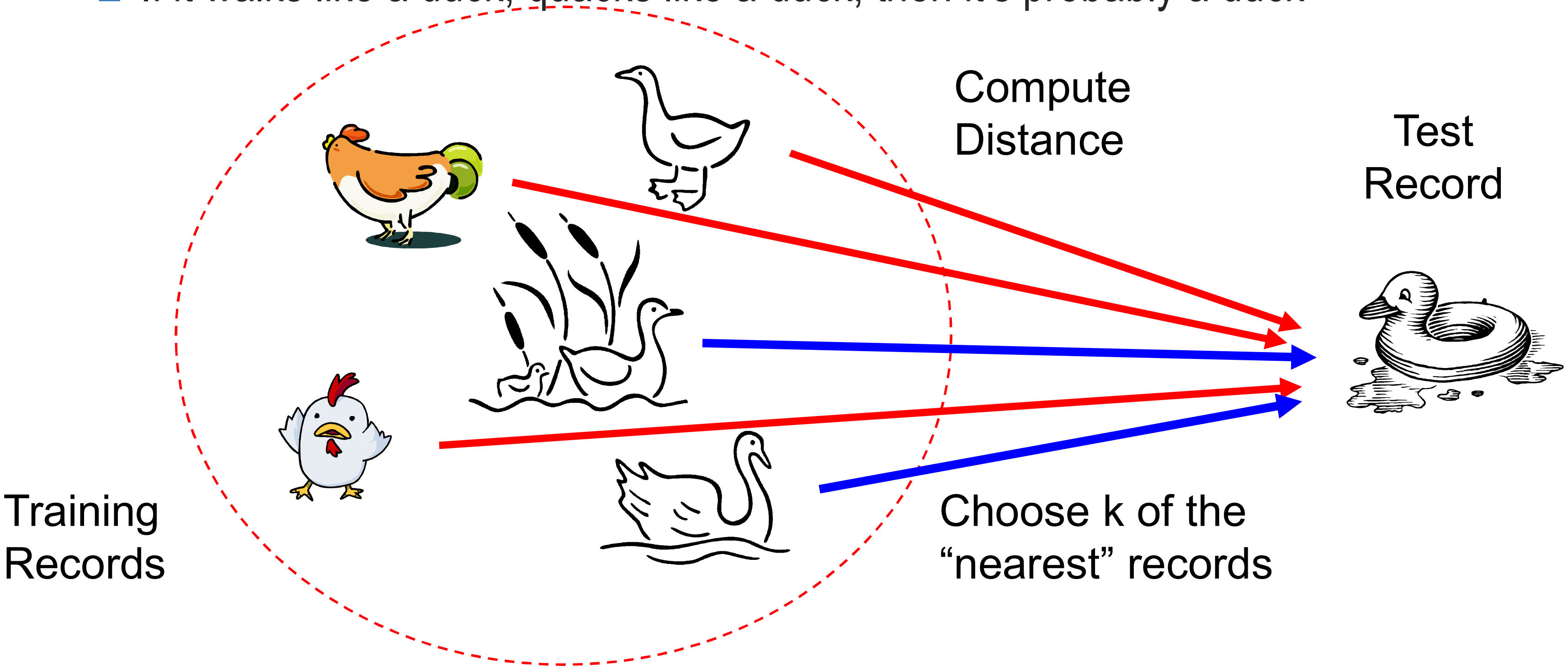
数据科学导论
Introduction to Data Science

第四章 数据挖掘基础

刘 淇
Email: qiliuql@ustc.edu.cn
课程主页：
http://staff.ustc.edu.cn/~qiliuql/DS2017.html

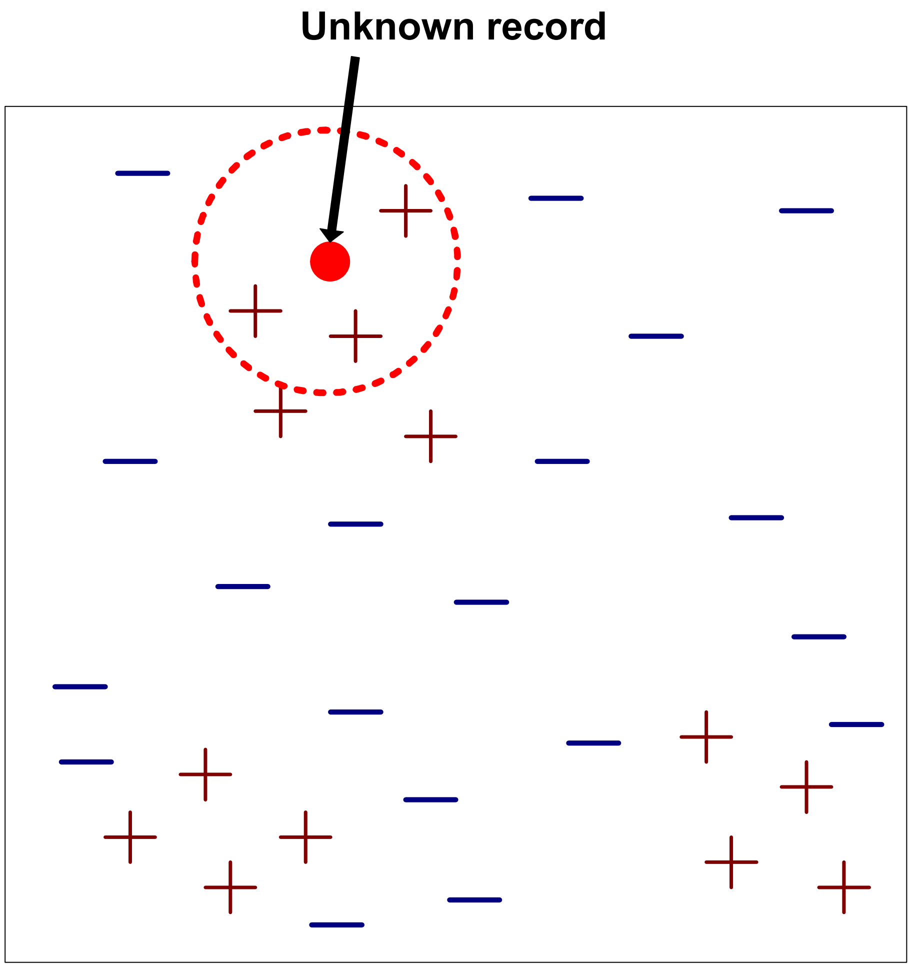
数据挖掘基础

- 分类——K近邻方法
  - 使用k个最近的点用来进行分类任务
- Basic idea:
  - If it walks like a duck, quacks like a duck, then it's probably a duck



数据挖掘基础

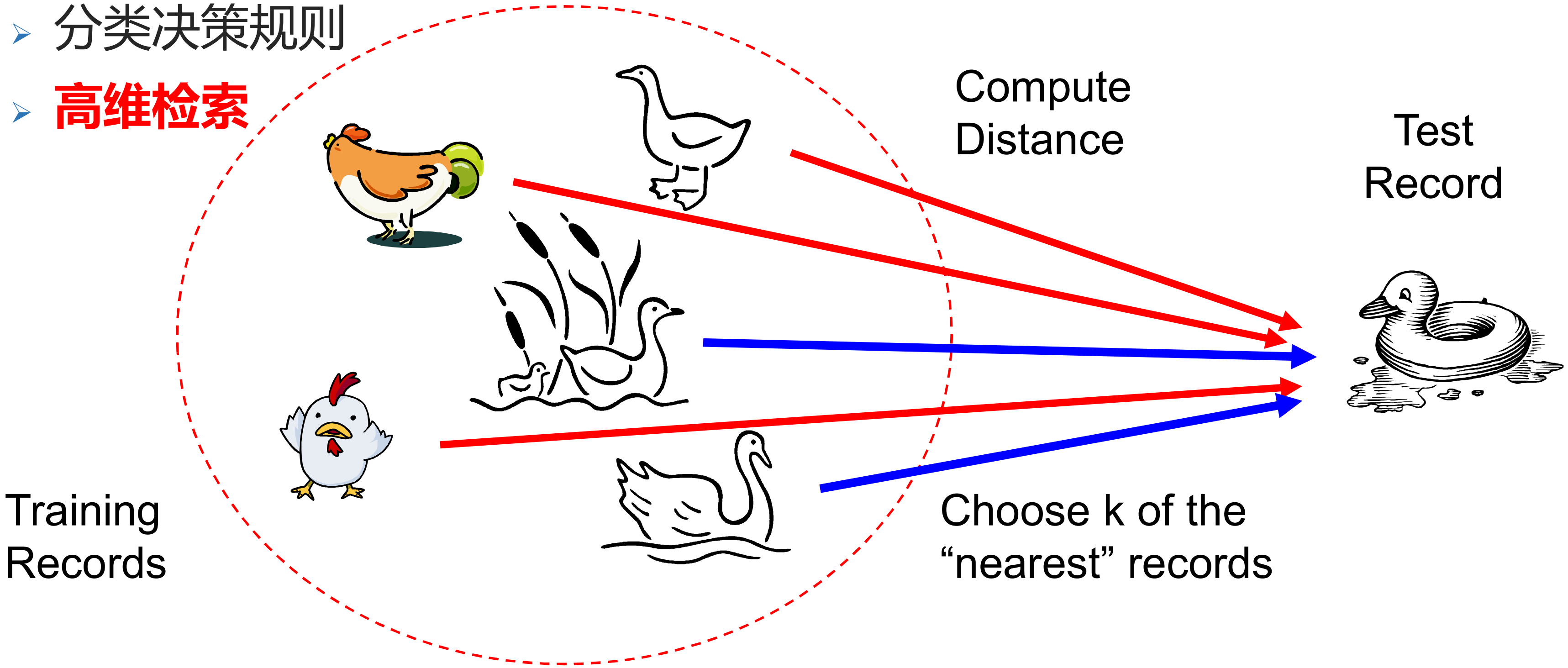
分类——K近邻方法



- Requires three things
  - The set of stored records
  - Distance Metric（距离矩阵）to compute distance between records
  - The value of *k*, the number of nearest neighbors to retrieve

- To classify an unknown record:
  - 计算到其他训练数据的距离
  - 找到 *k* 最近邻邻居
  - 使用邻居的label来预测未知数据的label(投票方法等)

- 分类——K近邻方法
  - 距离度量
  - k值选取
  - 分类决策规则
  - **高维检索**



- 分类——感知机（perceptron）
  - 1957年由Rosenblatt提出，是神经网络与支持向量机的基础
- 感知机，是二类分类的线性分类模型，其输入为样本的特征向量，输出为样本的类别，取+1和-1二值，即通过某样本的特征，就可以准确判断该样本属于哪一类。感知机能够解决的问题首先要求特征空间是线性可分的，再者是二类分类，即将样本分为{+1, -1}两类。由输入空间到输出空间的符号函数：

$$f(x) = sign(w \cdot x + b)$$

称为感知机，w和b为感知机参数，w为权值（weight），b为偏置（bias）。

数据挖掘基础

- 分类——感知机（perceptron）
- sign为符号函数：

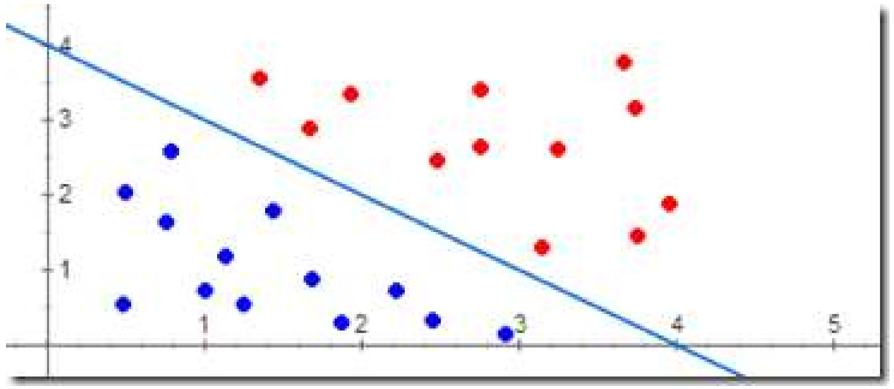
$$sign(x)=\begin{cases}+1, & x \geq 0\\ -1, & x<0\end{cases}$$

- 在感知机的定义中，线性方程w·x + b = 0对应于问题空间中的一个超平面S，位于这个超平面两侧的样本分别被归为两类，例如下图，红色作为一类，蓝色作为另一类，它们的特征很简单，就是它们的坐标

目标函数：

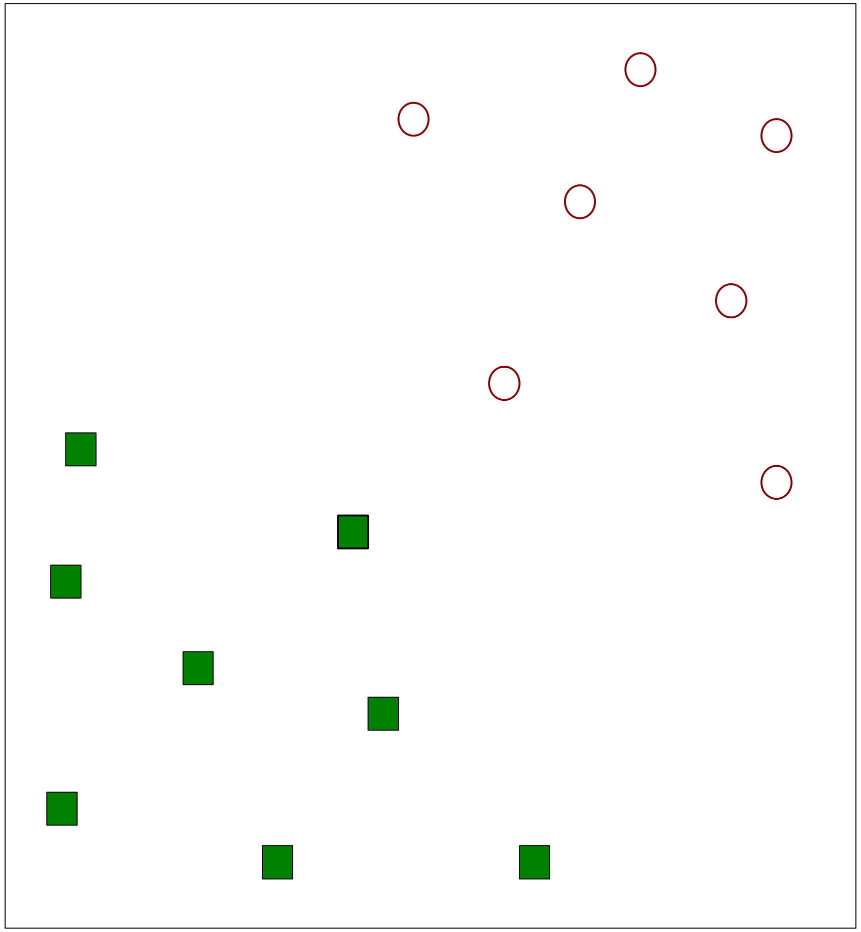
$$\min_{w,b} L(w,b) = -\sum_{x_i \in M} y_i(w \cdot x_i + b)$$

其中M是错分类的数据集合

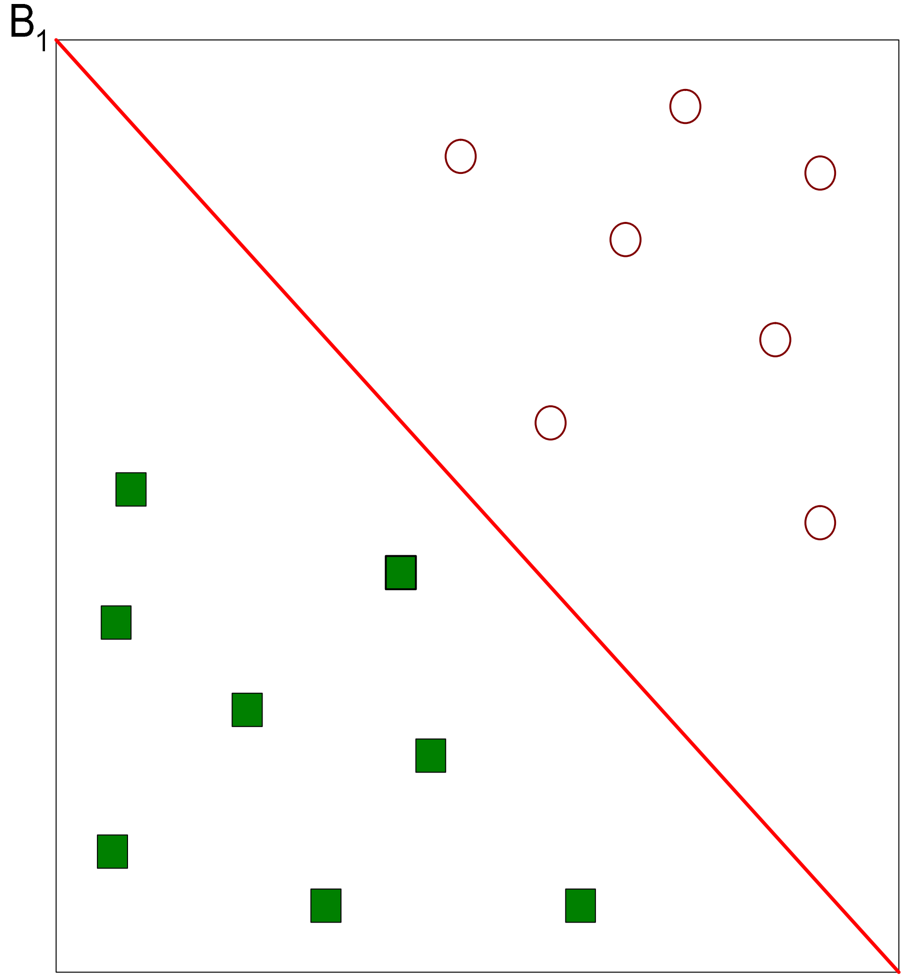


数据挖掘基础

- 分类——感知机（**perceptron**）

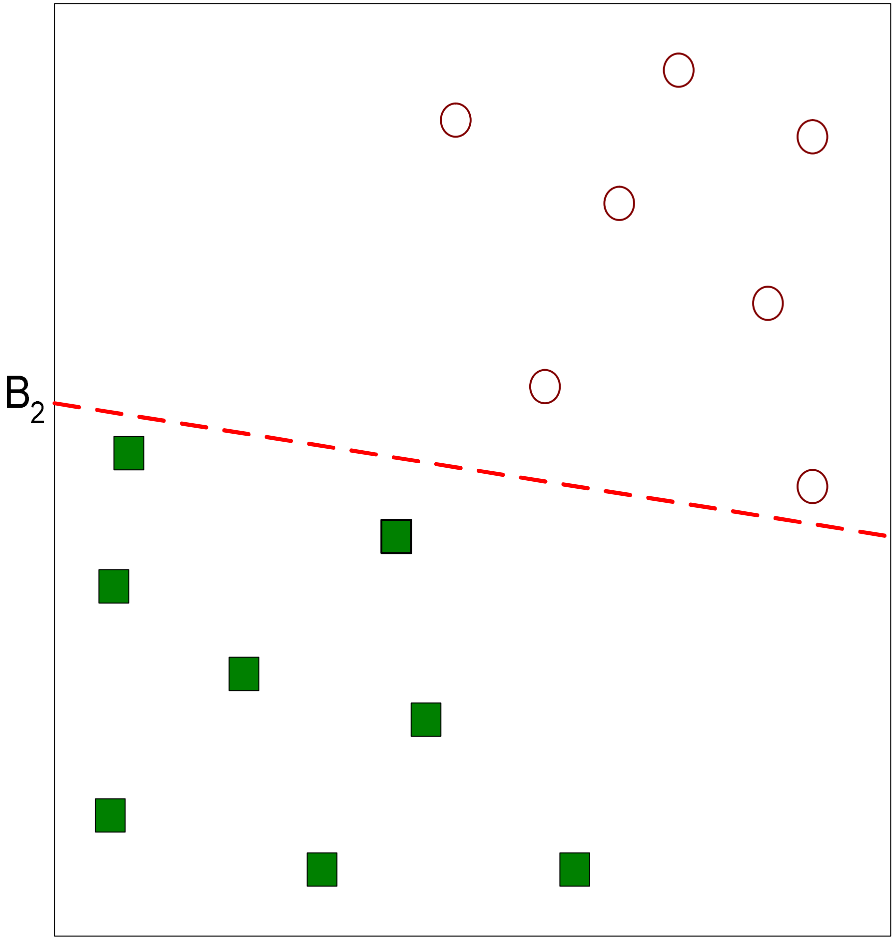


- 分类——支持向量机（Support Vector Machine）



一个可行解

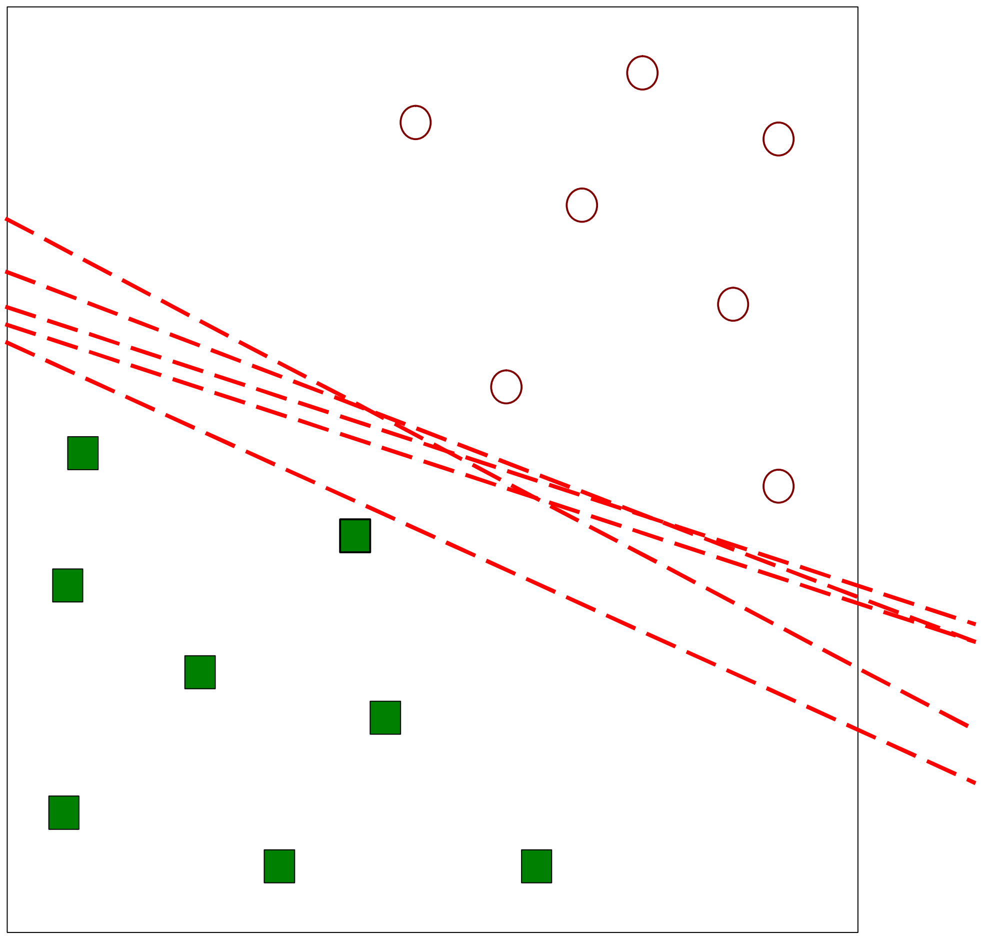
- 分类——支持向量机（**Support Vector Machine**）



另一个可行解

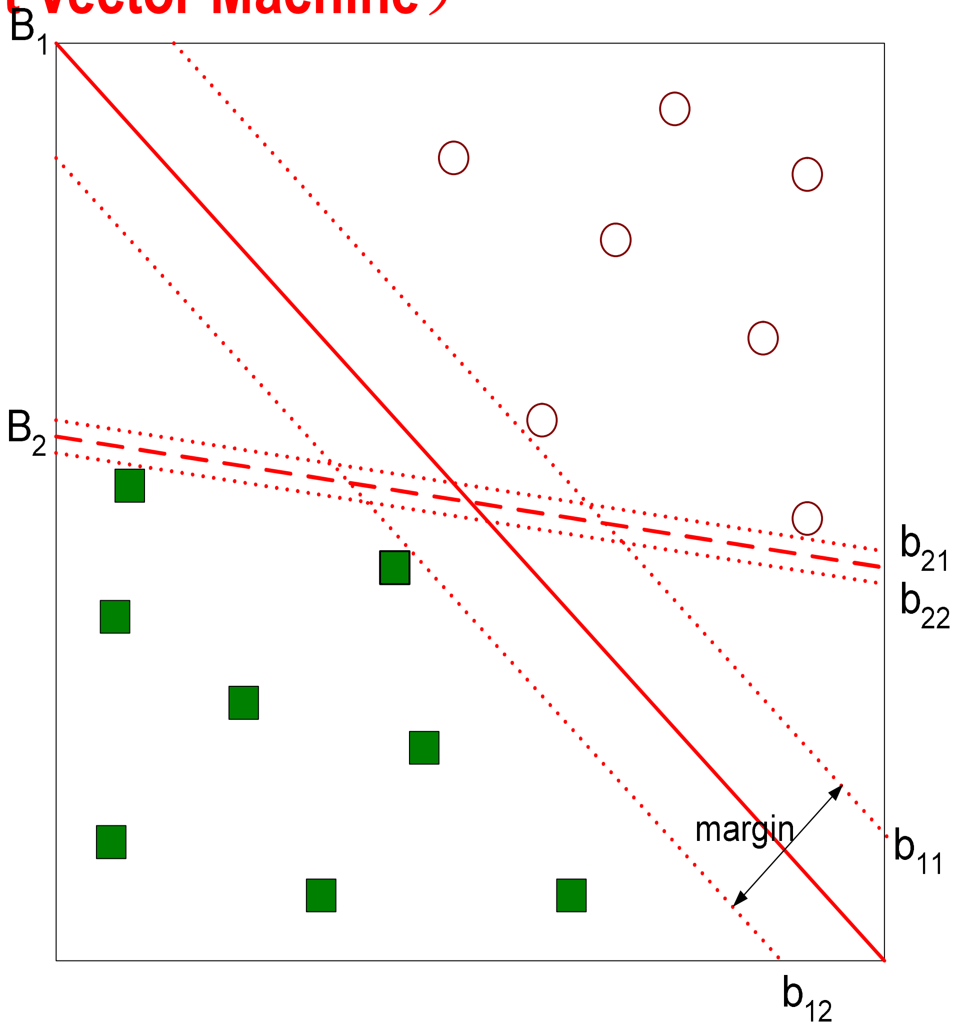
数据挖掘基础

- 分类——支持向量机（Support Vector Machine）



其他可行解

- 分类——支持向量机（**Support Vector Machine**）

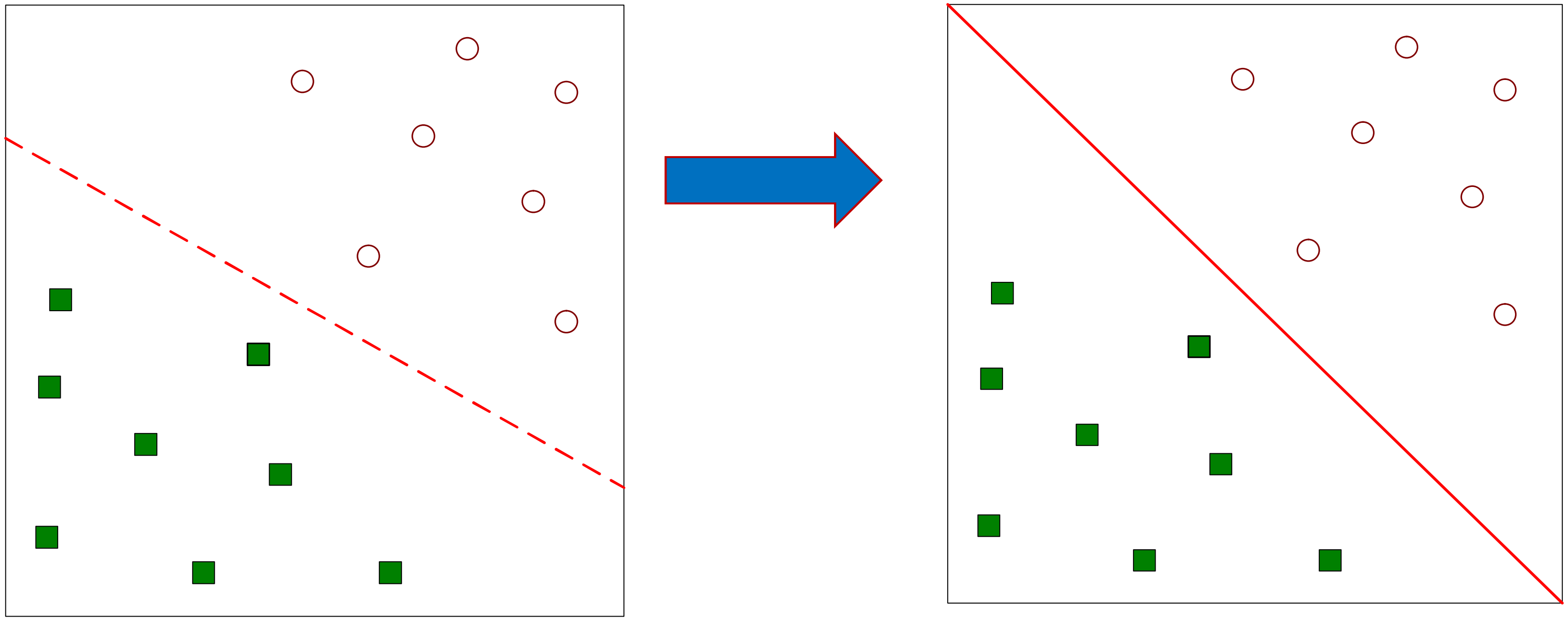


找到使间隔最大化的超平面

=>B1比B2更好

- 分类——区别

感知机 $f(x) = sign(w \cdot x + b)$ SVM



优化目标：

$$\min_{w,b} L(w,b) = -\sum_{x_i \in M} y_i (w \cdot x_i + b)$$

$$\min_{w,b} \frac{1}{2}\|w\|^2$$

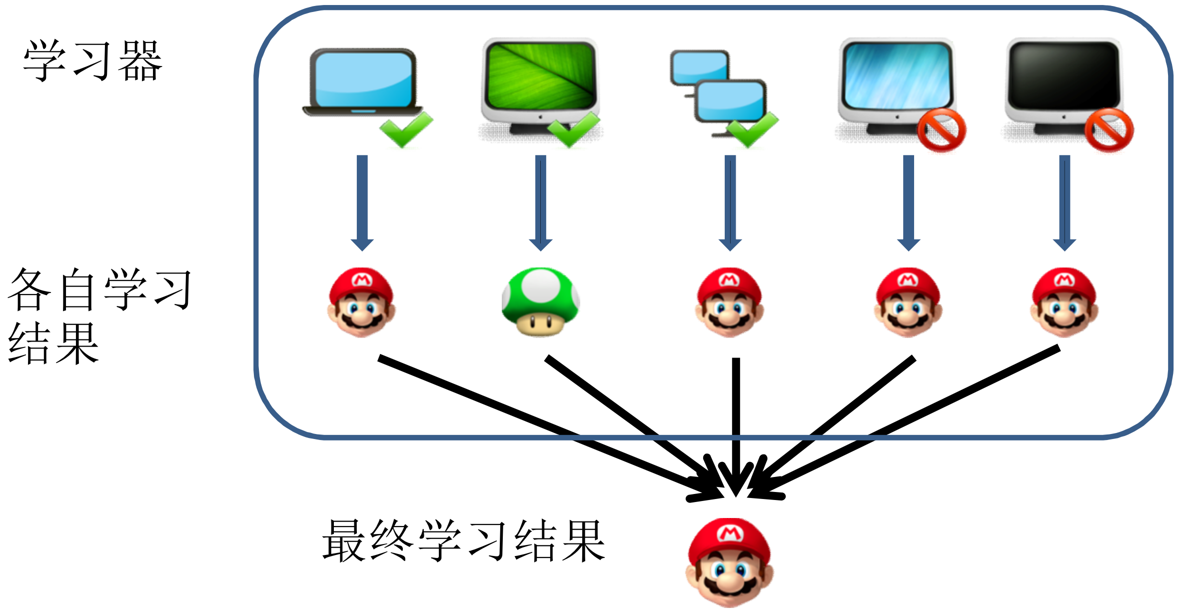
$$s.t.\ y_i(w^T.x_i + b) \geq 1$$

数据挖掘基础

- 常用方法——分类
  - 基本分类
    - 决策树
    - 规则方法
    - 贝叶斯方法
    - 最近邻方法
    - 支持向量机（SVM）
    - 神经网络
  - 集成分类
    - Boosting, Bagging, 随机森林
  - 模型评估方法
  - Class Imbalance Problem(类不平衡问题)

分类——集成学习

- All the competitors of data mining competition, such as KDD CUP, adopt ensemble methods to enhance the performance of their algorithm.
  - Bagging(装袋)、Boosting(提升)
- General Idea



数据挖掘基础

- 分类——集成学习： Bagging（装袋）
  - Decision Tree
  - X<=0.35 or X<=0.75 with precision 70%

x	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
y	1	1	1	-1	-1	-1	-1	1	1	1

  - Bagging

Round 1

x	0.1	0.2	0.2	0.3	0.4	0.4	0.5	0.6	0.9	0.9
y	1	1	1	1	-1	-1	-1	-1	-1	-1

X<=0.35 y=1
X>0.35 y=-1

Round 2

x	0.1	0.2	0.3	0.4	0.5	0.8	0.9	1	1	1
y	1	1	1	-1	-1	-1	-1	1	-1	-1

X<=0.65 y=1
X>0.65 y=-1

One Round ,
One Classifier

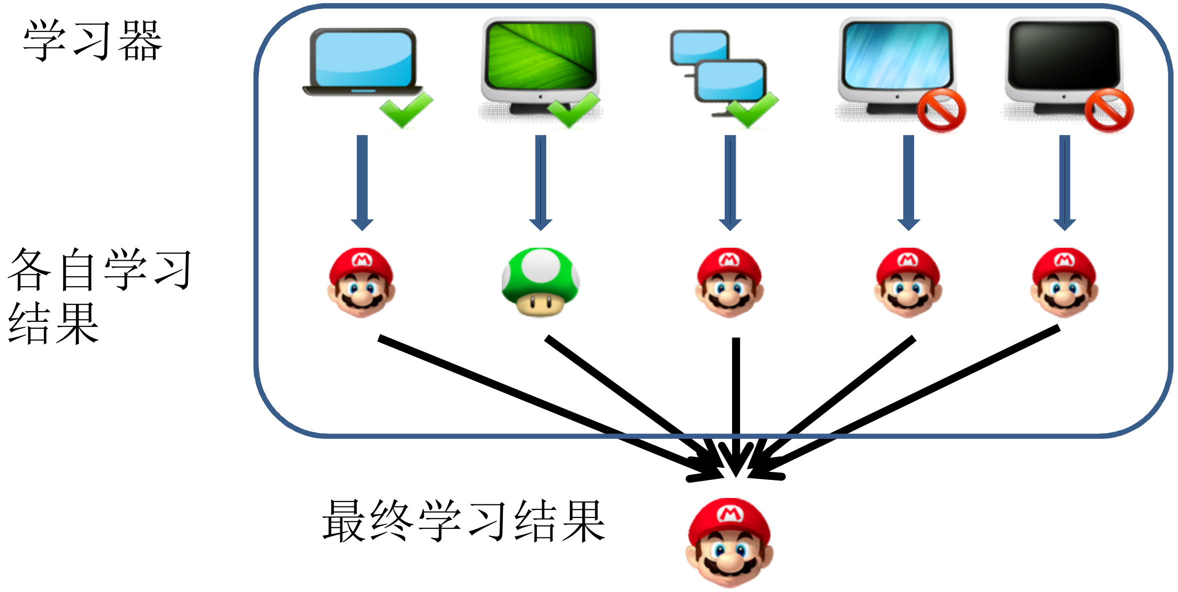
0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
1	1	1	-1	-1	-1	-1	1	1	1

Round	x=0.1	x=0.2	x=0.3	x=0.4	x=0.5	x=0.6	x=0.7	x=0.8	x=0.9	x=1.0
1	1	1	1	-1	-1	-1	-1	-1	-1	-1
2	1	1	1	1	1	1	1	1	1	1
3	1	1	1	-1	-1	-1	-1	-1	-1	-1
4	1	1	1	-1	-1	-1	-1	-1	-1	-1
5	1	1	1	-1	-1	-1	-1	-1	-1	-1
6	-1	-1	-1	-1	-1	-1	-1	1	1	1
7	-1	-1	-1	-1	-1	-1	-1	1	1	1
8	-1	-1	-1	-1	-1	-1	-1	1	1	1
9	-1	-1	-1	-1	-1	-1	-1	1	1	1
10	1	1	1	1	1	1	1	1	1	1
Sum	2	2	2	-6	-6	-6	-6	2	2	2
Sign	1	1	1	-1	-1	-1	-1	1	1	1
True Class	1	1	1	-1	-1	-1	-1	1	1	1

Figure 5.36. Example of combining classifiers constructed using the bagging approach.

Accuracy of ensemble classifier: 100% ☺

- 分类——集成学习： Bagging Summary
  - Works well if the base classifiers are unstable (complement each other)
  - Increased accuracy because it ***reduces the variance*** （*方差*） of the individual classifier（提升准确率的原因）
  - **Does not focus on any particular instance of the training data**
    - Therefore, less susceptible to model over-fitting when applied to noisy data
  - What if we want to focus on a particular instances of training data?



数据挖掘基础

- 分类——集成学习： Boosting(提升)
- An iterative procedure to adaptively change distribution of training data by **focusing more on previously misclassified records**
  - Initially, all N records are assigned equal weights（每个基分类器开始权值是相同的）
  - Unlike bagging, weights may change at the end of a boosting round（训练后权值会发生改变）

- 分类——集成学习： Boosting(提升)
- Records that are wrongly classified will have their weights increased（错误分类的权值会得到提升）
- Records that are classified correctly will have their weights decreased（正确分类的权值会下降）

Original Data	1	2	3	4	5	6	7	8	9	10
Boosting (Round 1)	7	3	2	8	7	9	4	10	6	3
Boosting (Round 2)	5	4	9	4	2	5	1	7	4	2
Boosting (Round 3)	4	4	8	10	4	5	4	6	3	4

- **Example 4 is hard to classify**
- Its weight is increased, therefore it is more likely to be chosen again in subsequent rounds

数据挖掘基础

- 分类——集成学习： Boosting(提升)
  - Adaboost (Adaptive Boost) Training
- Training data D contain N labeled data $(X_1,y_1), (X_2,y_2), (X_3,y_3),\ldots.(X_N,y_N)$
- Initially assign equal weight $1/N$ to each data（初始权值相同）
- To generate T base classifiers, we need T rounds or iterations（迭代 T 次）
  - Round i, data from D are sampled with replacement , to form D_i (size N)
- Each data's chance of being selected in the next rounds depends on its weight
  - Correctly classified: Decrease weight（分类器分类正确，权值下降）
  - Incorrectly classified: Increase weight（分类器分类错误，权值提高）

$$w_j^{(i+1)} = \frac{w_j^{(i)}}{Z_i}\begin{cases} \exp^{-\alpha_i} & \text{if } C_i(x_j) = y_j \\ \exp^{\alpha_i} & \text{if } C_i(x_j) \neq y_j \end{cases}$$

where Z_i is the normalization factor

- 分类——集成学习： Boosting(提升)
  - Adaboost (Adaptive Boost) Testing
    - **The lower a classifier error rate**, the more accurate it is, and therefore, **the higher its weight for voting** （投票）should be
    - Weight of a classifier C_i’ s vote is

$$\alpha_i = \frac{1}{2}\ln\left(\frac{1-\varepsilon_i}{\varepsilon_i}\right)$$

- Testing:
  - For each class c, sum the weights of each classifier that assigned class c to X (unseen data)
  - The class with the highest sum is the WINNER!

$$C^*(x_{test}) = \arg\max_y \sum_{i=1}^{T} \alpha_i \delta\left(C_i(x_{test}) = y\right)$$

分类——模型评估方法：混淆矩阵

- 着重于评估模型的预测能力
  - Rather than how fast it takes to classify or build models, scalability, etc.
- Confusion Matrix (混淆矩阵):

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

数据挖掘基础

- 分类——模型评估方法：混淆矩阵

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

分类——模型评估方法：样本不均衡问题

- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10
- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %
  - Accuracy is misleading because model does not detect any class 1 example

数据挖掘基础

分类——模型评估方法：Cost-Sensitive Measures

Count	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	a	b
ACTUAL CLASS Class=No	c	d

- 正确率 $$\text{Precision (p)} = \frac{a}{a+c}$$
- 召回率 $$\text{Recall (r)} = \frac{a}{a+b}$$
- F值 $$\text{F - measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

- Precision is biased towards C(Yes|Yes) & C(Yes|No)
- Recall is biased towards C(Yes|Yes) & C(No|Yes)
- F-measure is biased towards all except C(No|No)

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

数据挖掘基础

在一次垃圾邮件检测中，使用贝叶斯分类法认为有100篇邮件是垃圾邮件，后经过砖家判定，其中真是垃圾邮件的为60篇，其余的40篇为误分，那么请问本次分类的准确率Precision就等于__________。假如砖家发现邮件样本集里还有90篇垃圾邮件，由于各种原因而未被检出（漏检），那么按照上述公式，本次分类的查全率Recall就等于__________，F1值等于__________。

$$\text{Precision (p)} = \frac{TP}{TP+FP}$$

$$\text{Recall (r)} = \frac{TP}{TP+FN}$$

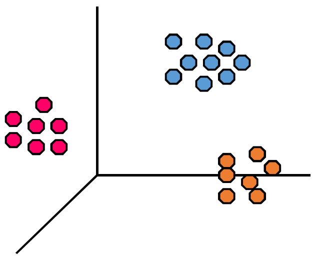
$$\text{F-measure }(\text{F}_1) = \frac{2rp}{r+p} = \frac{2\times TP}{2\times TP+FP+FN}$$

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

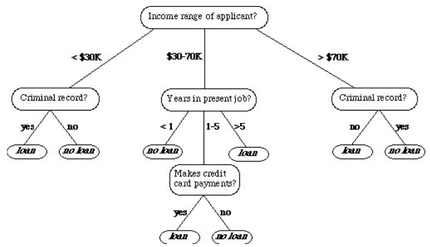
数据挖掘基础

- 常用方法——关于四个任务有哪些常用方法？

Clustering



Classification



Data

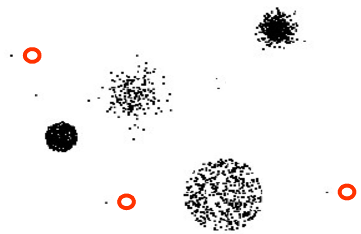
	T		H		P	
	L	H	L	H	L	H
J	-6.0	8.8	60	100	986	1044
F	-2.8	10.9	48	100	973	1025
M	-5.6	17.7	34	100	976	1037
A	-1.2	22.2	27	100	996	1036
M	-0.8	27.8	25	100	1003	1034
J	5.2	29.1	26	100	998	1030
J	9.8	30.6	23	99	997	1027
A	5.6	26.1	31	100	992	1029
S	5.2	24.8	35	100	998	1028
O	-0.4	21.3	42	100	990	1031
N	-7.6	17.3	55	100	963	1023
D	-10.4	9.2	53	100	987	1039

table 17a

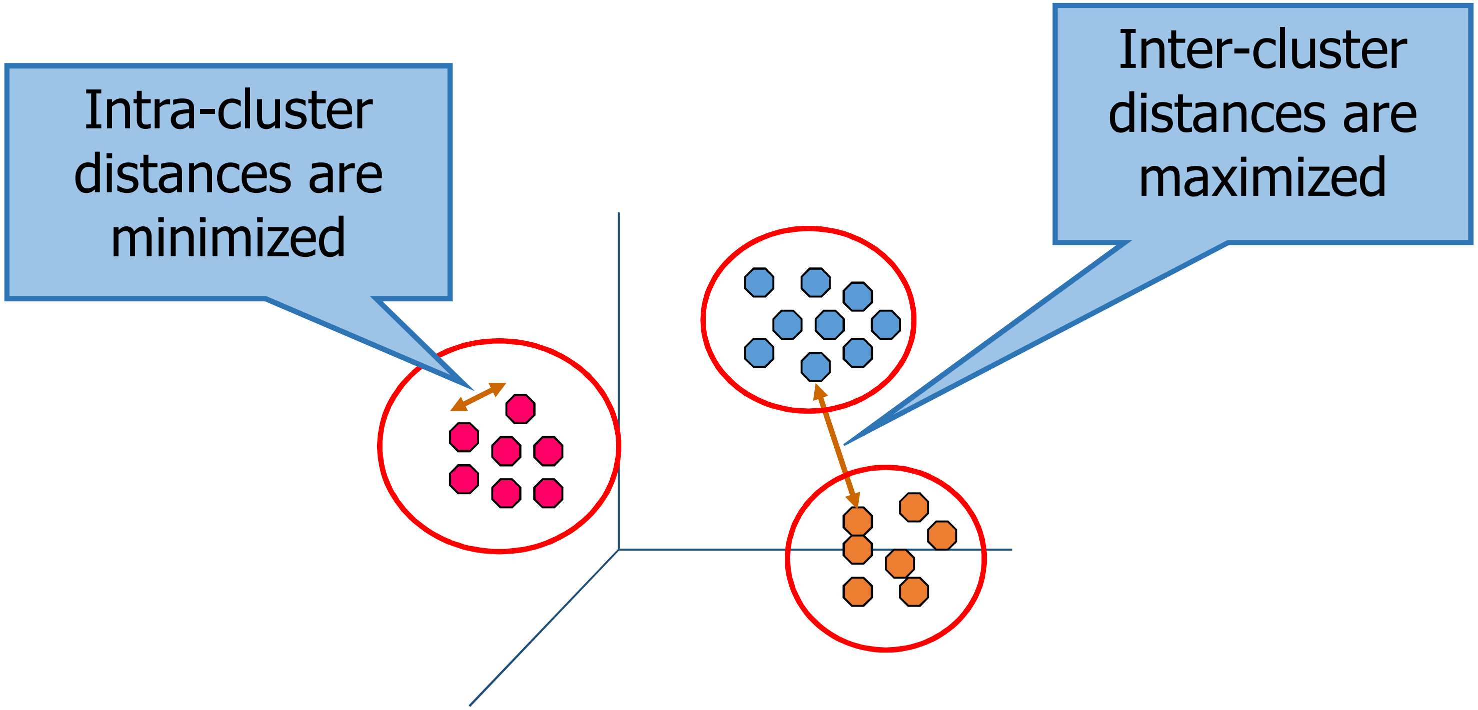
2010 monthly weather variation, Cambridge (UK)

Association Analysis

Anomaly Detection



- 四个任务——Clustering（聚类）
  - Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



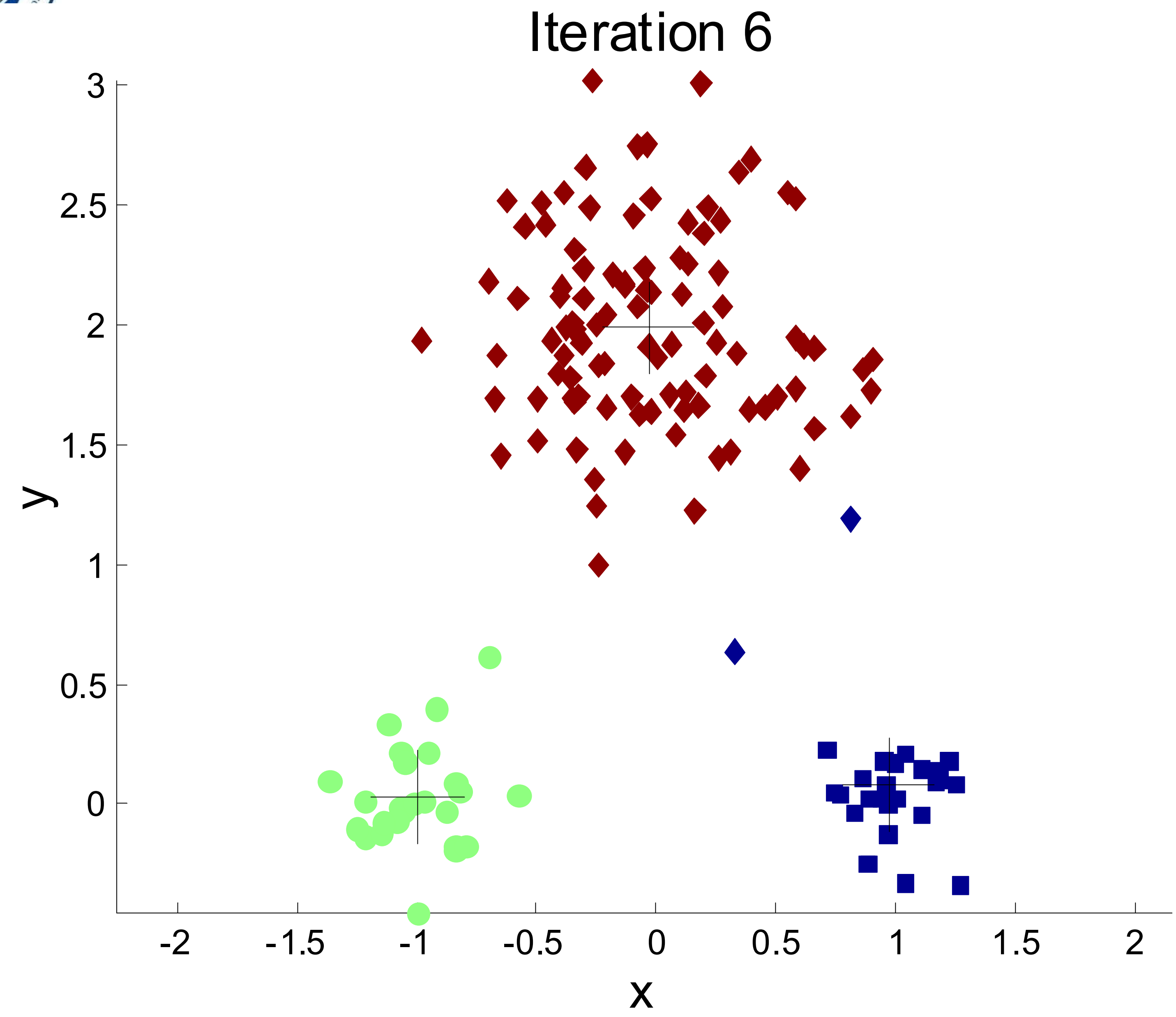
数据挖掘基础

- 常用方法——Clustering
  - K-means and its Variants （K均值聚类）
  - Hierarchical Clustering（层次聚类）
  - Density-based Clustering（密度聚类）
    - DBSCAN
  - Cluster Validation
  - 扩展方法
  - 面临挑战
    - Class Imbalance Problem(类不平衡问题)

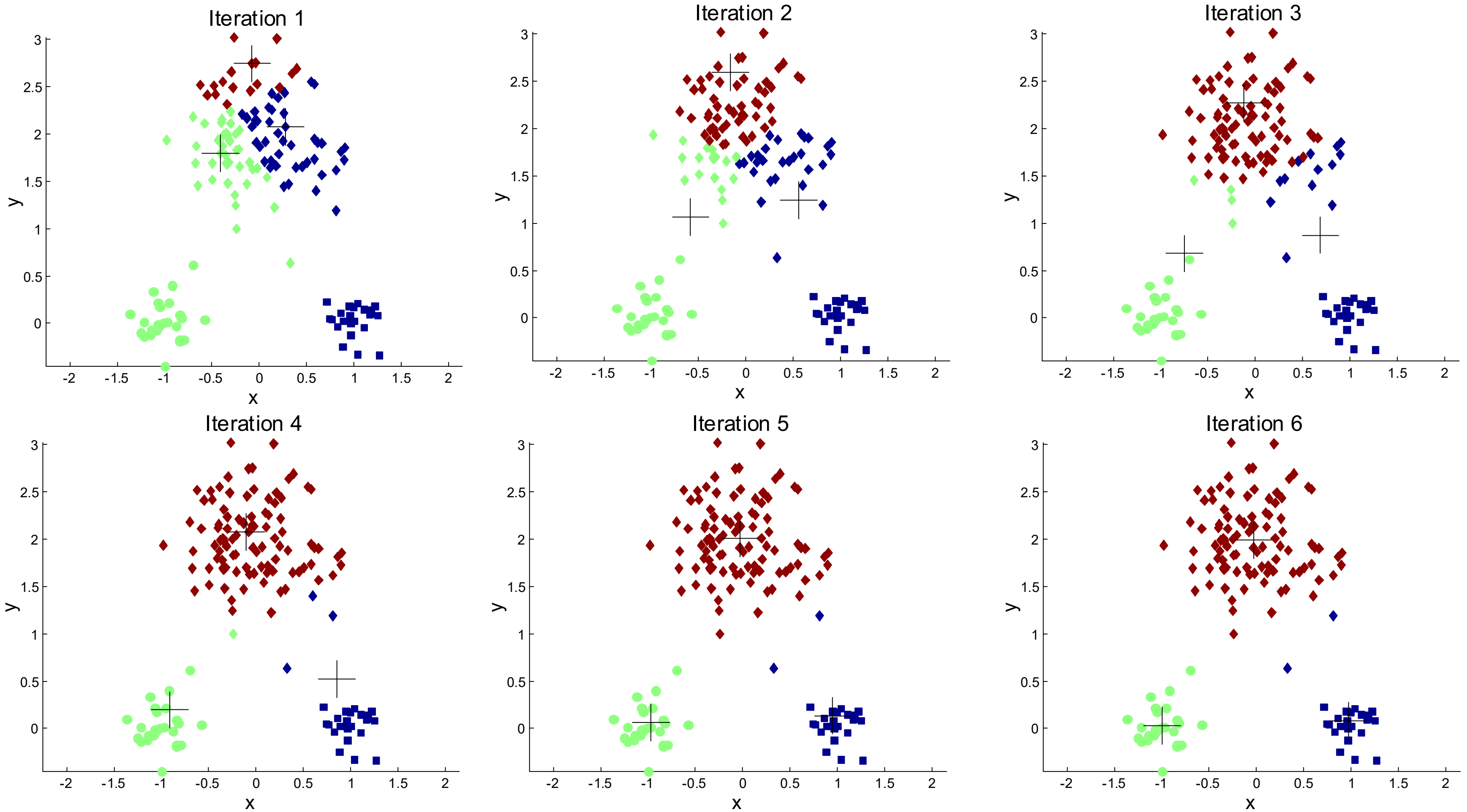
- Clustering——K-means and its Variants （ K均值聚类 ）
  - Partitional clustering approach
  - Number of clusters, K, must be specified
  - Each cluster is associated with a centroid (中心点)
  - Each point is assigned to the cluster with the closest centroid
  - The basic algorithm is very simple

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

Example of K-means Clustering



- Clustering——Example of K-means Clustering



Clustering——Characteristics of K-means Clustering

- Initial centroids are often chosen randomly.
- Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- 'Closeness' is measured by **Euclidean distance, cosine similarity, correlation**, etc.
- K-means will converge for common similarity measures mentioned above.
- Most of the convergence（收敛） happens in the first few iterations.
- Often the stopping condition is changed to 'Until relatively few points change clusters'
- Complexity is $O(n * K * I * d)$
- n = number of points, K = number of clusters,
  I = number of iterations, d = number of attributes

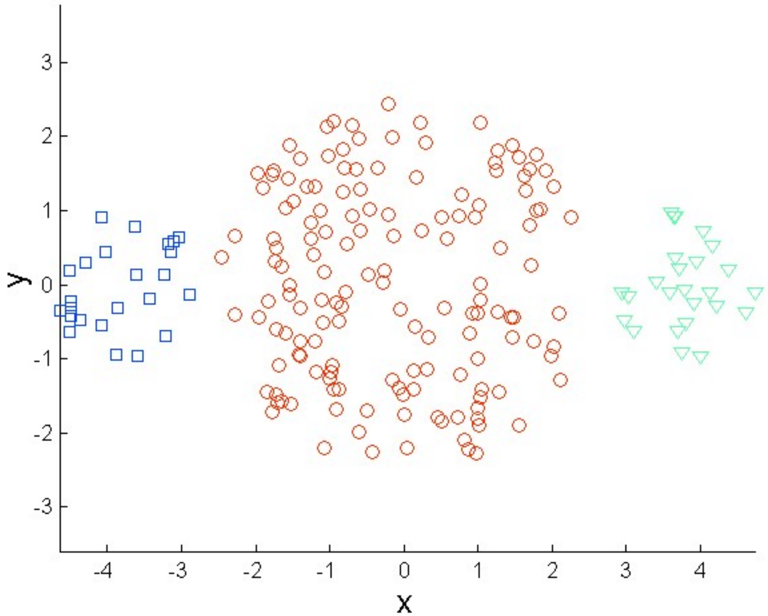
- Clustering——Evaluating K-means Clusters
  - Most common measure is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster
  - To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

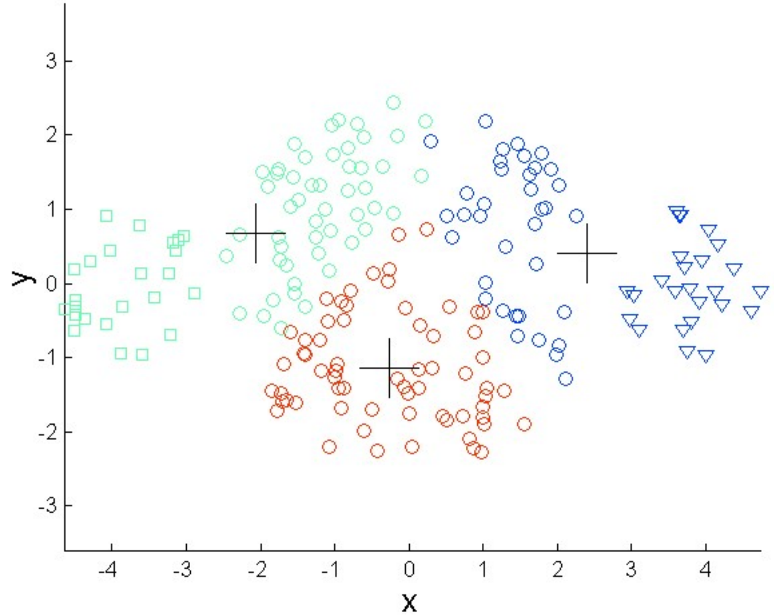
  - x is a data point in cluster C_i and m_i is the representative point(质心) for cluster C_i
  - can show that m_i corresponds to the center (mean) of the cluster
  - Given two sets of clusters, we prefer the one with the smallest error
  - One easy way to reduce SSE is to increase K, the number of clusters
  - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

- Clustering——Limitations of K-means
  - K-means has problems when clusters are of differing（不同的簇有不同的）
    - Sizes
    - Densities
    - Non-globular shapes
  - K-means has problems when the data contains outliers.

- Clustering——Limitations of K-means
  - Differing Sizes

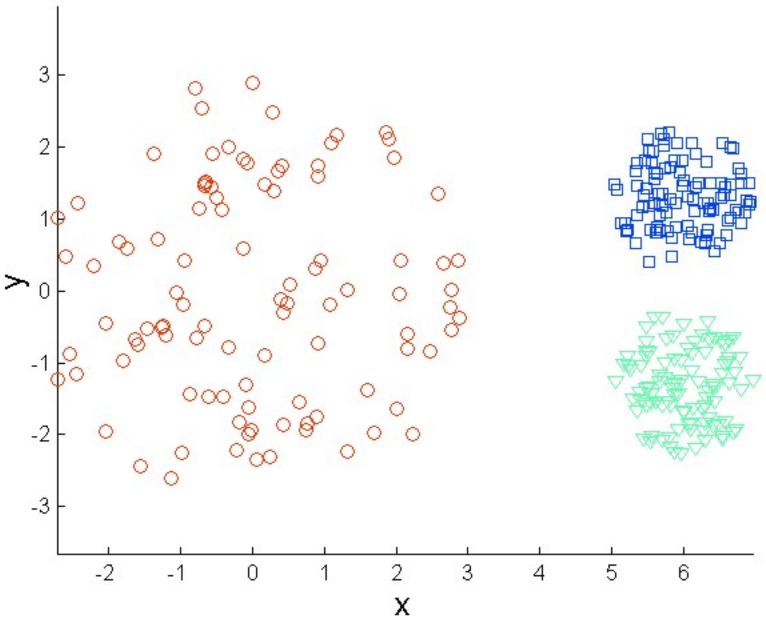


Original Points

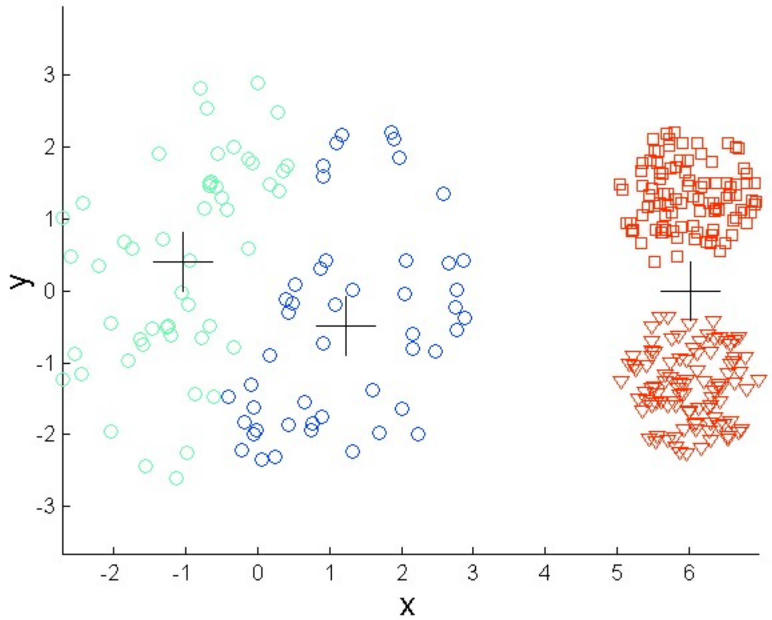


K-means (3 Clusters)

- Clustering——Limitations of K-means
  - Differing Densities



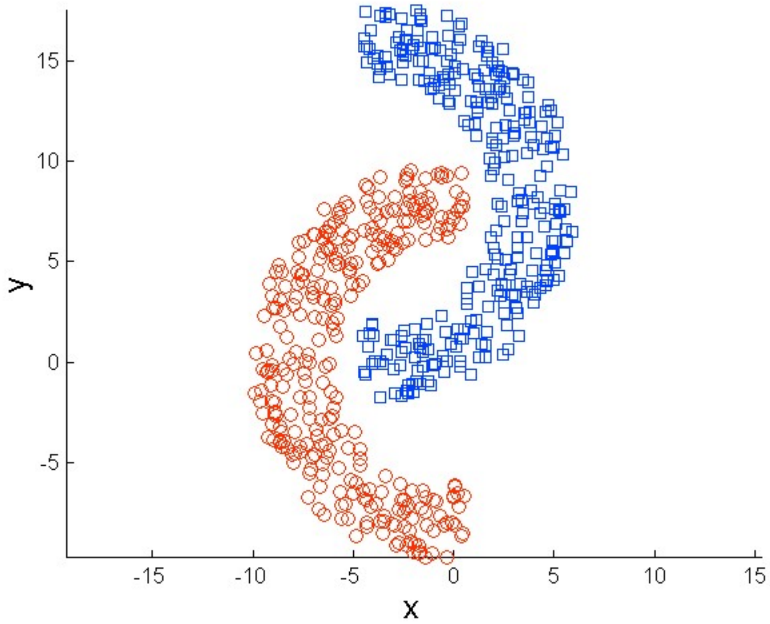
Original Points



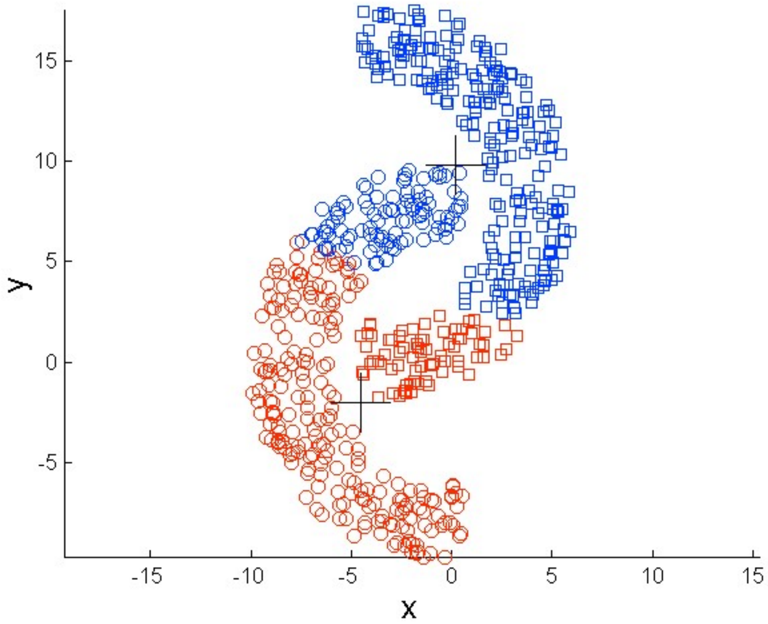
K-means (3 Clusters)

Clustering——Limitations of K-means

- Non-globular（非球形） Shapes

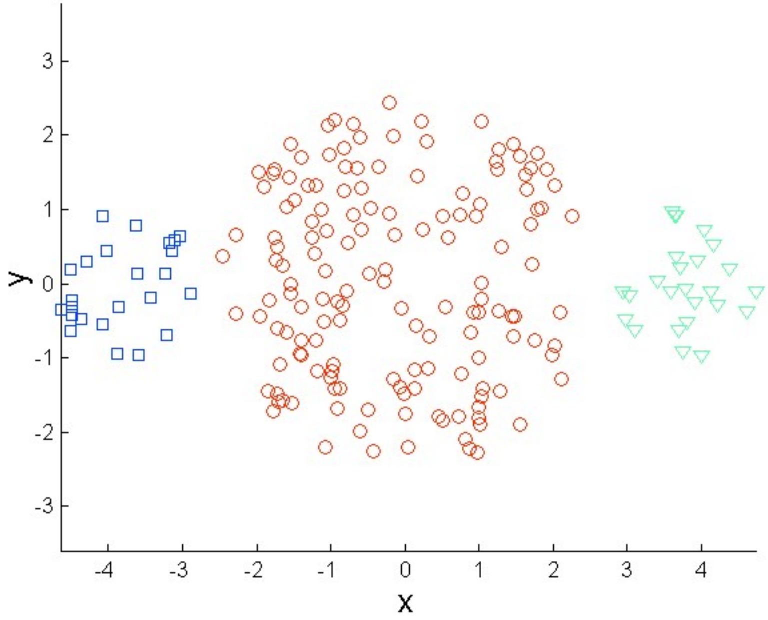


Original Points

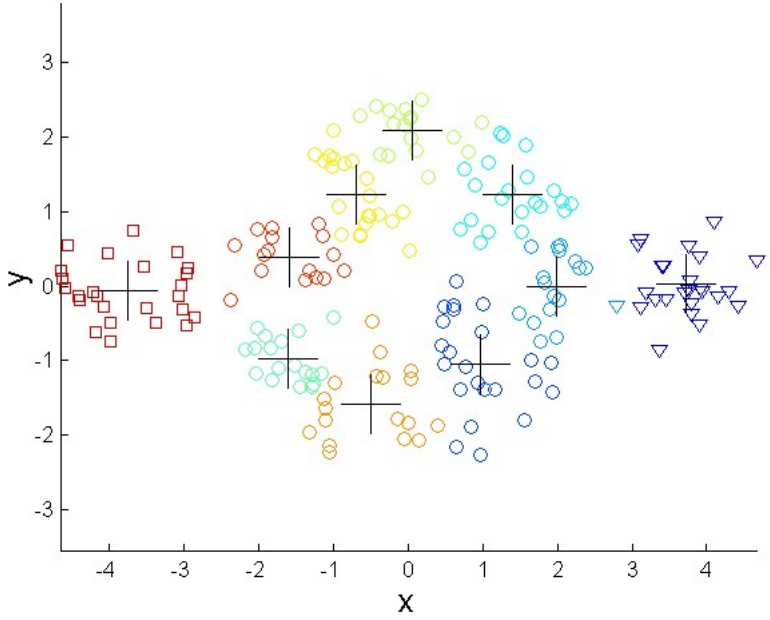


K-means (2 Clusters)

- Clustering——Overcoming K-means Limitations
  - One solution is to use many clusters.
    - Find parts of clusters, but need to put together.

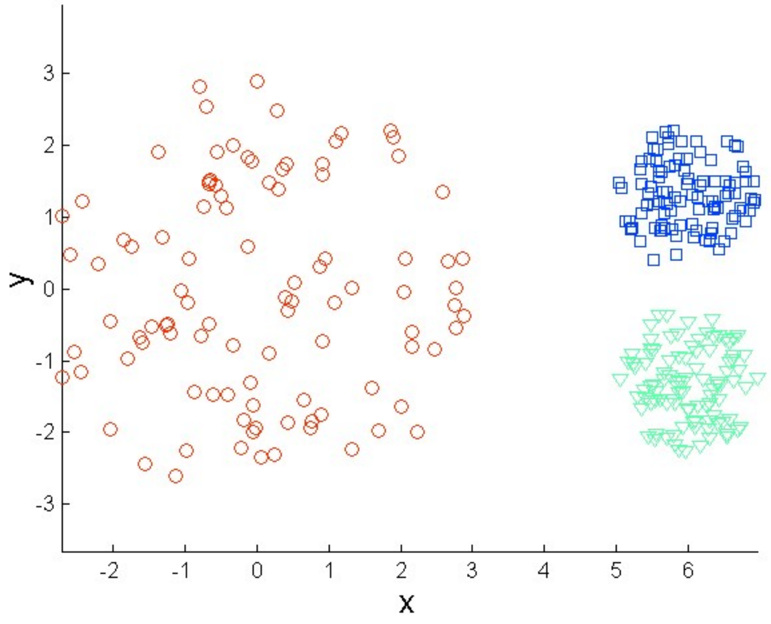


Original Points

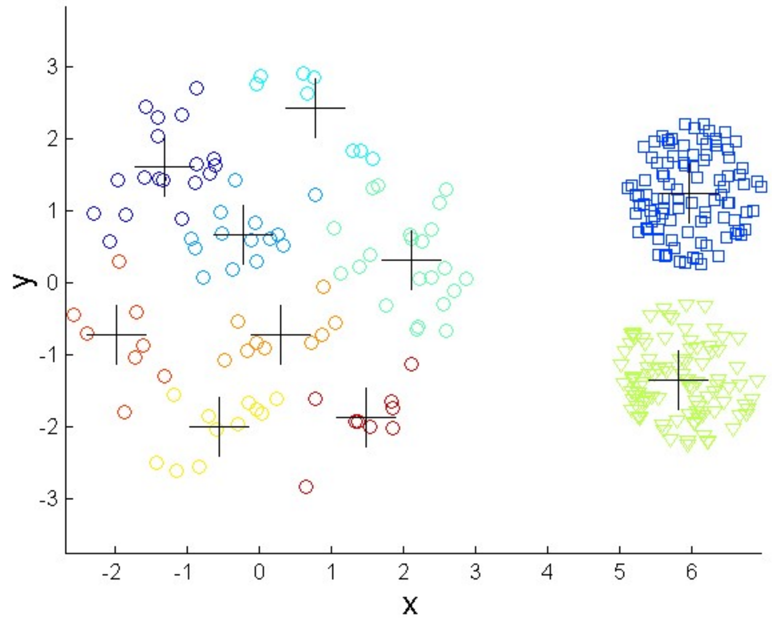


K-means Clusters

- Clustering——Overcoming K-means Limitations
  - One solution is to use many clusters.
    - Find parts of clusters, but need to put together.

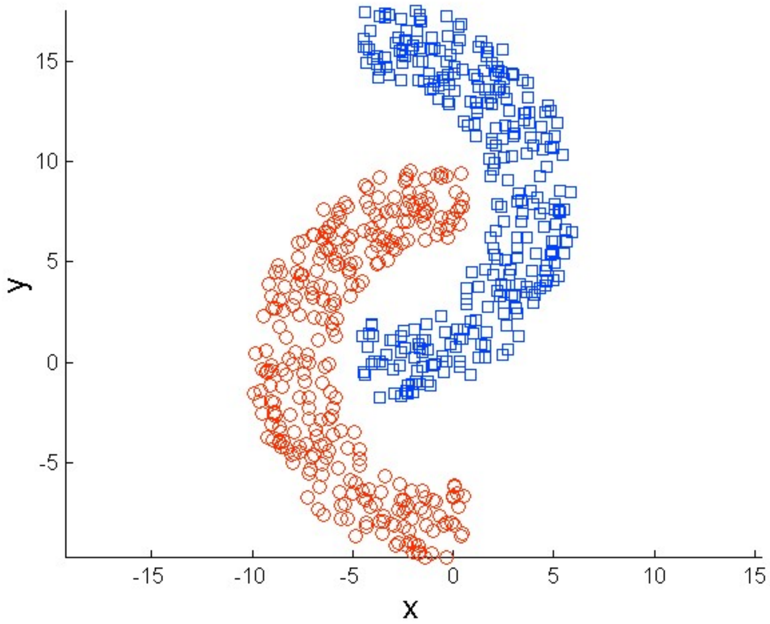


Original Points

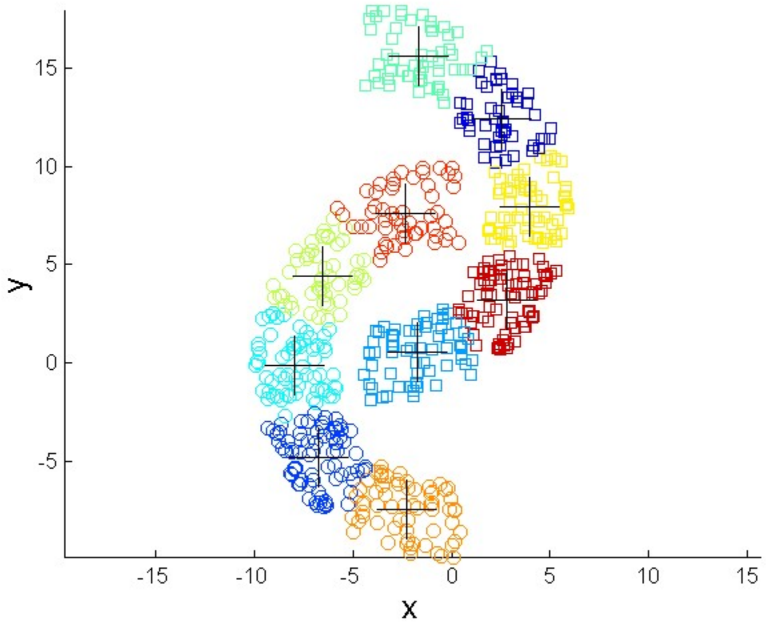


K-means Clusters

- Clustering——Overcoming K-means Limitations
  - One solution is to use many clusters.
    - Find parts of clusters, but need to put together.



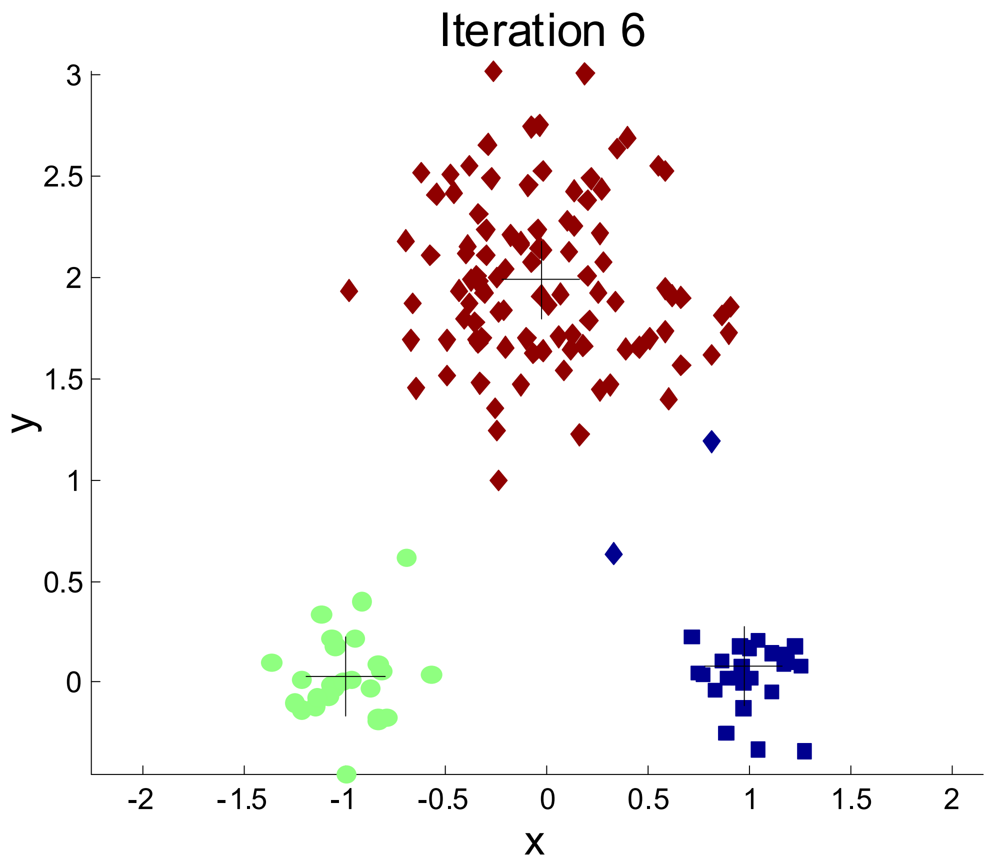
Original Points



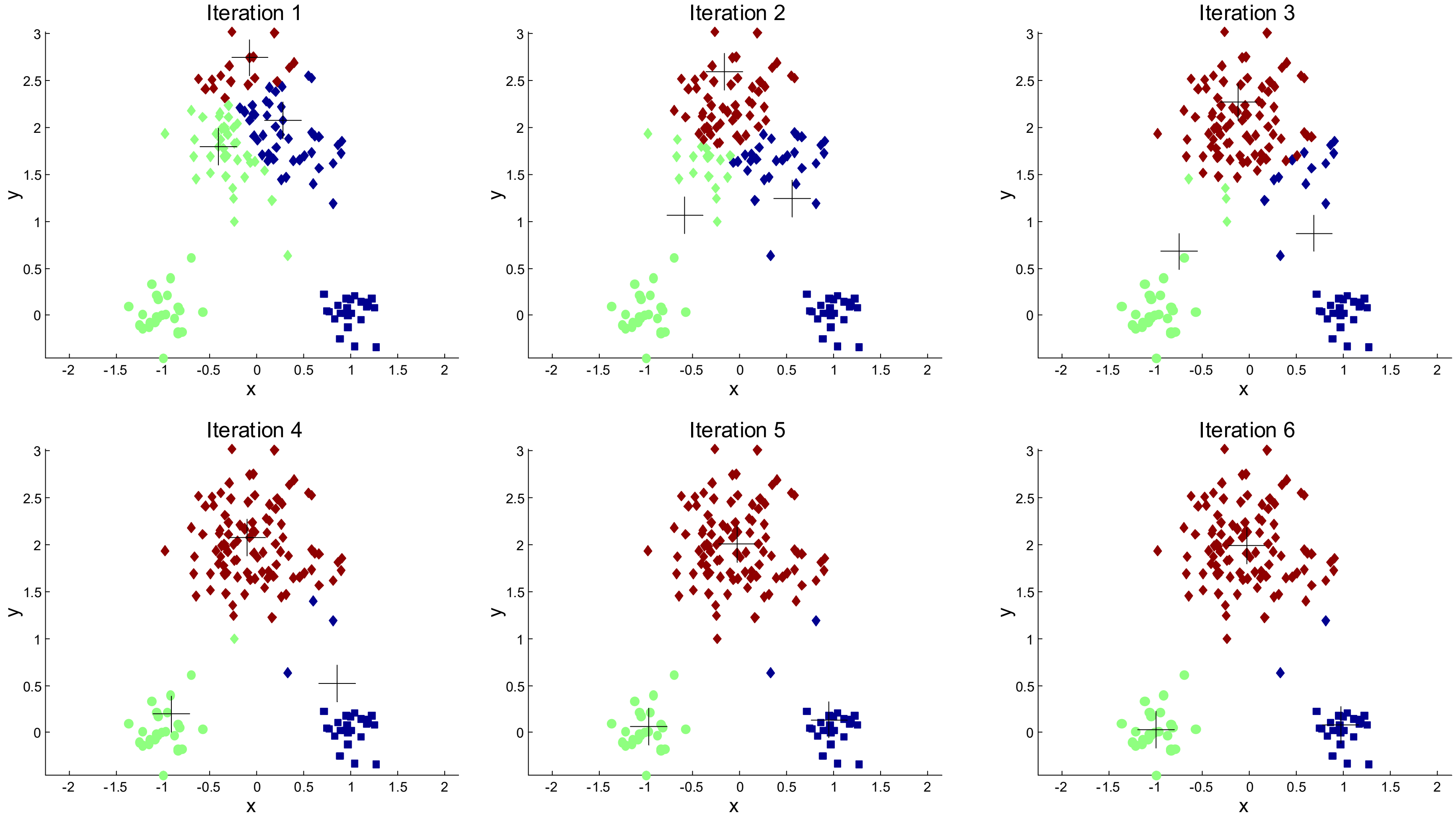
K-means Clusters

数据挖掘基础

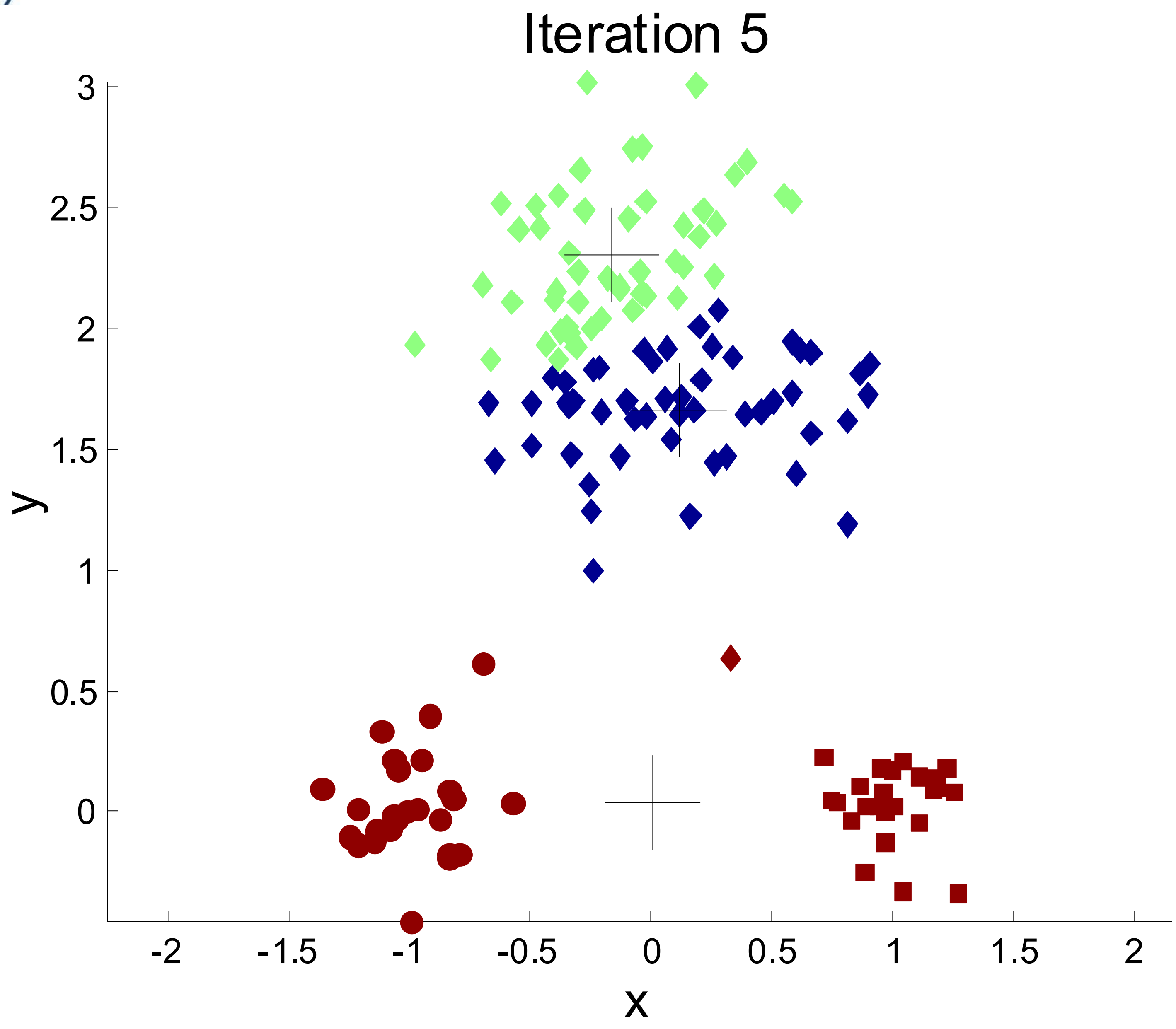
- Clustering——Importance of Choosing Initial Centroids



43 □ Clustering——Importance of Choosing Initial Centroids



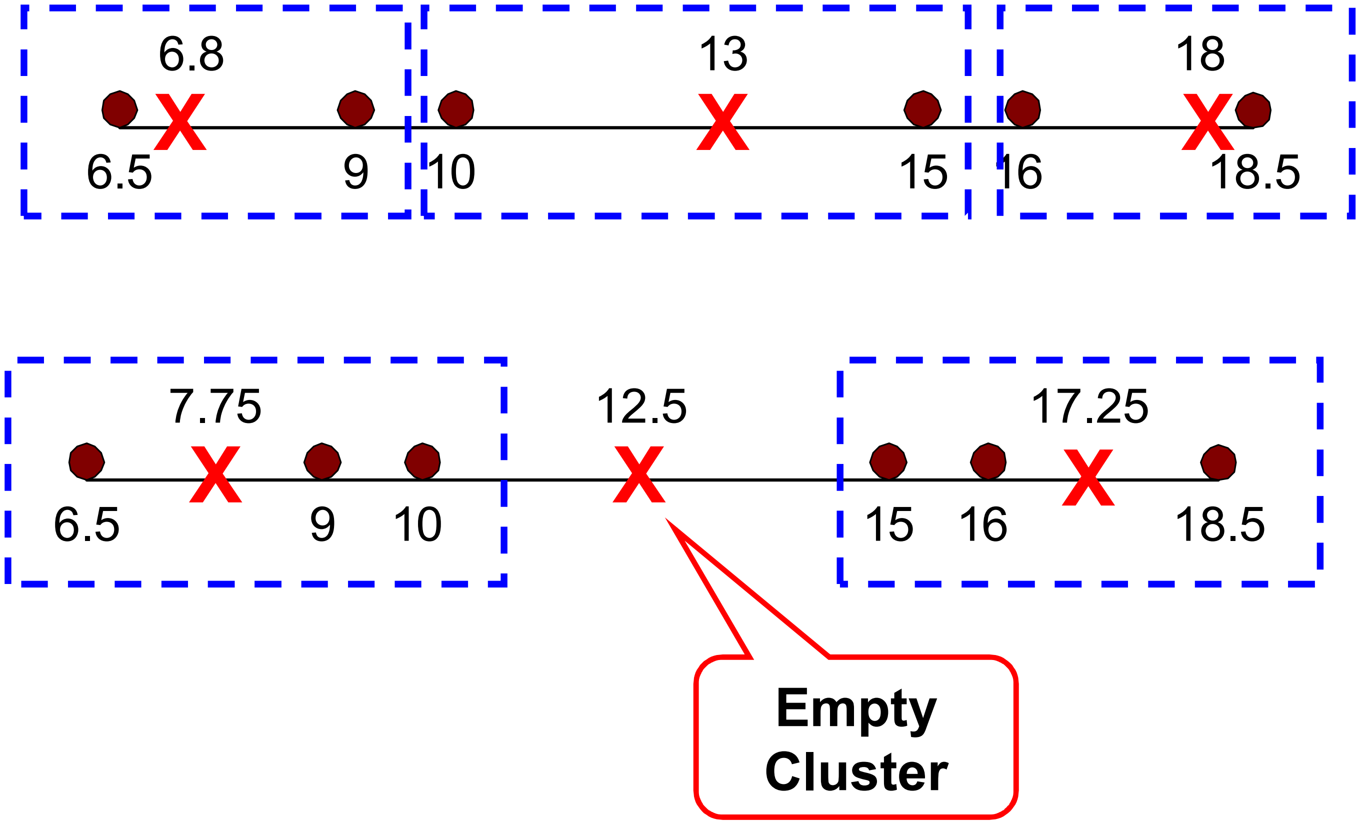
Importance of Choosing Initial Centroids …



45 Clustering——Solutions to Initial Centroids Problem

- Multiple runs
  - Helps, but probability is not on your side
- Sample and use hierarchical clustering to determine initial centroids
- Select more than k initial centroids and then select among these initial centroids
  - Select most widely separated
- Postprocessing
- Bisecting(二分) K-means
  - Not as susceptible to initialization issues

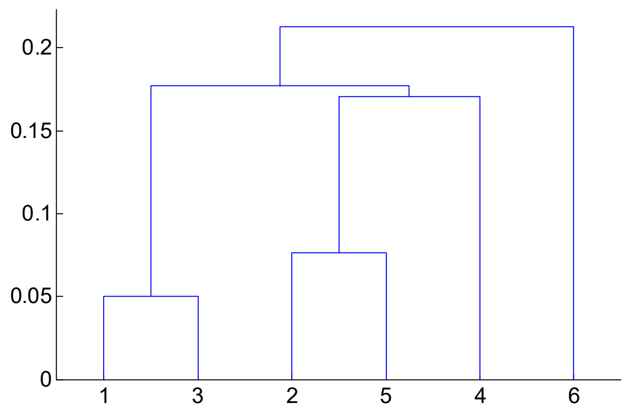
- Clustering----K-means can yield empty clusters

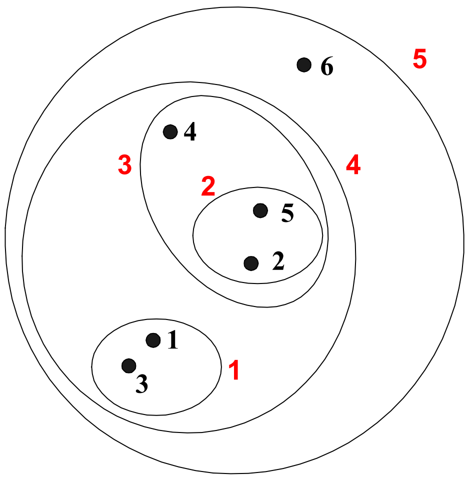


- Clustering----Basic K-means algorithm can yield empty clusters
- Several strategies(处理空簇)
  - Choose the point that contributes most to SSE
  - Choose a point from the cluster with the highest SSE
  - If there are several empty clusters, the above can be repeated several times.

数据挖掘基础

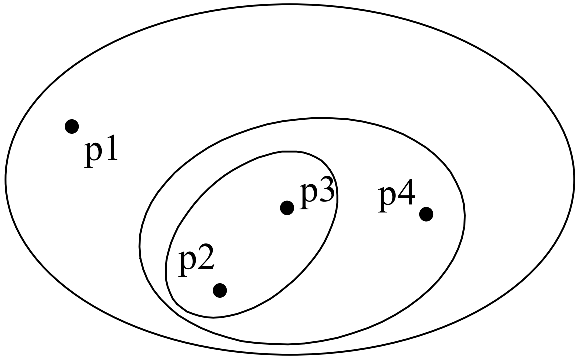
- Clustering——Hierarchical Clustering（层次聚类）
  - Produces a set of nested clusters organized as a hierarchical tree
  - Can be visualized as a dendrogram(树状图)
  - A tree like diagram that records the sequences of merges or splits





- Clustering——Strengths of Hierarchical Clustering（层次聚类）
  - Do not have to assume any particular number of clusters
  - Any desired number of clusters can be obtained by 'cutting' the dendrogram(树状图)at the proper level
  - They may correspond to meaningful taxonomies(分类标准)
  - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, …)

- Clustering——Hierarchical Clustering（层次聚类）
  - Two main types of hierarchical clustering
  - Agglomerative(凝聚的)
    - Start with the points as individual clusters
    - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left



  - Divisive(分裂的)
    - Start with one, all-inclusive cluster
    - At each step, split a cluster until each cluster contains a point (or there are k clusters)
  - Traditional hierarchical algorithms use a similarity or distance matrix
    - Merge or split one cluster at a time

- Hierarchical Clustering——Agglomerative(凝聚的)
  - More popular hierarchical clustering technique
  - Basic algorithm is straightforward

    Compute the proximity matrix

    Let each data point be a cluster

    Repeat

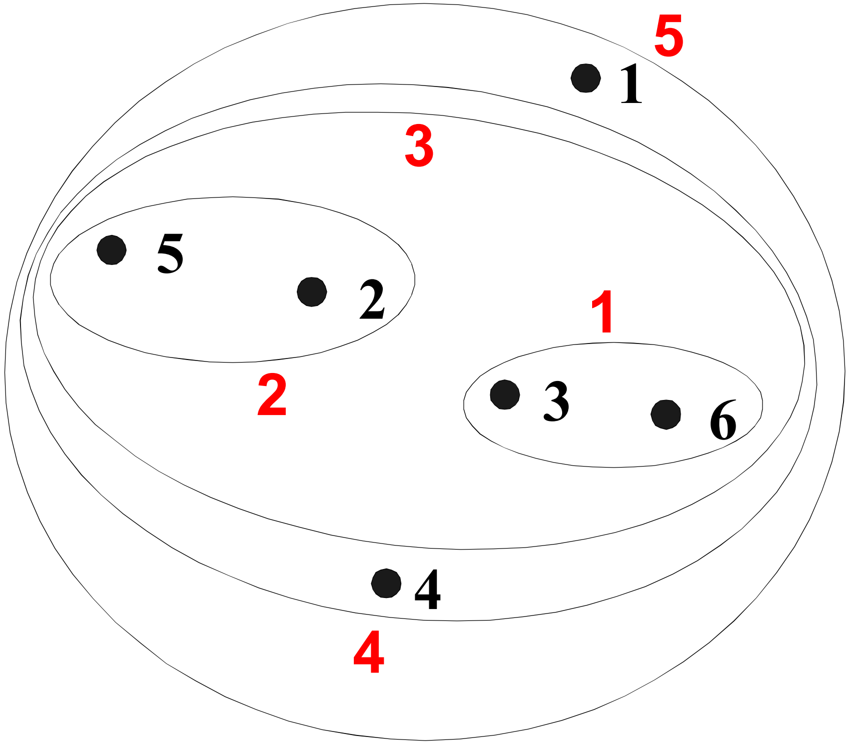
        Merge the two closest clusters

        Update the proximity matrix

    Until only a single cluster remains
  - Key operation is the computation of the proximity of two clusters
    - Different approaches to defining the distance between clusters distinguish the different algorithms

数据挖掘基础

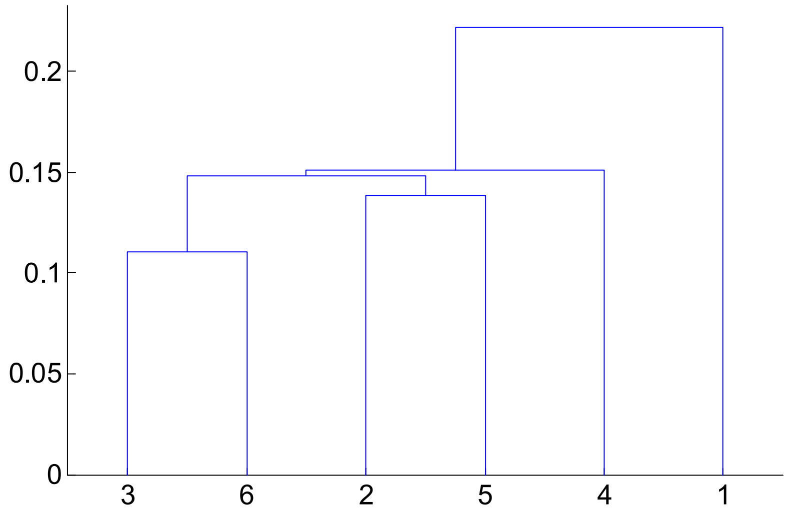
- Hierarchical Clustering——Agglomerative(凝聚的) --MIN



Nested Clusters

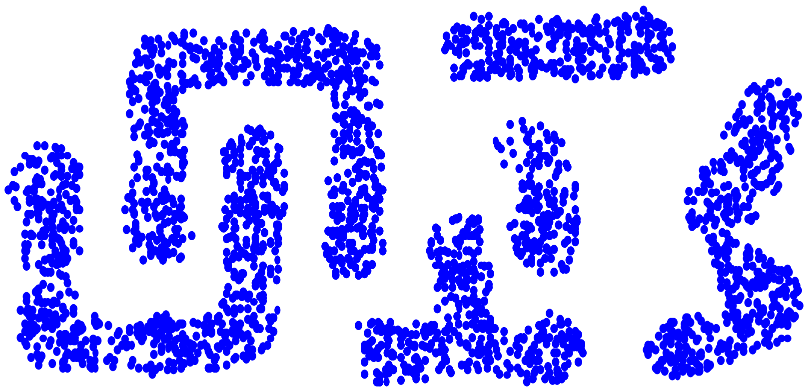
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

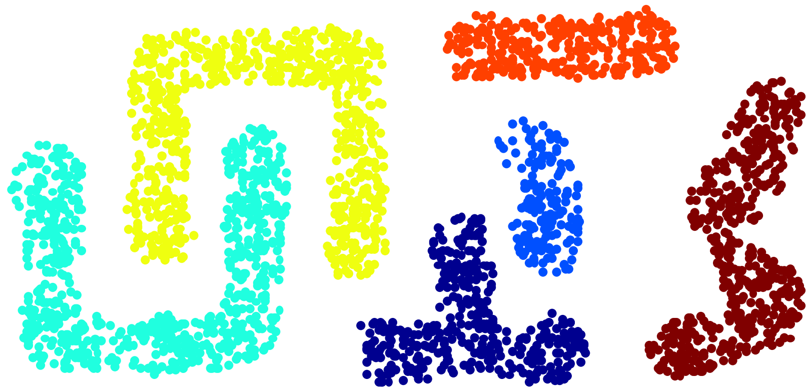


数据挖掘基础

- Hierarchical Clustering——Agglomerative(凝聚的) – Strength of MIN



Original Points

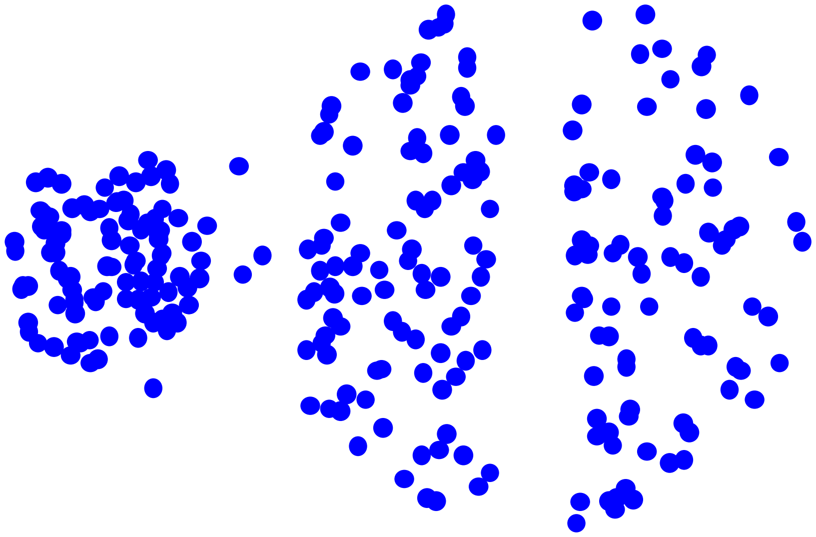


Six Clusters

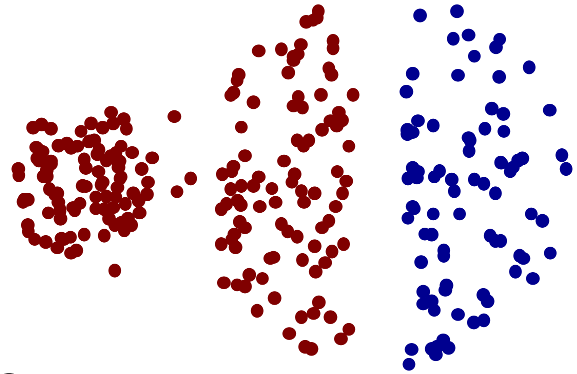
- **Can handle non-elliptical(非椭圆) shapes**

数据挖掘基础

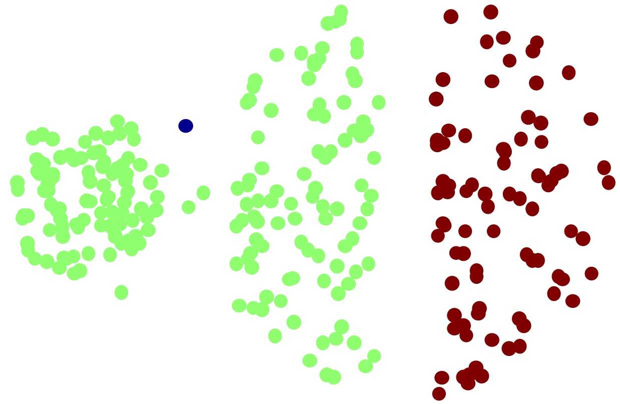
- Hierarchical Clustering——Agglomerative(凝聚的) – Limits of MIN



Original Points



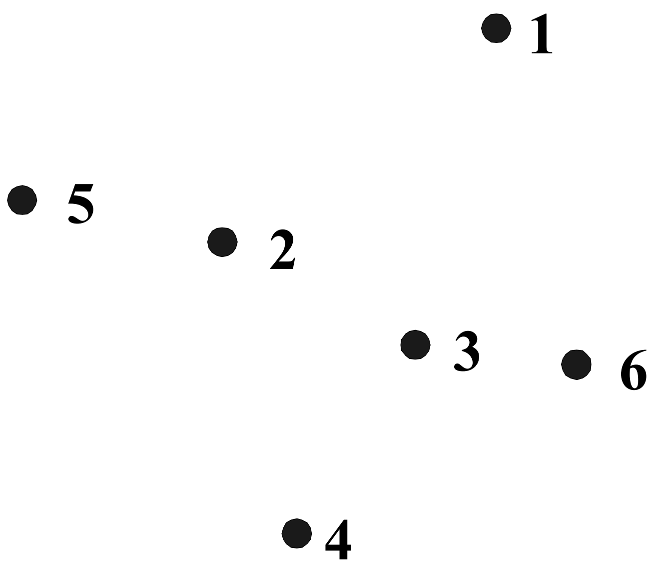
Two Clusters



Three Clusters

- **Sensitive to noise and outliers**

- Hierarchical Clustering——Agglomerative(凝聚的) – MAX ？？？



Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

- Hierarchical Clustering——Problems and Limitations

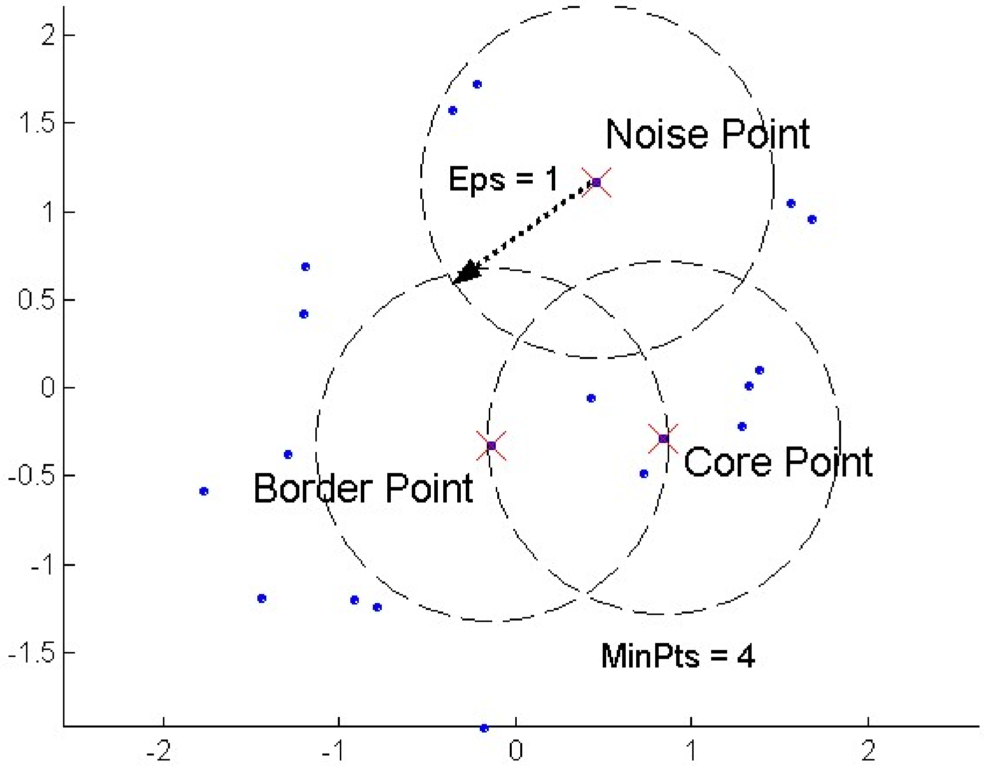
  - Once a decision is made to combine two clusters, it cannot be undone(不可撤销)

  - No objective function is directly minimized

  - Different schemes have problems with one or more of the following:
    - Sensitivity to noise and outliers
    - Difficulty handling different sized clusters and convex shapes
    - Breaking large clusters

- Clustering——Density-based Clustering（密度聚类）
  - Density-Based Spatial Clustering of Applications with Noise
  - DBSCAN is a density-based algorithm.
    - Density = number of points within a specified radius (Eps)
    - A point is a core point（核心点） if it has more than a specified number of points (MinPts) within **Eps**
    - These are points that are at the interior of a cluster
    - A border point（边界点） has fewer than MinPts within **Eps**, but is in the neighborhood of a core point
    - A noise point （噪音点）is any point that is not a core point or a border point.

- Clustering——Density-based Clustering（密度聚类）
  - Core, Border, and Noise Points



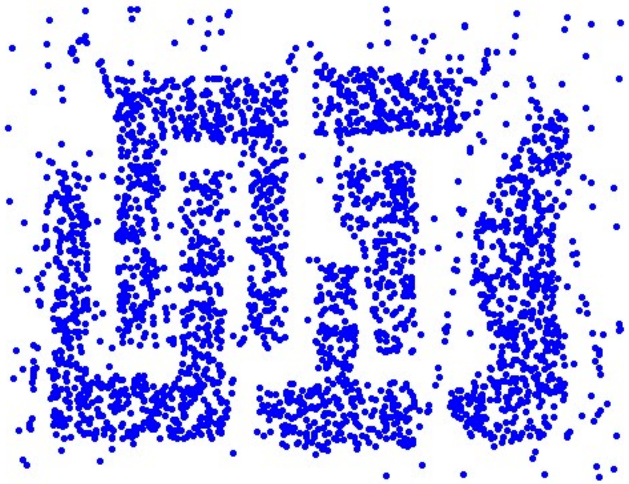
- Clustering——Density-based Clustering（密度聚类）
  - Eliminate noise points
  - Perform clustering on the remaining points

$current\_cluster\_label \leftarrow 1$
for all core points **do**
  if the core point has no cluster label **then**
    $current\_cluster\_label \leftarrow current\_cluster\_label + 1$
    Label the current core point with cluster label $current\_cluster\_label$
  end if
  for all points in the Eps-neighborhood, except i^{th} the point itself **do**
    if the point does not have a cluster label **then**
      Label the point with cluster label $current\_cluster\_label$
    end if
  end for
end for

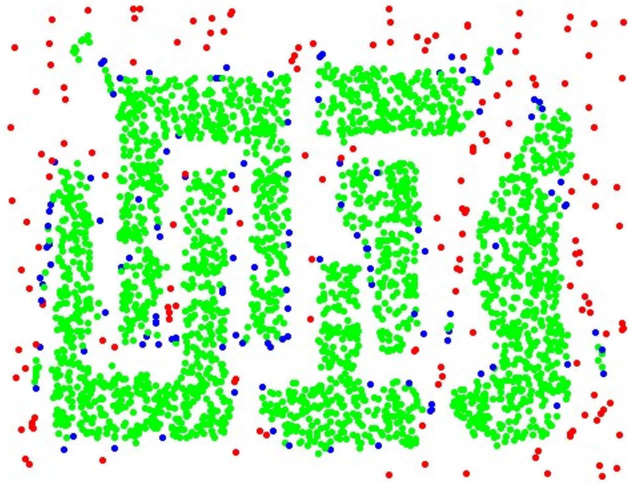
其他核结点也有可能是该核的边界点

数据挖掘基础

- Clustering——Density-based Clustering（密度聚类）
  - Eps = 10, MinPts = 4

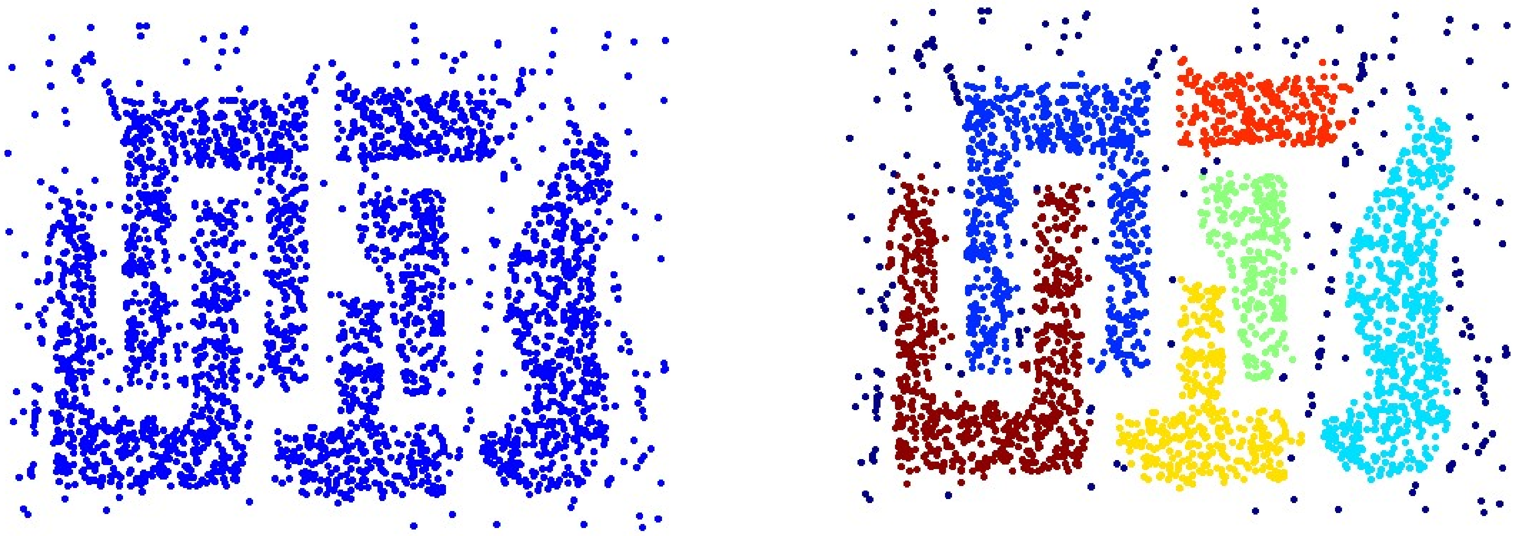


Original Points



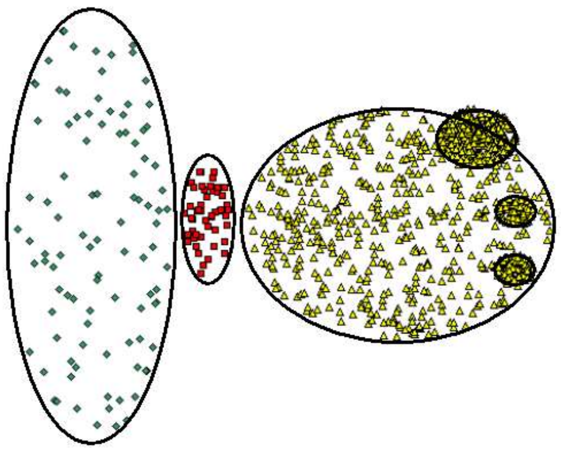
Point types: core, border and noise

- Clustering——Density-based Clustering（密度聚类）
  - When DBSCAN Works Well
    - Resistant to Noise
    - Can handle clusters of different shapes and sizes

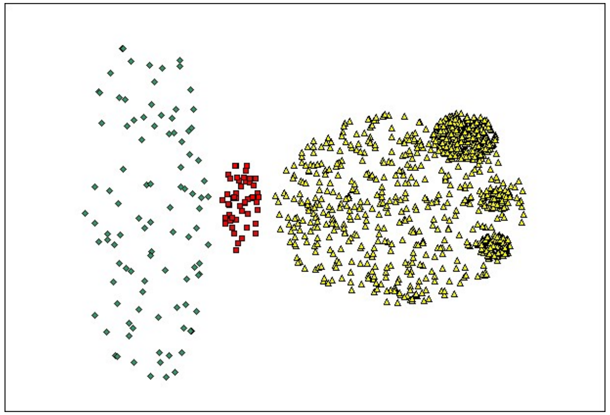


Original Points　　　　Clusters

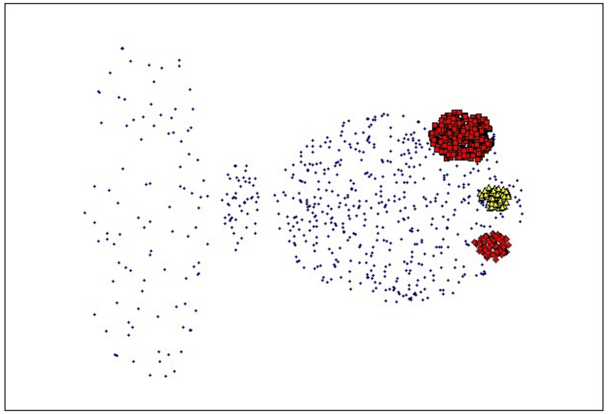
- Clustering——Density-based Clustering（密度聚类）
  - When DBSCAN Does NOT Work Well
    - Varying densities
    - High-dimensional data



Original Points



(MinPts=4, Eps=9.75).



(MinPts=4, Eps=9.92)

数据挖掘基础

2013 IEEE ICDM
Research Contributions Award
Computer Society
Professor Hans-Peter Kriegel,
ACM Fellow
Ludwig-Maximilians-Universitaet Munich,
Germany
– Number 5 for field rating in data mining by Microsoft Academic Search
– DBSCAN (KDD, 1996), 5200+ citations by Google Scholar
– 14 ICDM papers by 2012

- Clustering——Cluster Validation “***clusters are in the eye of the beholder***”!
  - Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following three types.
    - Internal Index（非监督的）: Used to measure the goodness of a clustering structure without respect to external information.
      - Sum of Squared Error (SSE)
    - External Index（有监督的）: Used to measure the extent to which cluster labels match externally supplied class labels.
      - Entropy
    - Relative Index（相对的）: Used to compare two different clusterings or clusters.
      - Often an external or internal index is used for this function, e.g., SSE or entropy
  - Sometimes these are referred to as criteria(标准) instead of indices(指标)
  - However, sometimes criterion is the general strategy and index is the numerical measure that implements the criterion.

数据挖掘基础

- Clustering——Cluster Validation
  - Clusters in more complicated figures aren't well separated
  - Internal Index: Used to measure the goodness of a clustering structure without respect to external information
    - SSE

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

    - Cluster Separation (SSB, 分离度) : Measure how distinct or well-separated a cluster is from other clusters
      - 簇质心m_i到所有数据点的总均值m的距离平方和（越大越好）
      - Where $|C_i|$ is the size of cluster I, m is the mean of all the nodes in the dataset.

$$SSB = \sum_{i} |C_i| (m - m_i)^2$$

Clustering——Cluster Validation: External Measures

Table 5.9. K-means Clustering Results for LA Document Data Set

Cluster	Entertainment	Financial	Foreign	Metro	National	Sports	Entropy	Purity
1	3	5	40	506	96	27	1.2270	0.7474
2	4	7	280	29	39	2	1.1472	0.7756
3	1	1	1	7	4	671	0.1813	0.9796
4	10	162	3	119	73	2	1.7487	0.4390
5	331	22	5	70	13	23	1.3976	0.7134
6	5	358	12	212	48	13	1.5523	0.5525
Total	354	555	341	943	273	738	1.1450	0.7203

entropy For each cluster, the class distribution of the data is calculated first, i.e., for cluster j we compute p_{ij}, the 'probability' that a member of cluster j belongs to class i as follows: $p_{ij} = m_{ij}/m_j$, where m_j is the number of values in cluster j and m_{ij} is the number of values of class i in cluster j. Then using this class distribution, the entropy of each cluster j is calculated using the standard formula $\boxed{e_j = \sum_{i=1}^{L} p_{ij} \log_2 p_{ij}}$, where the L is the number of classes. The total entropy for a set of clusters is calculated as the sum of the entropies of each cluster weighted by the size of each cluster, i.e., $e = \sum_{i=1}^{K} \frac{m_i}{m} e_j$, where m_j is the size of cluster j, K is the number of clusters, and m is the total number of data points.

purity Using the terminology derived for entropy, the purity of cluster j, is given by $purity_j = \max p_{ij}$ and the overall purity of a clustering by $purity = \sum_{i=1}^{K} \frac{m_i}{m} purity_j$.

数据挖掘基础

- Y Liu, Z Li, H Xiong, X Gao, J Wu, "**Understanding of internal clustering validation measures**". **ICDM 2010.**
- Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, Junjie Wu, Sen Wu, "**Understanding and Enhancement of Internal Clustering Validation Measures**", **IEEE Transactions on Cybernetics (TC)**, Vol. 43, No. 3, pp. 982-994, 2013.
- J Wu, H Xiong, J Chen, "**Adapting the right measures for k-means clustering**". **KDD 2009.**

"The validation of clustering structures is the most difficult and frustrating part of cluster analysis. Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage."

----------------------*Algorithms for Clustering Data*, Jain and Dubes

- Clustering——Advanced Concepts and Algorithms
  - Prototype-based(基于原型的聚类)
    - Fuzzy K-means
    - Mixture Model Clustering
    - Self-Organizing Maps
  - Density-based(基于密度的聚类)
    - Grid-based clustering
    - Subspace clustering
  - Graph-based (基于图的聚类)
    - Chameleon
    - Jarvis-Patrick
    - Shared Nearest Neighbor (SNN)
  - Characteristics of Clustering Algorithms