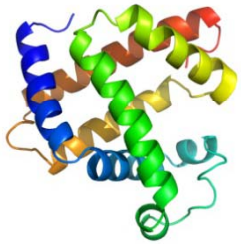
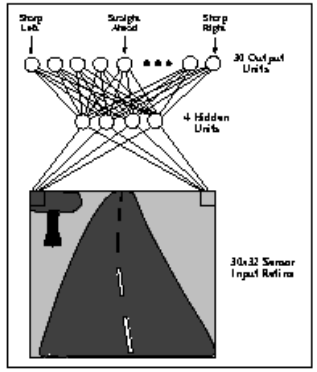
Unsupervised Learning
无监督学习

Supervised learning has many successes

- Document classification
- Protein prediction
- Face recognition
- Speech recognition
- Vehicle steering

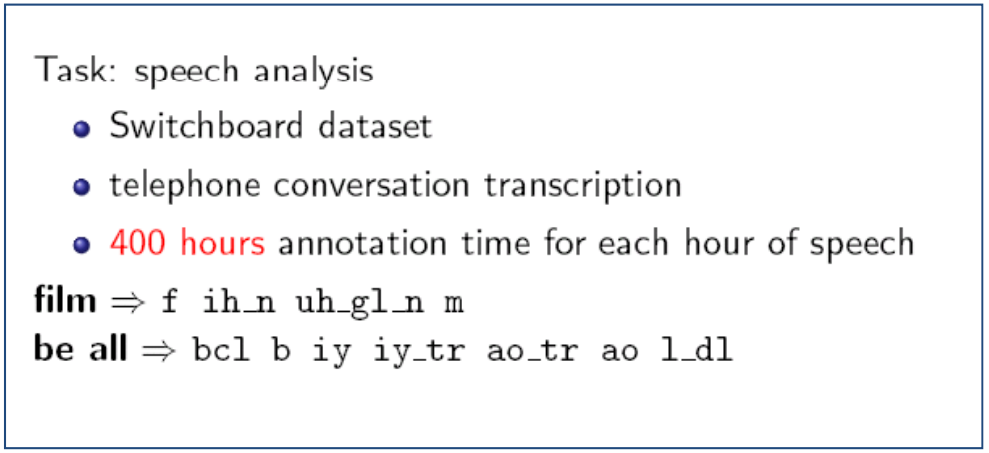
etc.





However…

- Labeled data can be rare or expensive in many real applications
  - Speech
  - Medical data
  - Protein
  - …



- Unlabeled data is much cheaper and abundant

Question: Can we use unlabeled data to help?

Can we use unlabeled data to help?

- Unlabeled data is missing important information…

- But maybe still has useful regularities that we can use.

Unsupervised learning

Unsupervised learning

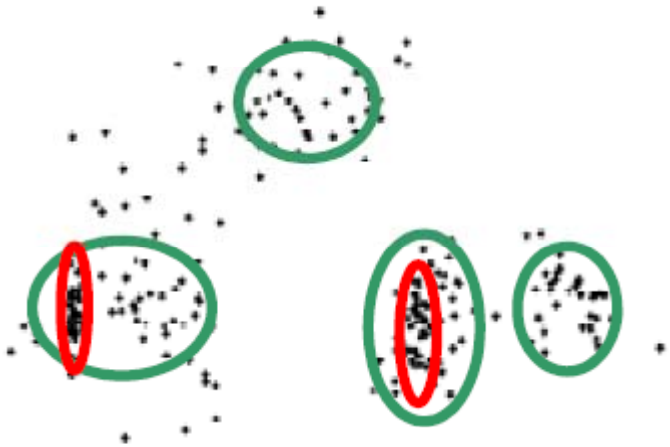
Learning from unlabeled data (without supervision)

Training data
$D = \{x_1, x_2 \ldots, x_N\}$

Prediction rule
$f(\mathbf{x}) \mapsto y$

What can we predict from unlabeled data?

– Groups or clusters in the data

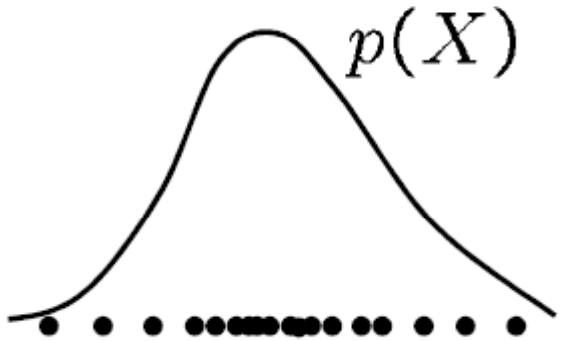


Unsupervised learning

Learning from unlabeled data (without supervision)

What can we predict from unlabeled data?

- Groups or clusters in the data
- Density estimation（密度估计）



Unsupervised learning

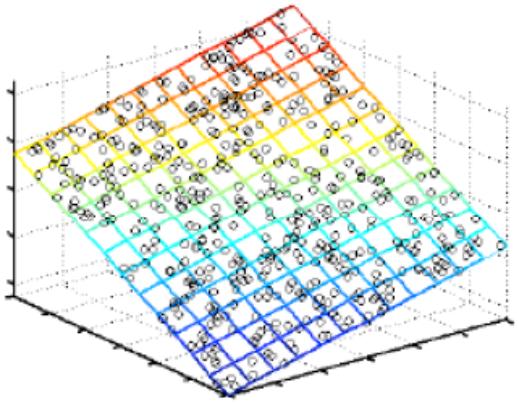
Learning from unlabeled data (without supervision)

Training data
$D = \{x_1, x_2 \ldots, x_N\}$

Prediction rule
$f(\mathbf{x}) \mapsto y$

What can we predict from unlabeled data?

- Groups or clusters in the data
- Density estimation（密度估计）
- Low-dimensional structure
  - Principal Component Analysis 主元分析(PCA) (linear)

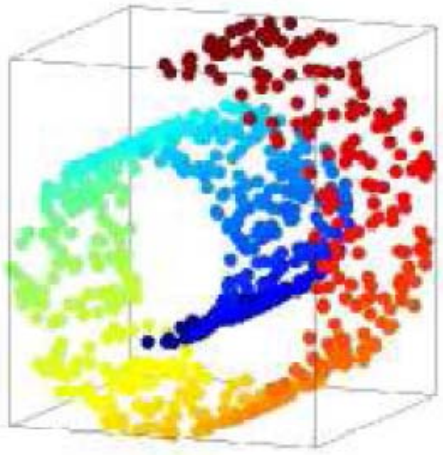


Unsupervised learning

Learning from unlabeled data (without supervision)

What can we predict from unlabeled data?

- Groups or clusters in the data
- Density estimation（密度估计）
- Low-dimensional structure
  - Principal Component Analysis 主元分析(PCA) (linear)
  - Manifold learning 流行学习 (non-linear)



Discover low dimensional structure

PRINCIPLE COMPONENT ANALYSIS

Principle component analysis

- What is dimensionality reduction?
- Why dimensionality reduction?
- Principal Component Analysis (PCA)
- Nonlinear PCA using Kernels

What is dimensionality reduction?

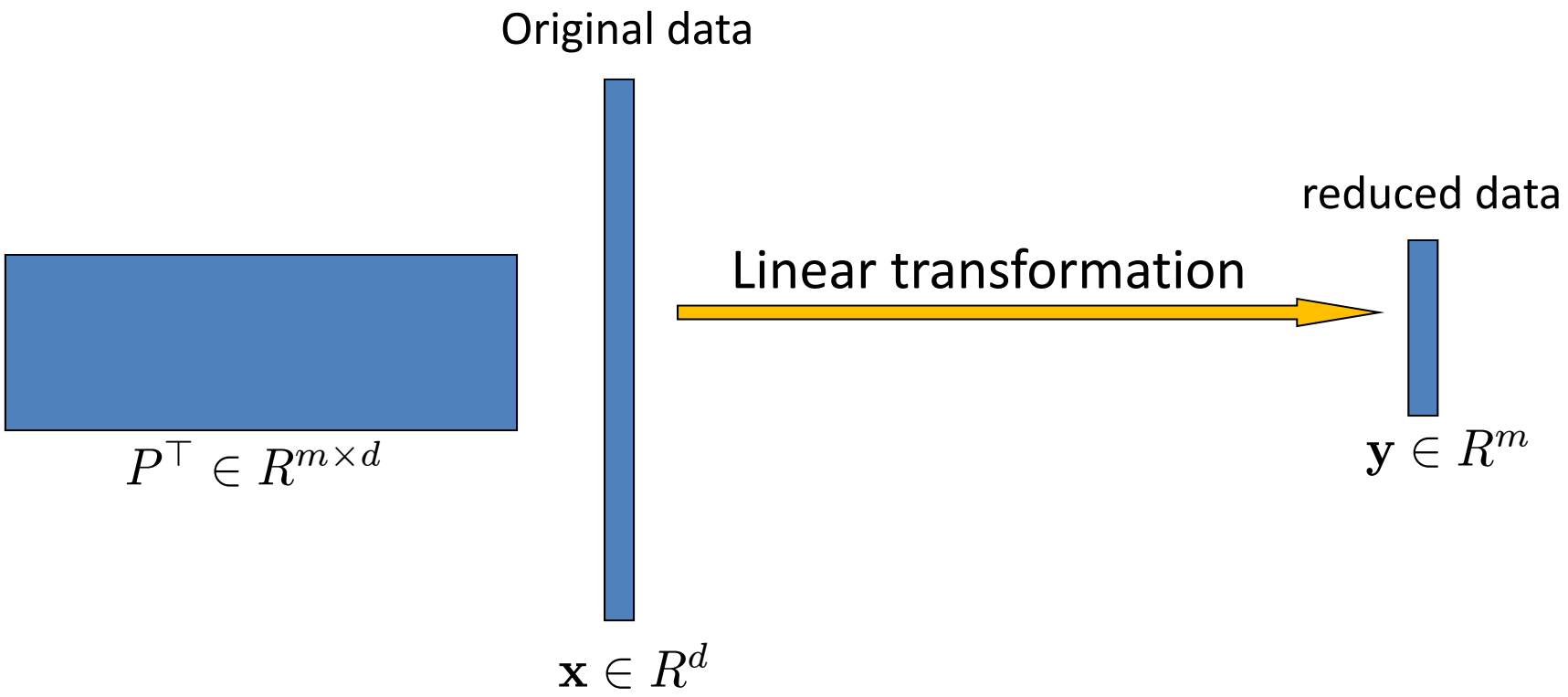
- Dimensionality reduction refers to the mapping of the original high-dimensional data onto a lower-dimensional space.
  - Criterion for dimensionality reduction can be different based on different problem settings.
    - Unsupervised setting: minimize the information loss
    - Supervised setting: maximize the class discrimination

- Given a set of data points of d variables $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$

  Compute the linear transformation (projection)

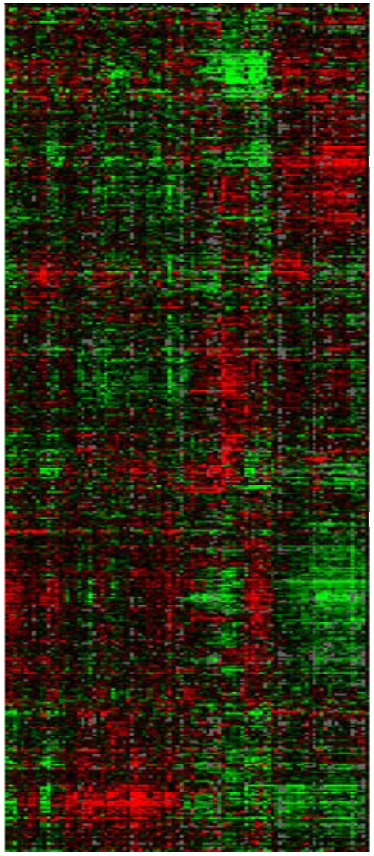
$$P \in R^{d \times m} : \mathbf{x} \in R^d \rightarrow \mathbf{y} = P^\top \mathbf{x} \in R^m (m << d)$$

What is dimensionality reduction?



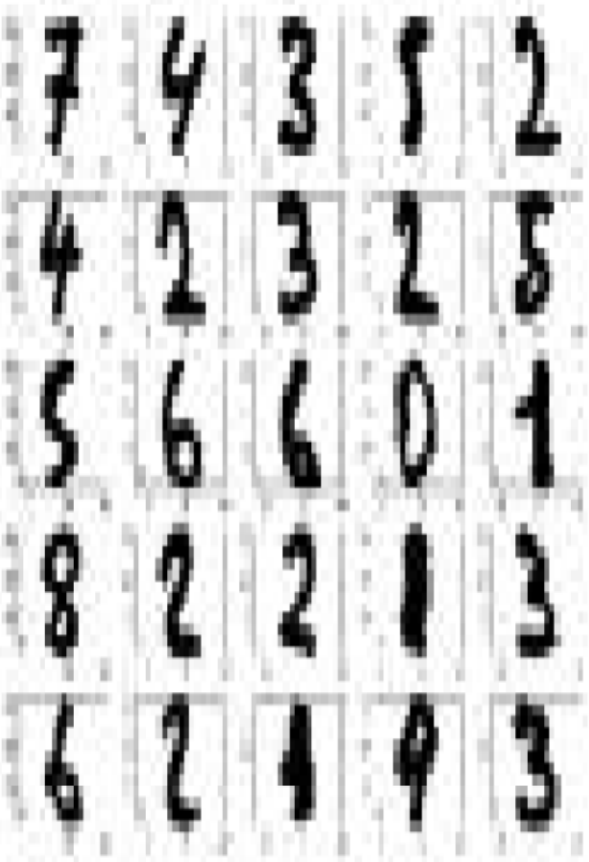
$$P \in R^{d \times m} : \mathbf{x} \to \mathbf{y} = P^\top \mathbf{x} \in R^m$$

High-dimensional data



Gene expression

Face images



Handwritten digits

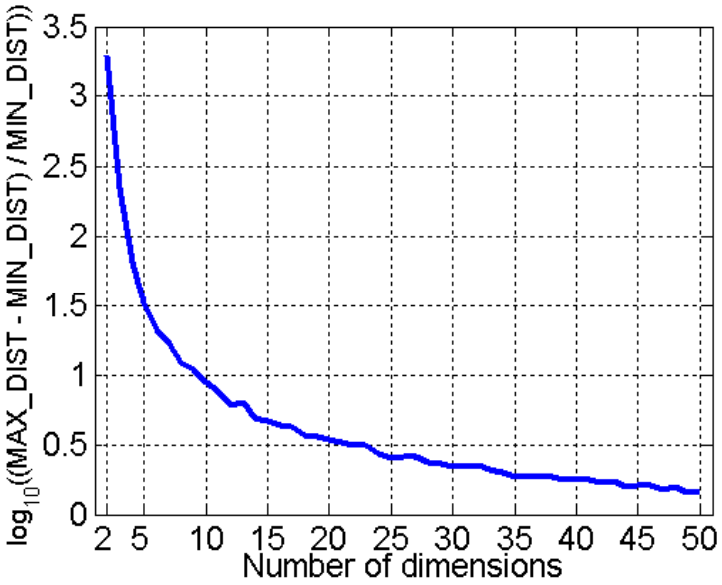
Why dimensionality reduction?

- Most machine learning and data mining techniques may not be effective for high-dimensional data
  - Curse of Dimensionality
  - Query accuracy and efficiency degrade rapidly as the dimension increases.

- The intrinsic dimension may be small.
  - For example, the number of genes responsible for a certain type of disease may be small.

Curse of Dimensionality

- When dimensionality increases, data becomes increasingly sparse in the space that it occupies
- Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful
- If $N_1 = 100$ represents a dense sample for a single input problem, then $N_{10} = 100^{10}$ is the sample size required for the same sampling density with dimension 10.
- The proportion of a hypersphere with radius *r* and dimension *d*, to that of a hyercube with sides of length *2r* and dimension *d* converges to 0 as *d* goes to infinity — nearly all of the high-dimensional space is "far away" from the center



- Randomly generate 500 points
- Compute difference between max and min distance between any pair of points

Why dimensionality reduction?

- Visualization: projection of high-dimensional data onto 2D or 3D.
- Data compression: efficient storage and retrieval.
- Noise removal: positive effect on query accuracy.

Application of feature reduction

- Face recognition
- Handwritten digit recognition
- Text mining
- Image retrieval
- Microarray data analysis
- Protein classification

……

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.

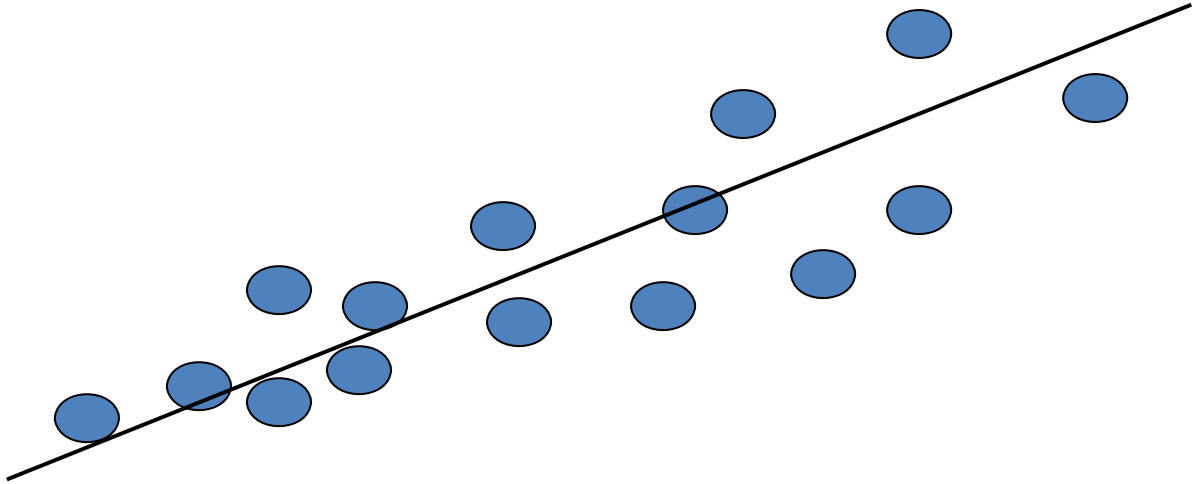
- By information we mean the variation present in the sample, given by the correlations between the original variables.
  - The new variables, called principal components (PCs), are uncorrelated, and are ordered by the fraction of the total information each retains.

Principal components (PCs)

- Given *n* points in a *d* dimensional space, for large *d*, how does one project on to a low dimensional space while preserving broad trends in the data and allowing it to be visualized?

Geometric picture of principal components

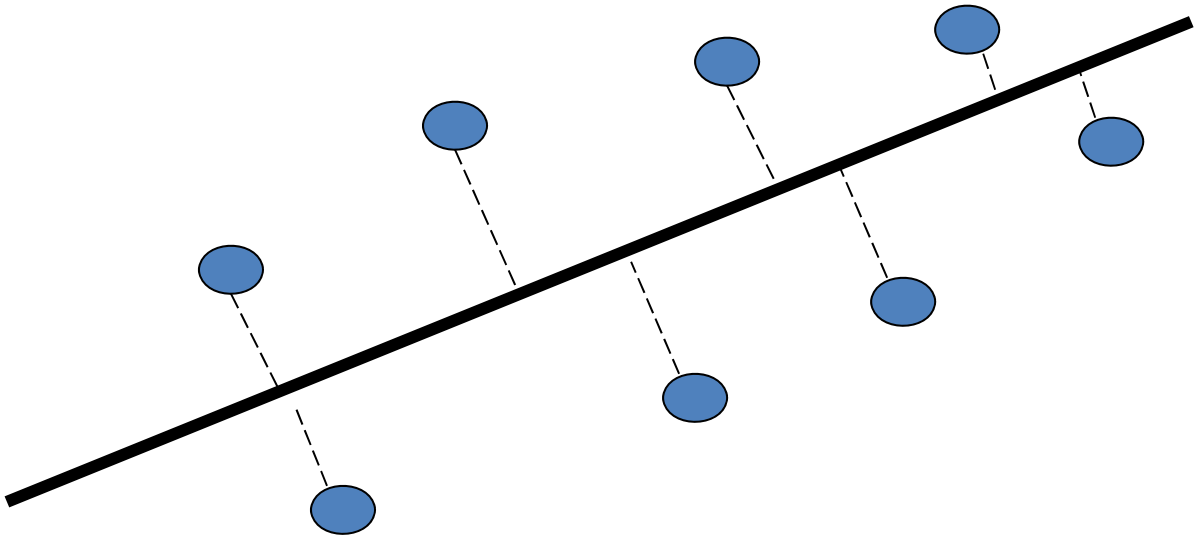
- Given *n* points in a d dimensional space, for large *d*, how does one project on to a 1 dimensional space?



- Choose a line that fits the data so the points are spread out well along the line

Geometric picture of principal components

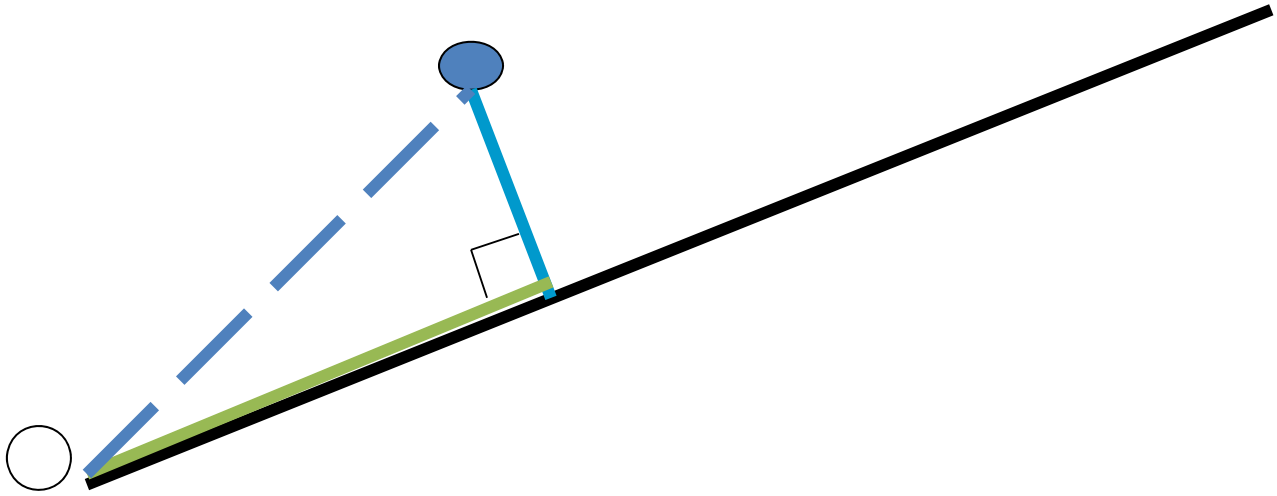
- Formally, minimize sum of squares of distances to the line.



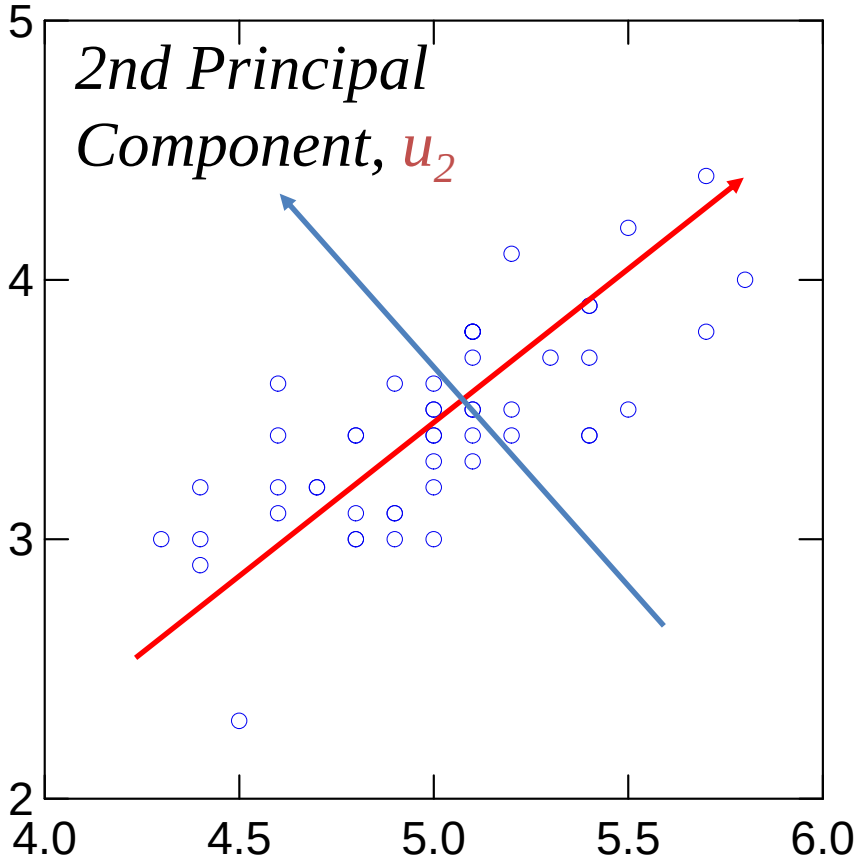
- Why sum of squares?

Algebraic Interpretation – 1D

- Minimizing sum of squares of distances to the line is the same as maximizing the sum of squares of the projections on that line, thanks to Pythagoras.



Geometric picture of principal components



Geometric picture of principal components

- the 1st PC $\mathbf{u}_1$ is a minimum distance fit to a line in X space
- the 2nd PC $\mathbf{u}_2$ is a minimum distance fit to a line in the plane perpendicular （垂直于） to the 1st PC

PCs are a series of linear least squares fits to a sample, each orthogonal （垂直于） to all the previous.

Algebraic derivation of PCs

- Given a sample of *n* observations on a vector of *d* variables $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\} \in R^d$
- First project the data onto a one-dimensional space with a *d*-dimensional vector $\mathbf{u}_1 : \mathbf{u}_1^\top \mathbf{u}_1 = 1$:

$$\{\mathbf{u}_1^\top \mathbf{x}_1, \mathbf{u}_1^\top \mathbf{x}_2, \cdots, \mathbf{u}_1^\top \mathbf{x}_n\}$$

- Find $\mathbf{u}_1$ to maximize the variance the projected data:

$$\frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}_1^\top \mathbf{x}_i - \mathbf{u}_1^\top \bar{\mathbf{x}})^2 = \mathbf{u}_1^\top S \mathbf{u}_1$$

Where $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$ and $S = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$

Algebraic derivation of PCs

- To solve $\max_{\mathbf{u}_1} \mathbf{u}_1^\top S \mathbf{u}_1 \quad \text{subject to} \quad \mathbf{u}_1^\top \mathbf{u}_1 = 1$
- Let λ be a Lagrangian multiplier

$$
\begin{aligned}
L &= \mathbf{u}_1^\top S \mathbf{u}_1 + \lambda_1 (1 - \mathbf{u}_1^\top \mathbf{u}_1) \\
\frac{\partial L}{\partial \mathbf{u}_1} &= S\mathbf{u}_1 - \lambda_1 \mathbf{u}_1 = 0 \\
S\mathbf{u}_1 &= \lambda_1 \mathbf{u}_1 \\
\Rightarrow \mathbf{u}_1 & \quad \text{is an eigenvector} \\
\mathbf{u}_1^\top S \mathbf{u}_1 &= \lambda_1 \\
\Rightarrow \mathbf{u}_1 & \quad \text{corresponds to the eigenvector with the largest eigenvalue } \lambda_1
\end{aligned}
$$

Algebraic derivation of PCs

- To find the second component $\mathbf{u}_2$
- Solve the following

$$\max_{\mathbf{u}_2} \ \mathbf{u}_2^\top S \mathbf{u}_2 \quad \text{subject to} \quad \mathbf{u}_2^\top \mathbf{u}_2 = 1 \ \ \& \ \ \mathbf{u}_1^\top \mathbf{u}_2 = 0$$

  - $\mathbf{u}_2$ is the eigenvector with the second largest eigenvalue λ_2

……

Algebraic derivation of PCs

- Main steps for computing PCs
  - Calculate the covariance matrix *S*

    $S = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$

    or first center the data: $\{\mathbf{x}'_1, \mathbf{x}'_2, \cdots, \mathbf{x}'_n\}$ and $\bar{\mathbf{x}}' = 0$

    let $X = [\mathbf{x}'_1, \mathbf{x}'_2, \cdots, \mathbf{x}'_n] \in R^{d\times n}$; then $S = \frac{1}{n}XX^\top$
  - Find the first *m* eigenvectors $\{\mathbf{u}_i\}_{i=1}^{m}$
  - Form the projection matrix $P = [\mathbf{u}_1\ \mathbf{u}_2\ \cdots \mathbf{u}_m] \in R^{d\times m}$
  - A new test point can be projected as:

    $\mathbf{x}_{new} \in R^d \rightarrow P^\top \mathbf{x}_{new} \in R^m$

Algebraic derivation of PCs

$$\mathbf{y} = P^\top \mathbf{x} \in R^m$$

- Getting the old data back?
  - If *P* is a square matrix, we can recover x by
    $$\mathbf{x} = (P^\top)^{-1}\mathbf{y} = P\mathbf{y} = PP^\top\mathbf{x}$$
  - Here P is not full, but we can still recover x by $\mathbf{x} = P\mathbf{y} = PP^\top\mathbf{x}$, and lose some information

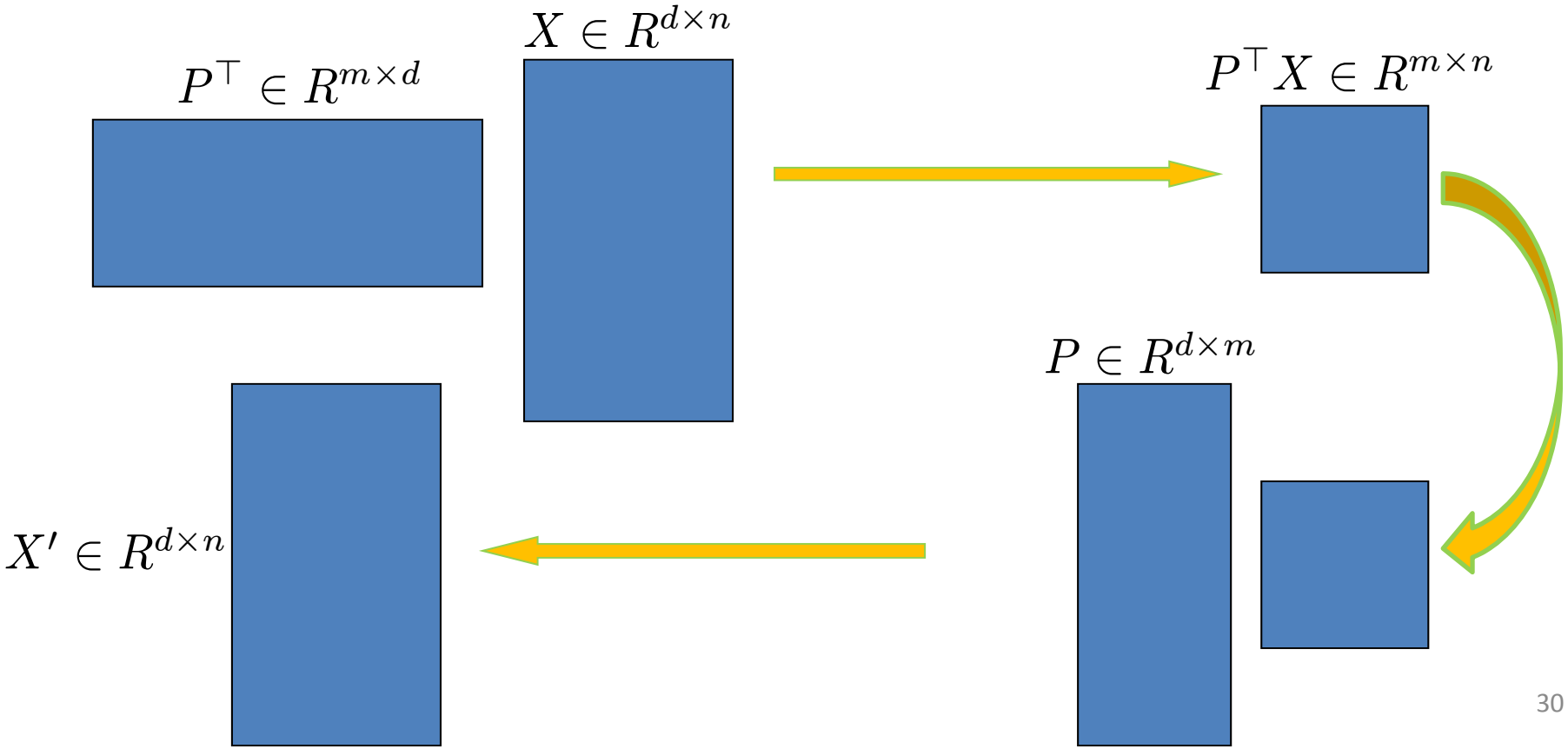
- Objective
  - Lose least amount of information

Optimality property of PCA

Reconstruction

Dimension reduction

$$X \in R^{d\times n} \rightarrow Y = P^\top X \in R^{m\times n} \rightarrow X' = PP^\top X \in R^{d\times n}$$



Optimality property of PCA

Main theoretical result:

The matrix P consisting of the first m eigenvectors of the covariance matrix S solves the following min problem:

$$\arg \min_{P \in R^{d \times m}} \|X - X'\|^2 = \arg \min_{P \in R^{d \times m}} \|X - PP^\top X\|^2$$

$$= \arg \max_{P \in R^{d \times m}} trace(X^\top PP^\top X)$$

$$= \arg \max_{P \in R^{d \times m}} trace(P^\top XX^\top P)$$

$$= \arg \max_{P \in R^{d \times m}} trace(P^\top SP)$$

$$\text{subject to} \quad P^\top P = I_m$$

Reconstruction error

PCA projection minimizes the reconstruction error among all linear projections of size m.

PCA for image compression

m=1

m=2

m=4

m=8

m=16

m=32

m=64

m=100

Original Image

Nonlinear PCA using Kernels

Rewrite PCA in terms of dot products

Assume the data has been centered, i.e. $\sum_i \mathbf{x}_i = 0$

The covariance matrix *S* can be written as $S = \frac{1}{n}\sum_i \mathbf{x}_i \mathbf{x}_i^\top$

If $\mathbf{u}$ is an eigenvector of *S* corresponding to nonzero eigenvalue

$$S\mathbf{u} = \frac{1}{n}\sum_i \mathbf{x}_i \mathbf{x}_i^\top \mathbf{u} = \lambda \mathbf{u} \Rightarrow \mathbf{u} = \frac{1}{n\lambda}\sum_i (\mathbf{x}_i^\top \mathbf{u})\mathbf{x}_i$$

Eigenvectors of S lie in the space spanned by all data points.

Nonlinear PCA using Kernels

$$S\mathbf{u} = \frac{1}{n}\sum_i \mathbf{x}_i \mathbf{x}_i^T \mathbf{u} = \lambda \mathbf{u} \Rightarrow \mathbf{u} = \frac{1}{n\lambda}\sum_i (\mathbf{x}_i^T \mathbf{u})\mathbf{x}_i$$

The covariance matrix can be written in matrix form:

$$S = \frac{1}{n} XX^T, \text{ where } \mathrm{X} = [\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_n].$$

$$\mathbf{u} = \sum_i \alpha_i \mathbf{x}_i = X\boldsymbol{\alpha} \qquad S\mathbf{u} = \frac{1}{n} XX^T X\boldsymbol{\alpha} = \lambda X\boldsymbol{\alpha}$$

$$\frac{1}{n}(X^T X)(X^T X)\boldsymbol{\alpha} = \lambda (X^T X)\boldsymbol{\alpha}$$

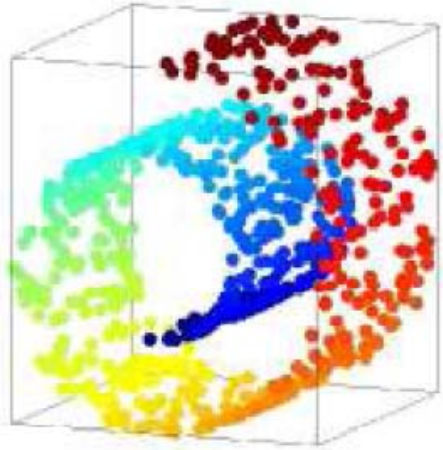
Any benefits?

$$\frac{1}{n}(X^T X)\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha} \qquad \frac{1}{n}K\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha}$$

Comments on PCA

- Linear dimensionality reduction method
- Can be kernelized
- Many nonlinear dimensionality reduction methods (Isomap, graph Laplacian eigenmap, and locally linear embedding/LLE) can be described as kernel PCA with a special kernel

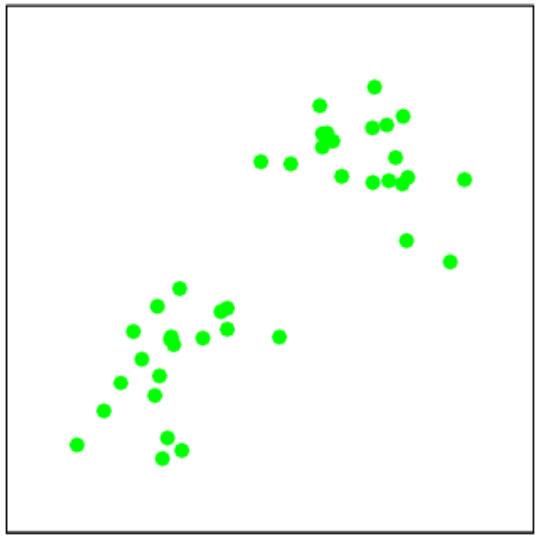
- Non-convex optimization problem
- But easy to solve…



From supervised to unsupervised classification

CLUSTERING

Clustering

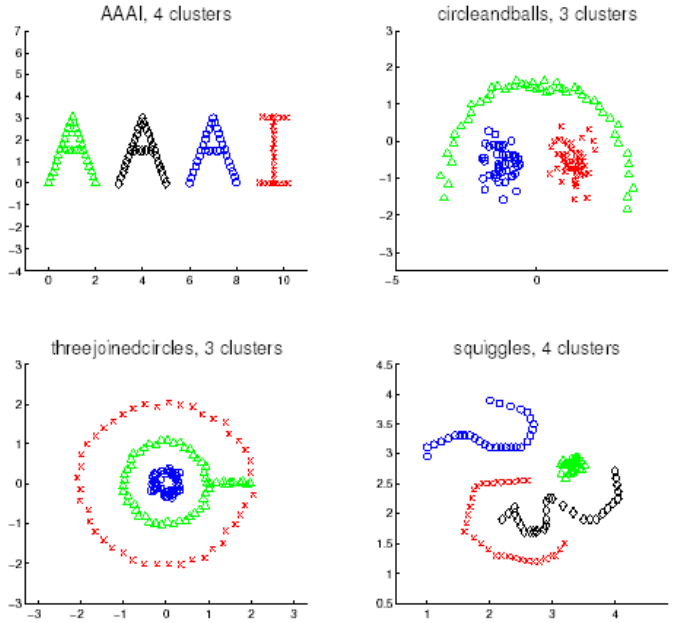


Are there any “groups” in the data ?
What is each group ?
How many ?
How to identify them?

Clustering

- Group the data objects into subsets or “clusters”:
  - High similarity within clusters
  - Low similarity between clusters

- A common and important task that finds many applications in Science, Engineering, information Science, and other places
  - Group genes that perform the same function
  - Group individuals that has similar political view
  - Categorize documents of similar topics
  - Identify similar objects from pictures



Clustering

Input: training set of input points

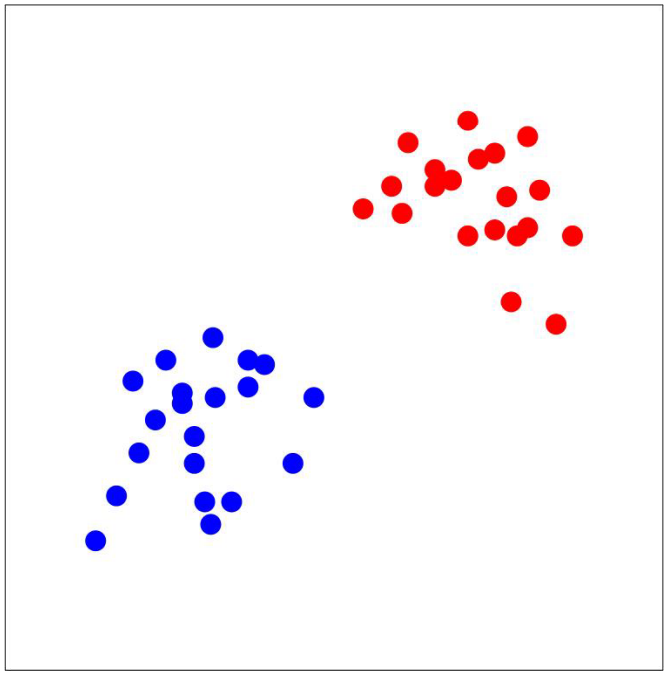
$$D_{train} = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$$

Output: assignment of each point to a cluster

$(C(1), ..., C(n))$ where $C(i) \in \{1, ..., k\}$

K-means clustering

- Create centers and assign points to centers to minimize sum of squared distance



K-means objective

- Each cluster is represented by a centroid μ
- Encode each point by its cluster center, pay a cost for deviation
- Loss function based on reconstruction

$$Loss_{\text{kmeans}} \sum_{j=1}^{n} \|\mu_{C(j)} - \mathbf{x}_j\|^2$$

K-means algorithm

- Goal: $\min_{\mu} \min_{C} \sum_{j=1}^{n} \|\mu_{C(j)} - \mathbf{x}_j\|^2$

- Strategy: alternating minimization
  - Step 1: if know cluster centers μ, can find best *C*
  - Step 2: if know cluster assignments *C*, can find best cluster centers

K-means algorithm

Optimize loss function $Loss(\mu, C)$

$$\min_{\mu} \min_{C} \sum_{j=1}^{n} \|\mu_{C(j)} - \mathbf{x}_j\|^2$$

(1) Fix μ, optimize C

$$\min_{C(1),C(2),\ldots,C(n)} \sum_{j=1}^{n} \|\mu_{C(j)} - \mathbf{x}_j\|^2$$

Assign each point to the nearest cluster center

(2) Fix C, optimize μ

$$\min_{\mu(1),\mu(2),\ldots,\mu(k)} \sum_{j=1}^{n} \|\mu_{C(j)} - \mathbf{x}_j\|^2$$

Solution: average of points in cluster i, exactly second step (re-center)

A few facts about K-means

- Simple and efficient
- Always converges
  - Why?
  - To a local minimum

- But…
  - K-means problem is **NP-hard**
  - No global solution
  - Not robust to noise and outliers

Want to Learn More?

- **Machine Learning: a Probabilistic Perspective,** *K. Murphy*
- **Pattern Classification,** *R. Duda, P. Hart, and D. Stork.* Standard pattern recognition textbook. Limited to classification problems. Matlab code. http://rii.ricoh.com/~stork/DHS.html
- **Pattern recognition and machine learning.** C. Bishop
- **The Elements of statistical Learning: Data Mining, Inference, and Prediction.** *T. Hastie, R. Tibshirani, J. Friedman,* Standard statistics textbook. Includes all the standard machine learning methods for classification, regression, clustering. R code. http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/
- **Introduction to Data Mining,** *P.-N. Tan, M. Steinbach, V. Kumar.* Addison-Wesley, 2006
- **Principles of Data Mining,** *D. Hand, H. Mannila, and P. Smyth*. MIT Press, 2001

- 统计学习方法，李航

Wrapping up the course

课程大纲

- 第一部分：人工智能概述/Introduction and Agents (chapters 1,2)
- 第二部分：问题求解/Search (chapters 3,4,5,6)
- 第三部分：知识与推理/Logic (chapters 7,8,9)
- 第四部分：不确定知识与推理/Uncertainty (chapters 13,14)
- 第五部分：学习/Learning (chapters 18,20)

Chap1-2. Intro

- Rational agents
- The performance measure evaluates the environment sequence
- PEAS descriptions define task environments

- Environments are categorized along several dimensions:

  observable? deterministic? episodic? static? discrete? single-agent?

Chap3. Uninformed search

A problem can be defined by 4 items: initial state（初始状态）, actions（行动） or successor function（后继函数）, goal test（目标测试）, path cost（路径损耗）

A solution is a sequence of actions leading from the initial state to a goal state

Uninformed search strategies use only the information available in the problem definition

- Breadth-first search
- Depth-first search
- Depth-limited search
- Iterative deepening search

Chap4. Informed search

- Best-first search（最佳优先搜索）
  - Greedy best-first search
  - A* search
- Heuristics
  - Heuristics（启发函数）
  - Admissible heuristics（可采纳的启发函数）
  - Relaxed problems（松弛问题）
- Local search algorithms
  - Hill-climbing search
  - Simulated annealing search（模拟退火搜索）
  - Genetic algorithms （遗传算法）

Chap6. Game search

- Games
- Perfect play（最优策略）
  - minimax decisions
  - α-β pruning（剪枝）
- Games of chance（包含几率因素的游戏）
  - ExpectiMinmax（期望极小极大值）

Chap7. Logical agents

Basic concepts of logic:

- **syntax: formal structure of sentences**
- **semantics: truth of sentences wrt models**
- **entailment: necessary truth of one sentence given another**
- **inference: deriving sentences from other sentences**
- **soundness: derivations produce only entailed sentences**
- **completeness: derivations can produce all entailed sentences**

Forward, backward chaining are linear-time, complete for Horn clauses

Resolution is complete for propositional logic

- 合取范式的转化

Chap8-9. First-Order Logic

- New concepts
  - Objects（对象）
  - Relations（关系）
  - Functions（函数）
- Inference
  - Unification合一
  - Forward and backward chaining
  - Resolution

Chap13. Uncertainty

Joint probability distribution specifies probability of every atomic event

全联合概率分布指定了对随机变量的每种完全赋值，即每个原子事件的概率

Inference

- Queries can be answered by summing over atomic events
  可以通过把对应于查询命题的原子事件的条目相加的方式来回答查询

- For nontrivial domains, we must find a way to reduce the joint size
  - Independence and conditional independence provide the tools

Bayes‘ Rule

Chap14. Bayesian networks

- Bayesian networks provide a natural representation for (causally induced) conditional independence
- Topology + CPTs = compact representation of joint distribution
- Inference
- Naïve Bayes model

Chap18,20. Learning

- Supervised learning
  - Given input data descriptions D = {x_1, x_2 . . . , x_n}, and target values y_1, y_2, . . . y_n, learn a prediction function $f(\mathbf{x}) \mapsto y$
  - Classification: y_1, y_2, . . . y_n are discrete class labels
    - Naïve Bayes model
    - Decision tree learning
    - Least squares classification
    - SVM
  - Regression: y continuous
    - Least squares regression

Chap18,20. Learning

- Unsupervised learning
  - Given input data descriptions $D = \{x_1, x_2 \ldots, x_n\}$, learn a prediction function $f(\mathbf{x}) \mapsto y$
  - Clustering: y discrete
    - Kmeans
  - Dimensionality reduction: y continuous
    - Principle component analysis