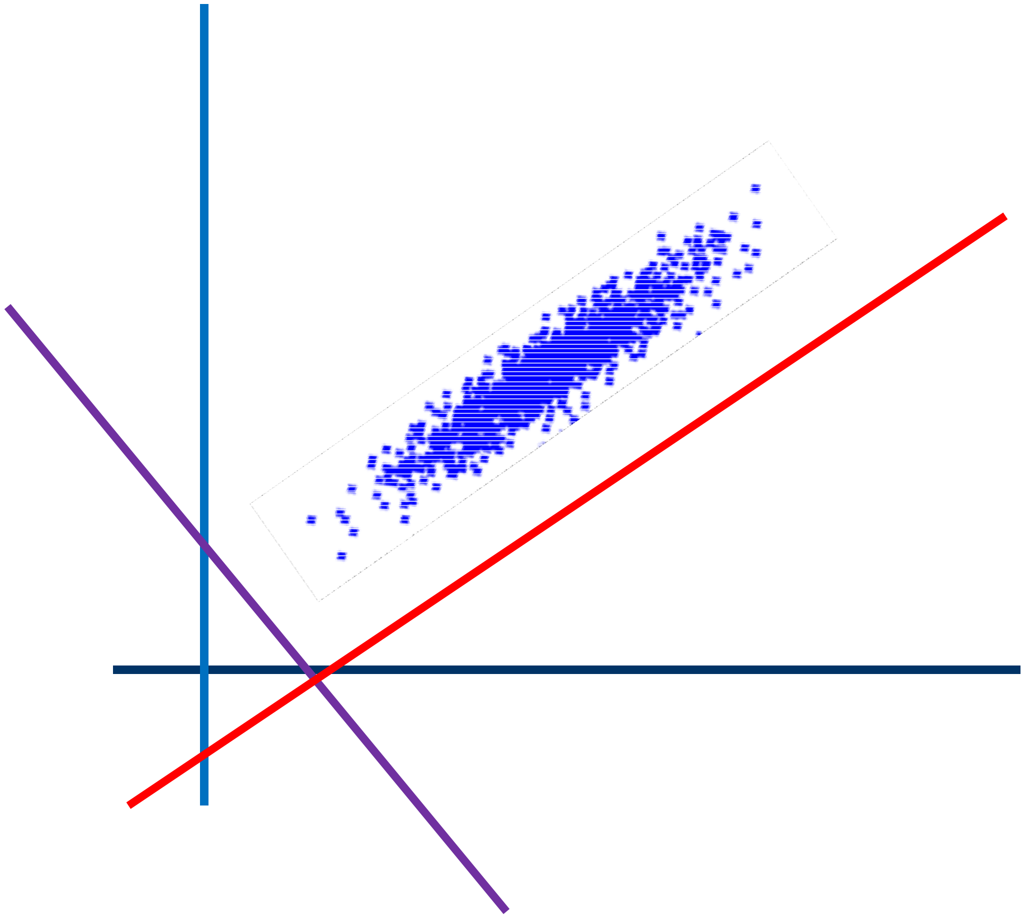
Dimensionality Reduction Algorithms

- Linear
  - Latent Semantic Indexing (LSI): truncated SVD
  - Principal Component Analysis (PCA)
  - Linear Discriminant Analysis (LDA)
  - Canonical Correlation Analysis (CCA)
- Nonlinear
  - Nonlinear feature reduction using kernels
  - Manifold learning

Algorithms

- Principal Component Analysis (PCA)
- Linear Discriminant Analysis (LDA)
- Locality Preserving Projections (LPP)
- The framework of graph based dimensionality reduction.
- Laplacian Eigenmap

Principal Component Analysis



What is Principal Component Analysis?

- Principal component analysis (PCA)
  - Reduce the dimensionality of a data set by finding a new set of variables, smaller than the original set of variables
  - Retains most of the sample's information.
  - Useful for the compression and classification of data.

- By information we mean the variation present in the sample, given by the correlations between the original variables.
  - The new variables, called principal components (PCs), are uncorrelated, and are ordered by the fraction of the total information each retains.

Algebraic Derivation of PCs

- Given a sample of n observations on a vector of p variables

$$\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\} \in \mathcal{R}^p$$

- Define the first principal component of the sample by the linear transformation

$$z_i^{(1)} = \boldsymbol{a}_1^T \boldsymbol{x}_i, \qquad i = 1, \cdots n$$

is chosen such that $var\left(z^{(1)}\right)$ is maximum.

Algebraic Derivation of PCs

$$var\left(z^{(1)}\right) = E\left(\left(z^{(1)} - \bar{z}^{(1)}\right)^2\right)$$

$$= \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{a}_1^T\boldsymbol{x}_i - \boldsymbol{a}_1^T\overline{\boldsymbol{x}}\right)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{a}_1^T(\boldsymbol{x}_i - \overline{\boldsymbol{x}})(\boldsymbol{x}_i - \overline{\boldsymbol{x}})^T\boldsymbol{a}_1 = \boldsymbol{a}_1^T S\boldsymbol{a}_1$$

Where $S = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{x}_i - \overline{\boldsymbol{x}})(\boldsymbol{x}_i - \overline{\boldsymbol{x}})^T$

is the covariance matrix and $\overline{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i$ is the mean.

Algebraic Derivation of PCs

$$\max_{\boldsymbol{a}_1} \boldsymbol{a}_1^T S \boldsymbol{a}_1$$

$$s.t. \quad \boldsymbol{a}_1^T \boldsymbol{a}_1 = 1$$

Let λ be a Lagrange multiplier

$$L = \boldsymbol{a}_1^T S \boldsymbol{a}_1 - \lambda\left(\boldsymbol{a}_1^T \boldsymbol{a}_1 - 1\right)$$

$$\frac{\partial L}{\partial \boldsymbol{a}_1} = 2S\boldsymbol{a}_1 - 2\lambda \boldsymbol{a}_1 = 0$$

$$S\boldsymbol{a}_1 = \lambda \boldsymbol{a}_1$$

therefor, $\boldsymbol{a}_1$is an eigenvector of S corresponding to the largest eigenvalue $\lambda = \lambda_1$.

Algebraic Derivation of PCs

$$\max_{\boldsymbol{a}_2} \boldsymbol{a}_2^T S \boldsymbol{a}_2$$

$$s.t. \quad \boldsymbol{a}_2^T \boldsymbol{a}_2 = 1, \mathrm{cov}\left(z^{(2)}, z^{(1)}\right) = 0$$

$$\mathrm{cov}\left(z^{(2)}, z^{(1)}\right) = \boldsymbol{a}_2^T S \boldsymbol{a}_1 = \lambda \boldsymbol{a}_2^T \boldsymbol{a}_1 = 0$$

$$S\boldsymbol{a}_2 = \lambda \boldsymbol{a}_2$$

$\boldsymbol{a}_2$ is an eigenvector of S corresponding to the second largest eigenvalue $\lambda = \lambda_2$.

Algebraic Derivation of PCs

- In general:

$$var\left(z^{(k)}\right) = \boldsymbol{a}_k^T S \boldsymbol{a}_k = \lambda_k$$

- The k^{th} largest eigenvalue of S is the variance of k^{th} PC.
- The k^{th} PC $z^{(k)}$ retains the k^{th} greatest fraction of the variation in the sample.

Principle Component Analysis

- Main steps for computing PCs:
  - Form the covariance matrix S.
  - Compute its eigenvectors: $\{\boldsymbol{a}_i\}_{i=1}^{p}$
  - Use the first d eigenvectors $\{\boldsymbol{a}_i\}_{i=1}^{d}$ to form the d PCs.
  - The transformation A is given by

$$A = [\boldsymbol{a}_1, \cdots \boldsymbol{a}_d]$$

- A test point $\boldsymbol{x} \in \mathcal{R}^p \rightarrow A^T\boldsymbol{x} \in \mathcal{R}^d$