数据分析及实践

Analysis and Practice of the Data

第四章 数据挖掘基础

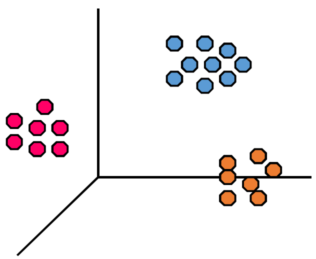
刘 淇

Email: qiliuql@ustc.edu.cn

数据挖掘基础

- 常用方法——关于四个任务有哪些常用方法?

Clustering



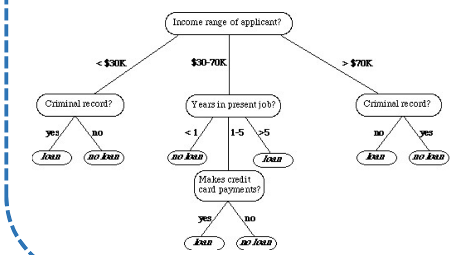
Data

	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

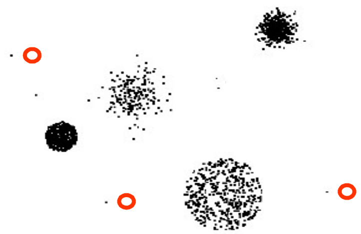
2010 monthly weather variation, Cambridge (UK)

Classification



Association Analysis

Anomaly Detection



分类——贝叶斯分类器

- A probabilistic framework for solving classification problems （概率模型来处理分类问题）

$$P(C \mid A) = \frac{P(A, C)}{P(A)}$$

- Conditional Probability : $$P(A \mid C) = \frac{P(A, C)}{P(C)}$$

回想参数估计：

- Bayes theorem: $$P(C \mid A) = \frac{P(A \mid C) P(C)}{P(A)}$$

$$p(\theta|X) = \frac{p(X|\theta) \cdot p(\theta)}{p(X)}$$

$$posterior = \frac{likelihood \cdot prior}{evidence}$$

数据挖掘基础

分类——贝叶斯分类器

- Consider each attribute and class label as random variables （随机变量）

- Given a record with attributes (A1, A2,…,An)
  - Goal is to predict class C （目标是预测C）
  - Specifically, we want to find the value of C that maximizes P(C| A1, A2,…,An) （最大化这个值）

- Can we estimate P(C| A1, A2,…,An) directly from data?

	A1	A2	A3	C
Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No

分类——贝叶斯分类器

- Approach:
  - compute the posterior probability （后验概率） P(C | A1, A2, …, An) for all values of C using the Bayes theorem

$$\boxed{P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C)\, P(C)}{P(A_1 A_2 \ldots A_n)}}$$

  - Choose value of C that maximizes
    P(C | A1, A2, …, An)
  - Equivalent to choosing value of C that maximizes
    P(A1, A2, …, An|C) P(C)
- How to estimate P(A1, A2, …, An | C)?

数据挖掘基础

分类——贝叶斯分类器

- Assume independence（独立性） among attributes Ai when class is given(Naïve):
  - P(A1, A2, …, An |C) = P(A1| Cj) P(A2| Cj)… P(An| Cj)
  - Can estimate P(Ai| Cj) for all Ai and Cj.
  - New point is classified to Cj if P(Cj) Π P(Ai| Cj) is maximal.

$$\boxed{P(C \mid A_1 A_2 \ldots A_n) = P(A_1 \mid C)\; P(A_2 \mid C) \ldots P(A_n \mid C)\; P(C)}$$

数据挖掘基础

分类——贝叶斯分类器

- How to Estimate Probabilities from Data?

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Class: **$P(C) = N_c/N$**
  - e.g., $P(No) = 7/10$,
    $P(Yes) = 3/10$

- **For discrete attributes**:

  $P(A_i \mid C_k) = |A_{ik}| / N_c$

  - where $|A_{ik}|$ is number of k instances having attribute A_i and belongs to class C_k
  - Examples:

    P(Status=Married|No) = 4/7
    P(Refund=Yes|Yes)=0

- 分类——贝叶斯分类器
  - How to Estimate Probabilities from Data?
  - **For continuous attributes （对于连续属性）：**
    - Discretize （离散化）the range into bins
      - one ordinal attribute per bin
      - violates independence assumption
    - Two-way split（二路分裂）：(A < v) or (A > v)
      - choose only one of the two splits as new attribute
    - Probability density estimation（概率密度估计）：
      - Assume attribute follows a normal distribution
      - Use data to estimate parameters of distribution (e.g., mean and standard deviation)
      - Once probability distribution is known, can use it to estimate the conditional probability P(Ai|c)

数据挖掘基础

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- 分类——贝叶斯分类器
  - Normal distribution:

$$P(A_i \mid c_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(A_i-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

    - One for each (A_i, c_i) pair
  - For (Income, Class=No):
    - If Class=No
      - sample mean = 110
      - sample variance = 2975

$$P(Income = 120 \mid No) = \frac{1}{\sqrt{2\pi}(54.54)} e^{-\frac{(120-110)^2}{2(2975)}} = 0.0072$$

For example: Given a Test Record:

$$X = (\text{Refund} = \text{No}, \text{Married}, \text{Income} = 120\text{K})$$

naive Bayes Classifier:

P(Refund=Yes|No) = 3/7
P(Refund=No|No) = 4/7
P(Refund=Yes|Yes) = 0
P(Refund=No|Yes) = 1
P(Marital Status=Single|No) = 2/7
P(Marital Status=Divorced|No)=1/7
P(Marital Status=Married|No) = 4/7
P(Marital Status=Single|Yes) = 2/7
P(Marital Status=Divorced|Yes)=1/7
P(Marital Status=Married|Yes) = 0

For taxable income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

- P(X|Class=No) = P(Refund=No|Class=No)
  × P(Married| Class=No)
  × P(Income=120K| Class=No)
  = 4/7 × 4/7 × 0.0072 = 0.0024

- P(X|Class=Yes) = P(Refund=No| Class=Yes)
  × P(Married| Class=Yes)
  × P(Income=120K| Class=Yes)
  = 1 × 0 × 1.2 × 10^{-9} = 0

Since P(X|No)P(No) > P(X|Yes)P(Yes)

Therefore P(No|X) > P(Yes|X)
=> Class = No

For example:

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

A: attributes

M: mammals

N: non-mammals

1.P(M|A)

=P(M,A)/ ~~P(A)~~

=P(A|M)P(M)

=**P(Give Birth=Yes, Can Fly=no, Live in Water= Yes, Have Legs= No| M) P(M)**

= P(Give Birth=Yes|M) P(Can Fly=no |M) P(Live in Water= Yes |M) P(Have Legs= No |M) P(M)

2. P(N|A)=P(N,A)/P(A)

=P(A,N)=P(A|N)P(N)

For example:

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

A: attributes

M: mammals

N: non-mammals

$$P(A|M)=\frac{6}{7}\times\frac{6}{7}\times\frac{2}{7}\times\frac{2}{7}=0.06$$

$$P(A|N)=\frac{1}{13}\times\frac{10}{13}\times\frac{3}{13}\times\frac{4}{13}=0.0042$$

$$P(A|M)P(M)=0.06\times\frac{7}{20}=0.021$$

$$P(A|N)P(N)=0.004\times\frac{13}{20}=0.0027$$

P(A|M)P(M) > P(A|N)P(N)

=> Mammals

数据挖掘基础-练习

给定以下7个用户的数据，使用朴素贝叶斯方法预测用户8={有工作=否，婚姻状况=已婚}的拖欠贷款属性最有可能是Yes还是No，并给出求解过程。

用户ID	有工作	婚姻状况	拖欠贷款
1	否	已婚	No
2	是	单身	Yes
3	否	单身	No
4	是	已婚	Yes
5	否	单身	No
6	是	单身	Yes
7	否	已婚	No
8	否	已婚	？

数据挖掘基础

如果有一个属性值在训练样本中不存在，这时候算出的所有类别的后验概率都是0，导致无法准确分类（仅使用数据记录的比例来估计类条件概率的方法显得太脆弱了），尤其是当训练数据很少而属性数目又很大时。

一般可采用M估计（M-Estimate）来平滑类条件概率的计算，从而得到非0的可比较的近似概率值，达到分类的目的。

$$P(x_i|y_j)=\frac{n_c+mp}{n+m}$$

其中，n 是类 y_j 中的实例总数，n_c 是类 y_j 的训练样例中取值 x_i 的样例数，m 是称为等价样本大小的参数，而 p 是用户指定的参数。如果没有训练集（即 $n=0$），则 $P(x_i|y_j)=p$。因此 p 可以看作是在类 y_j 的记录中观察属性值 x_i 的先验概率。等价样本大小决定先验概率 p 和观测概率 n_c/n 之间的平衡。

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

例题：使用朴素贝叶斯m估计，对该测试样本*X*进行分类，设m=3, 且对类Yes的所有属性p=1/3，对类No的所有属性p=2/3

$$X = (\text{Refund} = \text{No}, \text{Married}, \text{Income} = 120\text{K})$$

$$P(\mathbf{X}|\text{No}) = P(\text{有房=否}|\text{No}) \times P(\text{婚姻状况=已婚}|\text{No}) \times P(\text{年收入=\$120K}|\text{No})$$
$$= 6/10 \times 6/10 \times 0.0072 = 0.0026$$

$$P(\mathbf{X}|\text{Yes}) = P(\text{有房=否}|\text{Yes}) \times P(\text{婚姻状况=已婚}|\text{Yes}) \times P(\text{年收入=\$120K}|\text{Yes})$$
$$= 4/6 \times 1/6 \times 1.2 \times 10^{-9} = 1.3 \times 10^{-10}$$

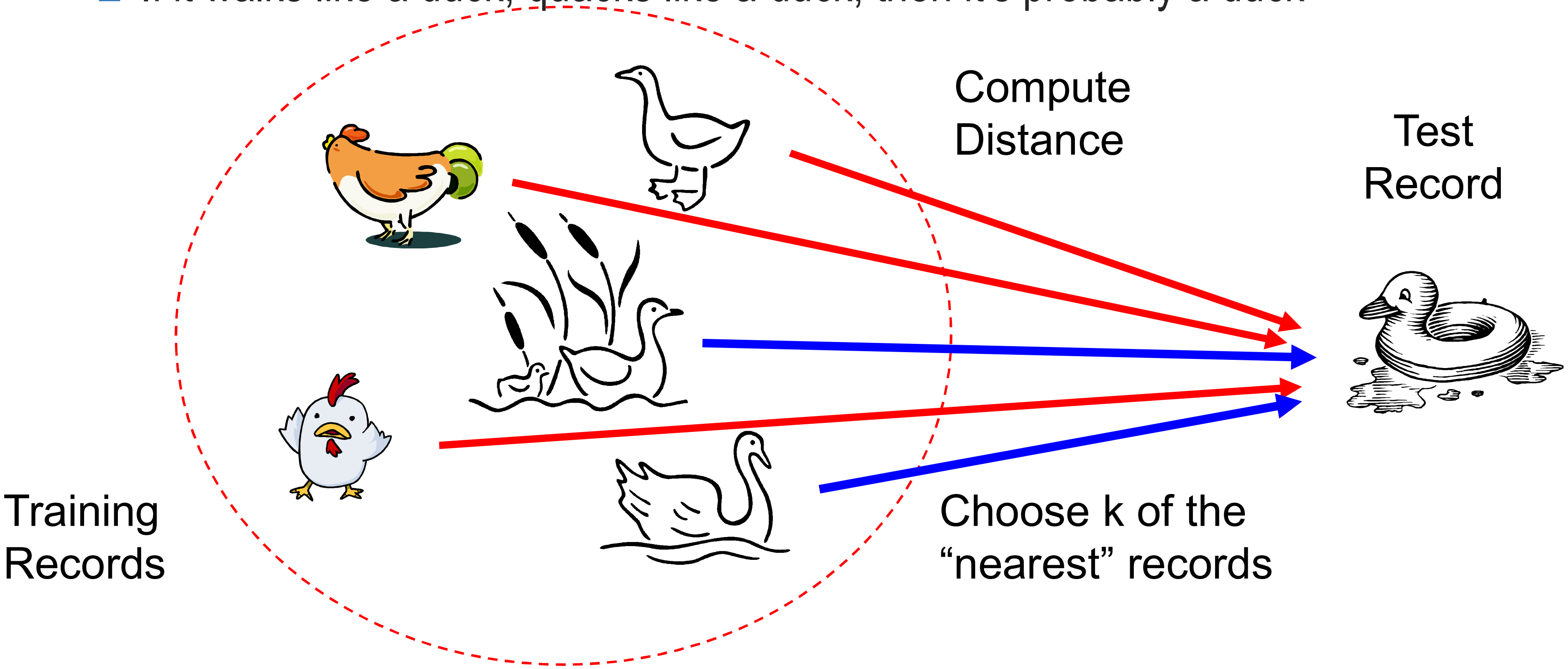
P(X|No)P(No)=0.0026*0.7=0.00182
P(X|Yes)P(Yes)=1.3*10^{-10}*0.3=3.9*10^{-11}

Since P(X|No)P(No) > P(X|Yes)P(Yes)

Therefore P(No|X) > P(Yes|X) => Class = No

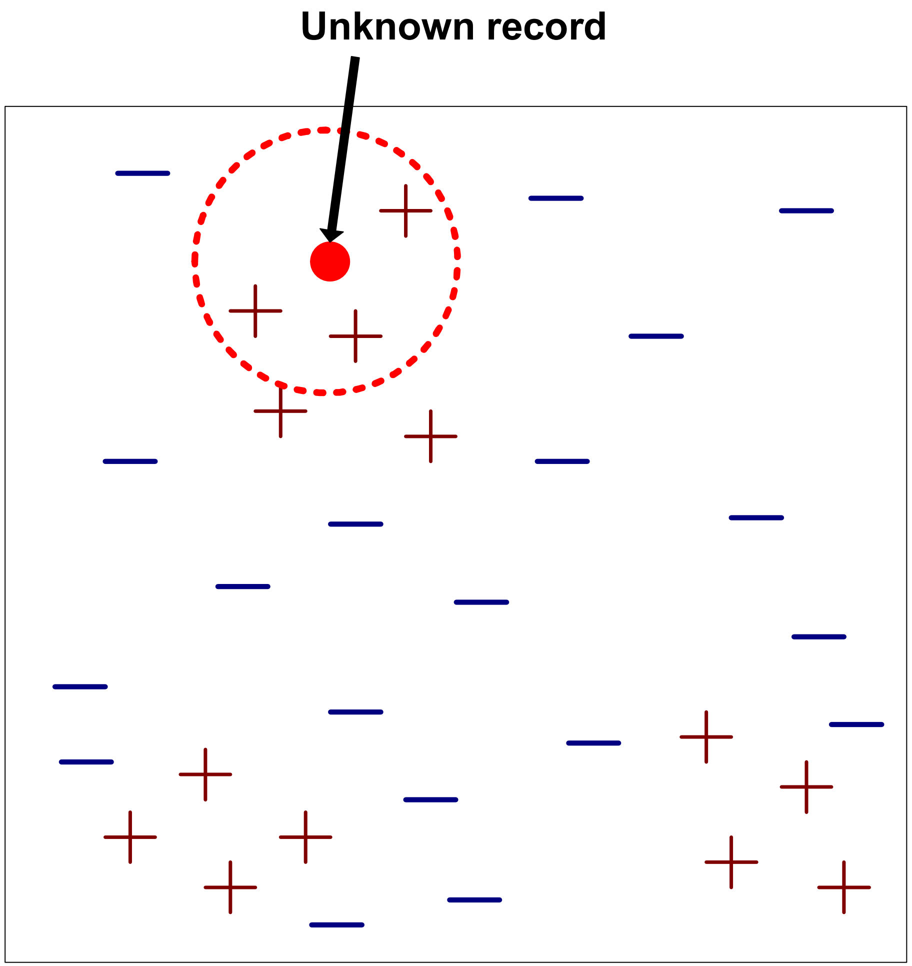
数据挖掘基础

- 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



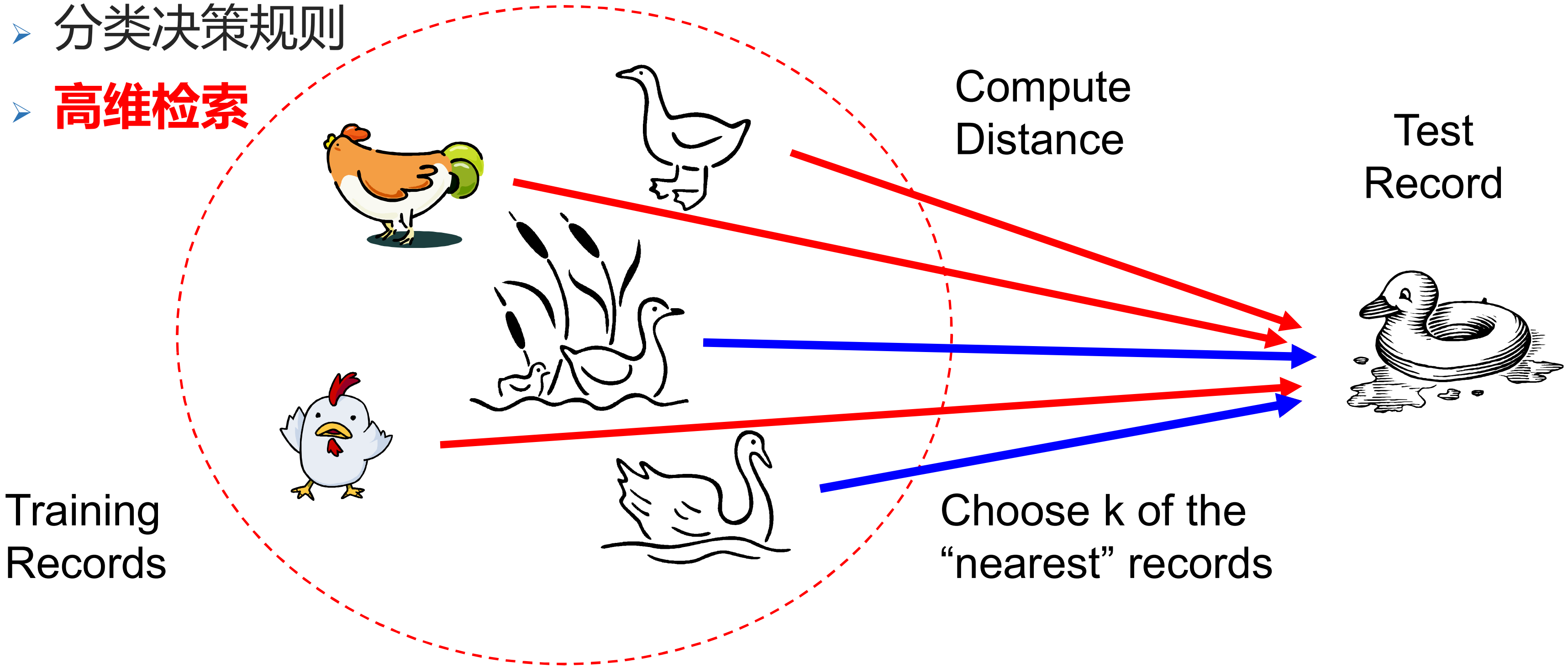
- Requires three things
  - The set of stored records
  - Distance Metric （距离矩阵） to compute distance between records
  - The value of *k*, the number of nearest neighbors to retrieve

- 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)

数据挖掘基础

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：

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

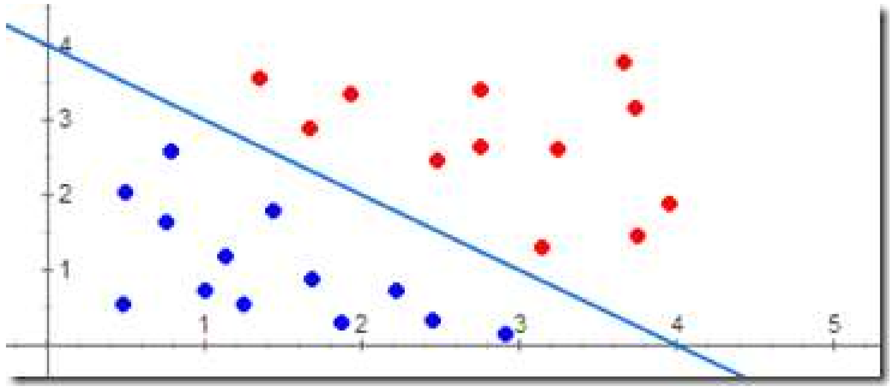
称为感知机，w和b为感知机参数，w为权值（weight），b为偏置（bias）。

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



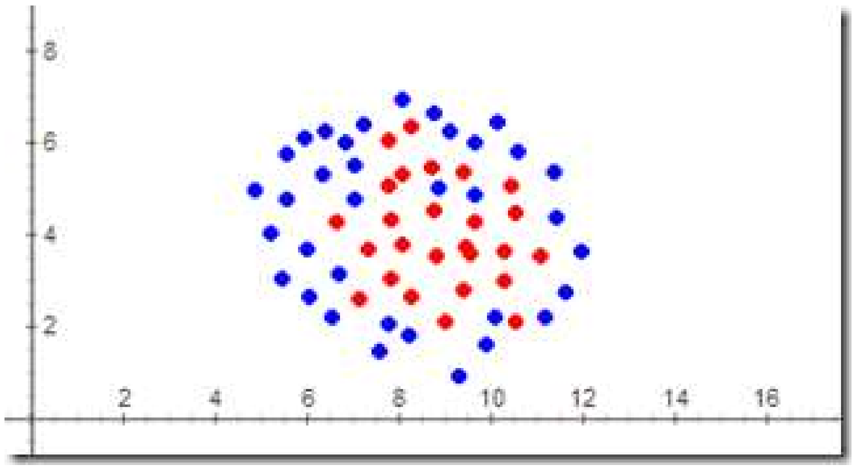
目标函数:

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

其中M是错分类的数据集合

数据挖掘基础

- 感知机学习策略
  - 数据集线性可分性
    - 在二维平面中，可以用一条直线将+1类和-1类完美分开，那么这个样本空间就是线性可分的。下图中的样本就是线性不可分的，感知机就不能处理这种情况。因此，感知机都基于一个前提：问题空间线性可分
  - 定义损失函数，找到参数w和b，使得损失函数最小



数据挖掘基础

- 损失函数的选取
  - 损失函数的一个自然选择就是误分类点的总数，但是这样的点不是参数w，b的连续可导函数，不易优化
  - 损失函数的另一个选择就是误分类点到划分超平面S(**w.x+b=0**)的总距离

假设数据集 $T=\{(x_1,y_1),(x_2,y_2)...(x_n,y_n)\}$ 中所有的 $y_i=+1$ 的实例i，有 $w \bullet x+b>0$ ；对 $y_i=-1$ 的实例有 $w \bullet x+b<0$
这里 先给出输入空间 R^n 中任意一点 x_0 到超平面S的距离：

$$\frac{1}{\|w\|}|w \bullet x_0+b|$$

这里 $\frac{1}{\|w\|}$ 是W的 l_2 范数。所以，对于误分类数据（x_i，y_i）有

$$-y_i(w \bullet x_i+b)>0$$

因为对x_i错分了，所以若y_i为-1，则计算的(w.x_i+b)>0，反之若y_i为+1，则计算的(w.x_i+b)<0

数据挖掘基础

- 点x_0到超平面S: **w.x**+b=0(注: x_0,**w,x**全为*N*维向量)距离*d*的计算过程为:
  - 设点 x_0在平面S上的投影为x_1,则**w**.x_1+b=0
  - 由于向量$\overrightarrow{x_0x_1}$与S平面的法向量w平行，所以（乘积的模=模的乘积）

$$|w \bullet \overrightarrow{x_0x_1}| = \|w\| \, |\overrightarrow{x_0x_1}| = \sqrt{(w^1)^2+\ldots+(w^N)^2}\, d = \|w\| d$$

L_2范数

$$\text{又} \quad w \bullet \overrightarrow{x_0x_1} = w^1(x^1{}_0 - x_1{}^1) + w^2(x^2{}_0 - x_1{}^2) + \ldots. + w^N(x^N{}_0 - x_1{}^N)$$
$$= w^1x^1{}_0 + w^2x^2{}_0 + \ldots + w^Nx^N{}_0 - (w^1x^1{}_1 + w^2x^2{}_1 + \ldots + w^Nx^N{}_1)$$
$$= w^1x^1{}_0 + w^2x^2{}_0 + \ldots + w^Nx^N{}_0 - (-b)$$

x_1在平面S上，所以 **w**.x_1+b=0

$$\text{所以} \quad \|w\| d = |w^1x^1{}_0 + w^2x^2{}_0 + \ldots + w^Nx^N{}_0 + b| = |w \bullet x_0 + b|$$

$$\text{即} \quad d = \frac{1}{\|w\|}|w \bullet x_0 + b|$$

数据挖掘基础

因此误分类点(x_i, y_i)到超平面S的距离可以写作：

$$-\frac{1}{\|w\|}y_i(w\bullet x_i+b)$$

假设误分类点的集合为M，那么所有误分类点到超平面S的总距离为：

$$-\frac{1}{\|w\|}\sum_{x_i\in M}y_i(w\bullet x_i+b)$$

这里的||w||值是固定的，不必考虑，这样就得到了感知机学习的损失函数：在误分类时是参数w，b的线性函数。也就是说，为求得正确的参数w，b，我们的目标函数为

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

而它是连续可导的，这就使得我们比较容易求得其最小值

数据挖掘基础

- 感知机学习算法的原始形式

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

  - 所谓原始形式，就是我们用梯度下降的方法，对参数w和b进行不断的迭代更新。先任意选取一个超平面 S_0 ，对应的参数分别为 w_0 和 b_0 ，当然现在是可以任意赋值的，比如说选取w_0为全为0的向量, b_0的值为0。然后用梯度下降不断地极小化损失函数：每次随机选取一个误分类点对w和b进行更新。设误分类点集合M是固定的，那么损失函数 L(w, b)的梯度为：

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

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

  - 接下来随机选取一个误分类点(x_i, y_i)对w,b进行更新

$$w \leftarrow w + \eta y_i x_i$$

$$b \leftarrow b + \eta y_i$$

其中 $\eta(0<\eta\leq 1)$ 为步长，也称为学习速率（learnin rate），一般在0到1之间取值，通过迭代，直到损失函数为0

数据挖掘基础

算法 1（感知机学习算法的原始形式）

输入：训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$，其中 $x_i\in\mathcal{X}=\mathbf{R}^n$，$y_i\in\mathcal{Y}=\{-1,+1\}$，$i=1,2,\cdots,N$；学习率 $\eta\,(0<\eta\leqslant 1)$；

输出：w,b；感知机模型 $f(x)=\mathrm{sign}(w\cdot x+b)$.

（1）选取初值 w_0,b_0

（2）在训练集中选取数据 (x_i,y_i)

（3）如果 $y_i(w\cdot x_i+b)\leqslant 0$

$$w\leftarrow w+\eta y_i x_i$$

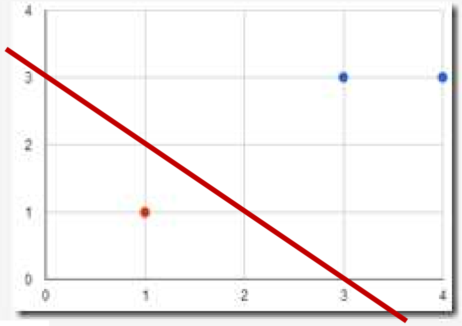
$$b\leftarrow b+\eta y_i$$

（4）转至（2），直至训练集中没有误分类点.

该算法直观上有如下解释：当一个样本被误分类时，就调整w，b使超平面S向误分类点的一侧移动，以减少误分类点到超平面的距离，直至超平面越过改点使之正确分类。

数据挖掘基础

例 如图3所示的训练数据集，其正实例点是$x_1=(3,3)^T$，$x_2=(4,3)^T$，负实例点是$x_3=(1,1)^T$，试用感知机学习算法的原始形式求感知机模型$f(x)=sign(w\cdot x+b)$，即求出w和b。这里$w=(w^{(1)},w^{(2)})^T$，$x=(x^{(1)},x^{(2)})^T$



假设学习速率为1，则每步更新为：

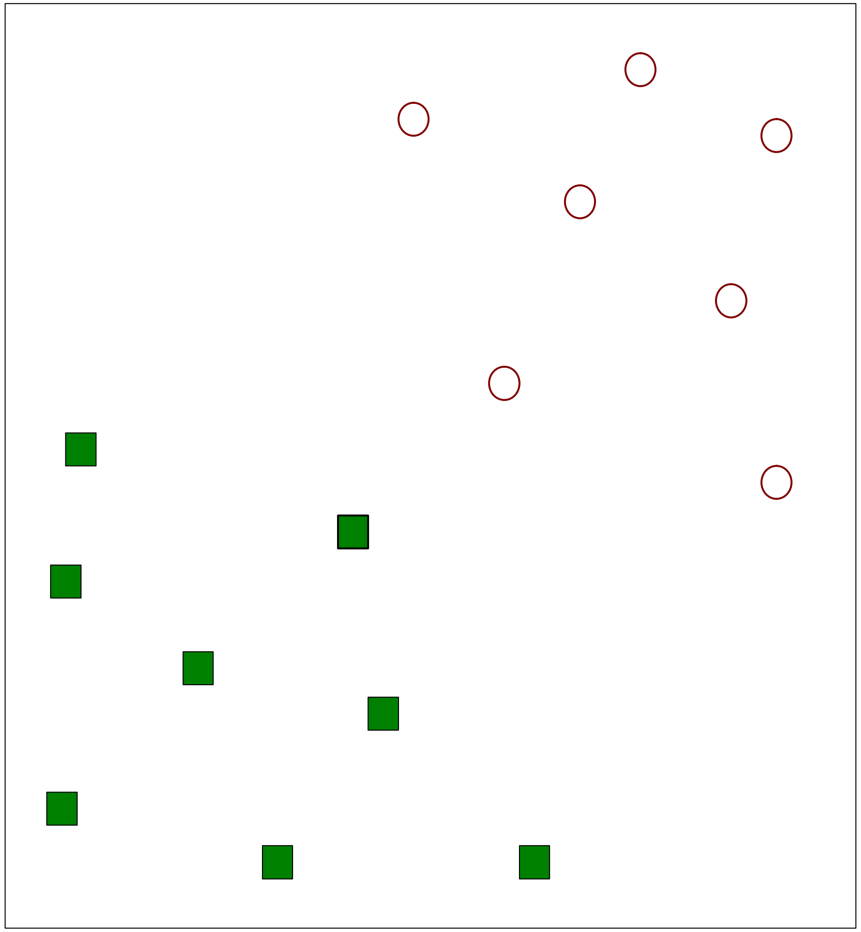
$w=w+y_ix_i$

$b=b+y_i$

$y_i(w\cdot x_i+b)\leqslant 0$

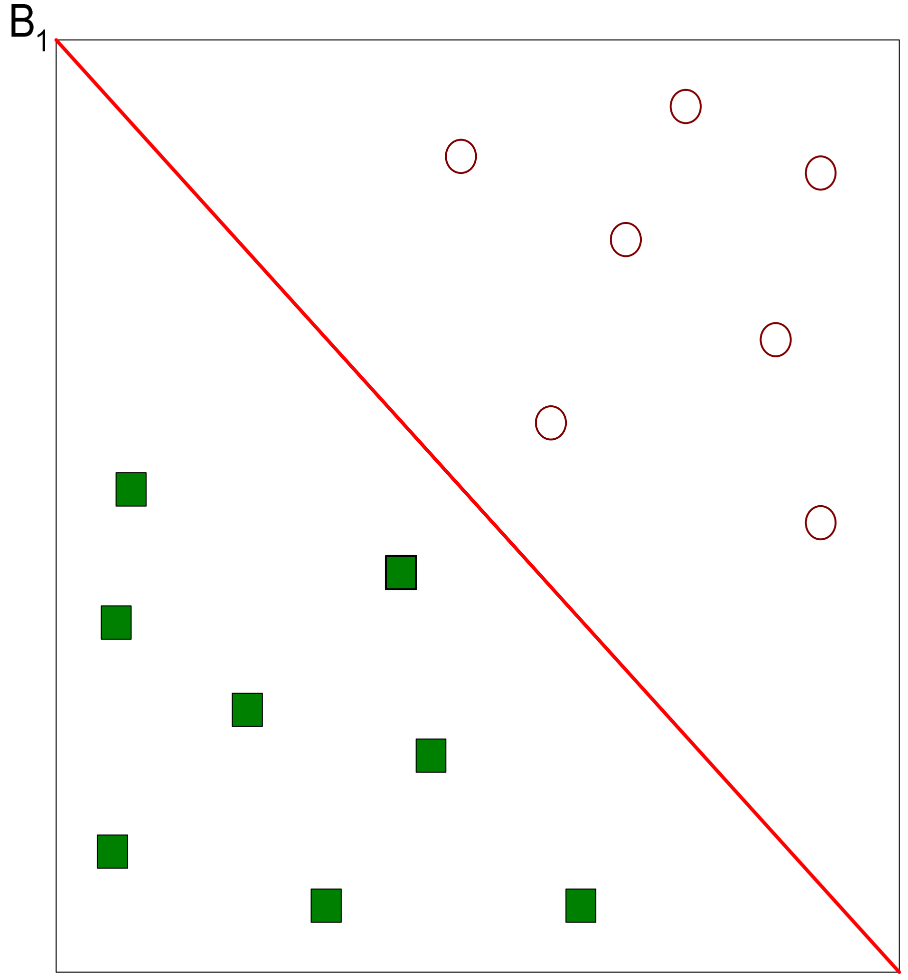
迭代次数	误分类点	w	b	$w\cdot x+b$
0		0	0	0
1	x_1	$(3,3)^T$	1	$3x^{(1)}+3x^{(2)}+1$
2	x_3	$(2,2)^T$	0	$2x^{(1)}+2x^{(2)}$
3	x_3	$(1,1)^T$	−1	$x^{(1)}+x^{(2)}-1$
4	x_3	$(0,0)^T$	−2	−2
5	x_1	$(3,3)^T$	−1	$3x^{(1)}+3x^{(2)}-1$
6	x_3	$(2,2)^T$	−2	$2x^{(1)}+2x^{(2)}-2$
7	x_3	$(1,1)^T$	−3	$x^{(1)}+x^{(2)}-3$
8	0	$(1,1)^T$	−3	$x^{(1)}+x^{(2)}-3$

- 分类——感知机（**perceptron**）



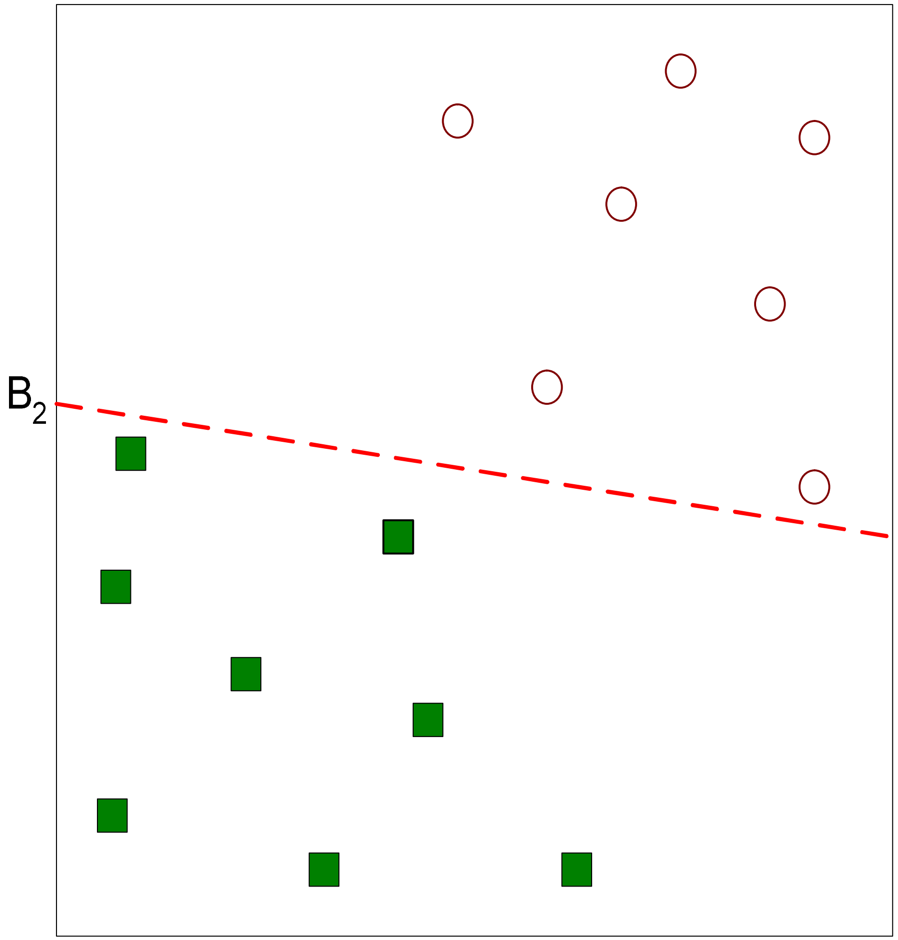
数据挖掘基础

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



一个可行解

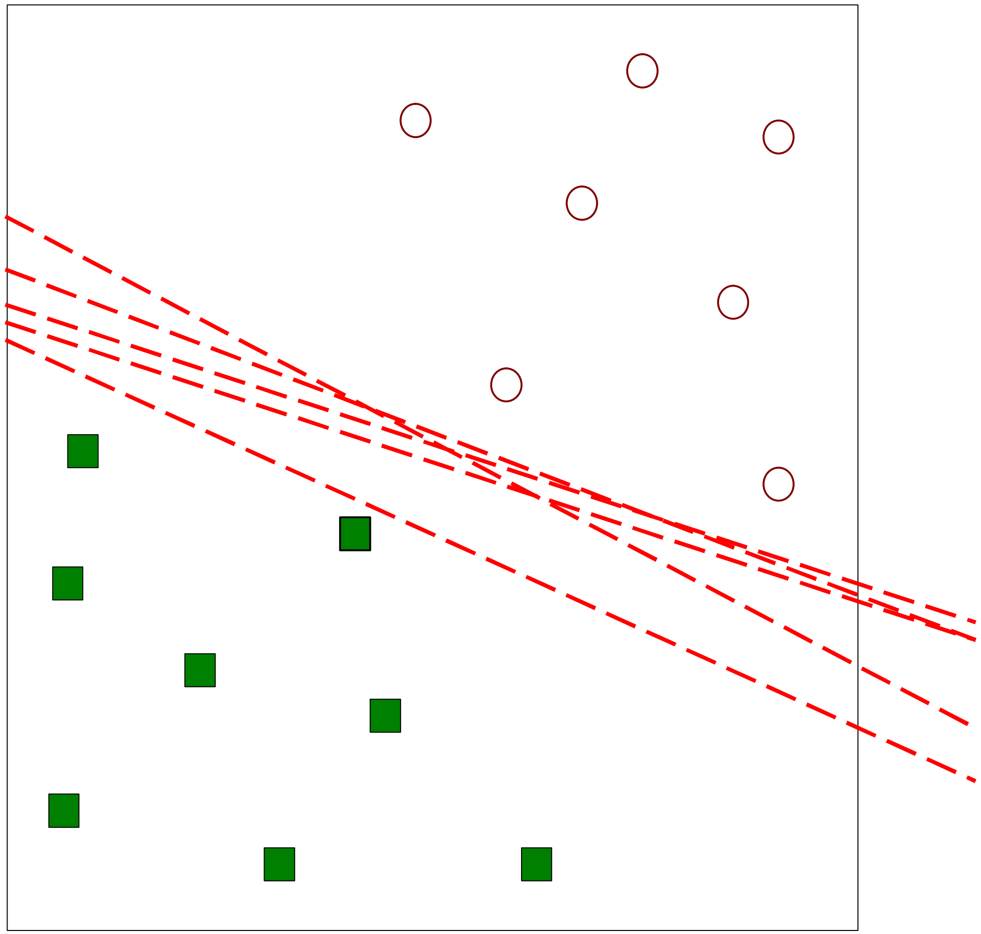
- 分类——支持向量机（**Support Vector Machine**）



另一个可行解

数据挖掘基础

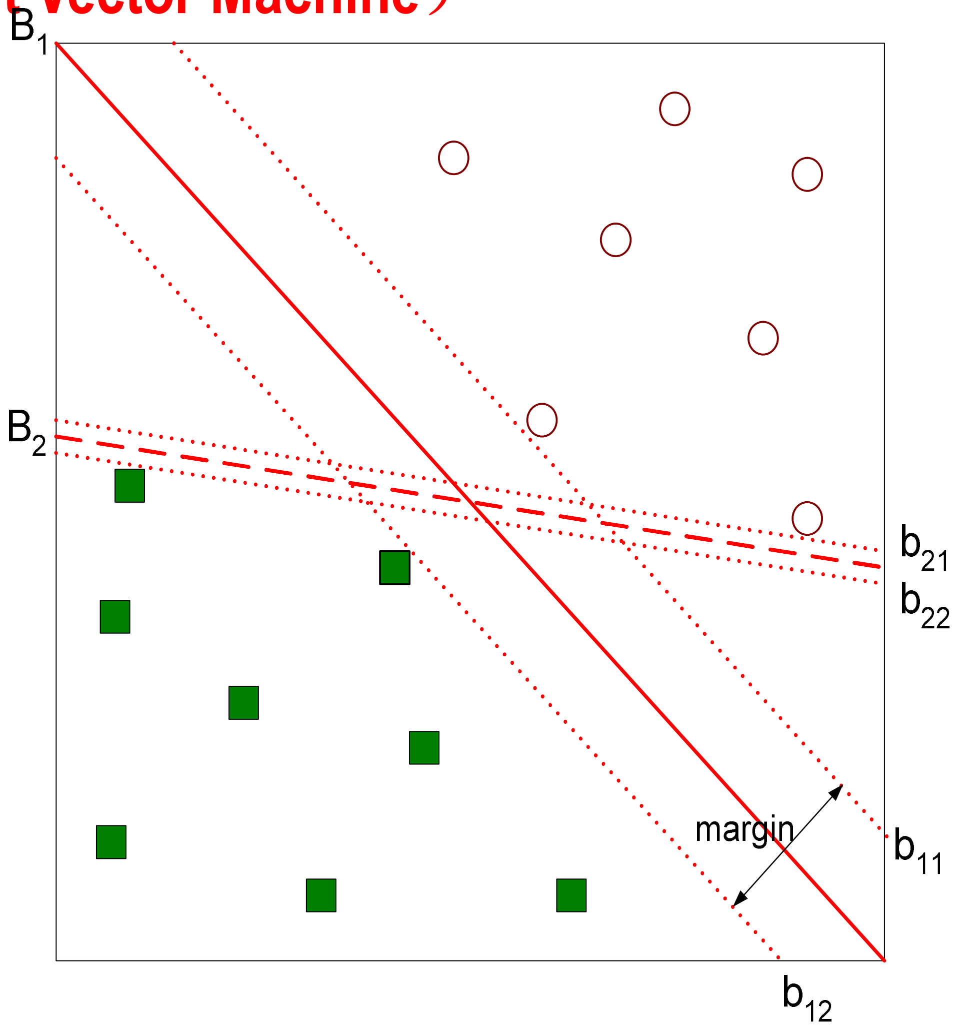
- 分类——支持向量机（**Support Vector Machine**）



其他可行解

数据挖掘基础

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



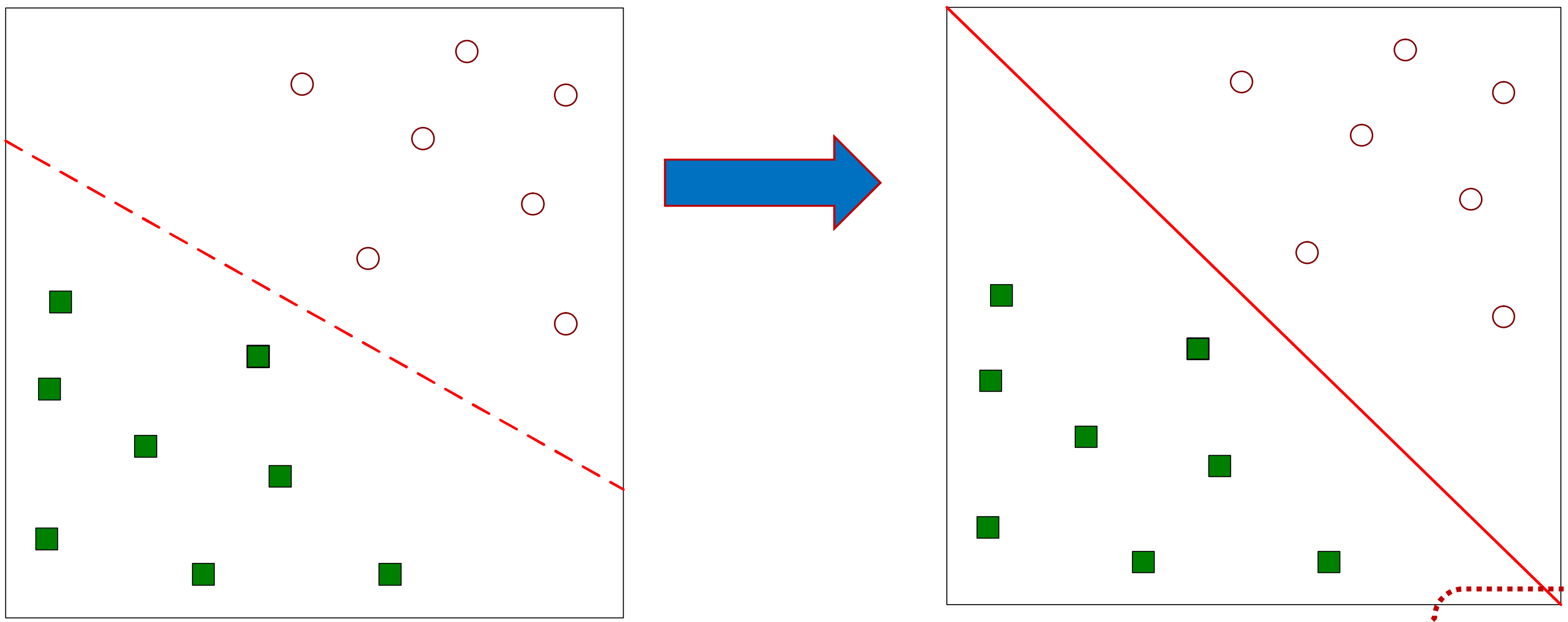
找到使间隔最大化的超平面

=>B1比B2更好

数据挖掘基础

- 分类——区别

感知机 $f(x) = sign(w \cdot x + b)$ SVM



接下来介绍

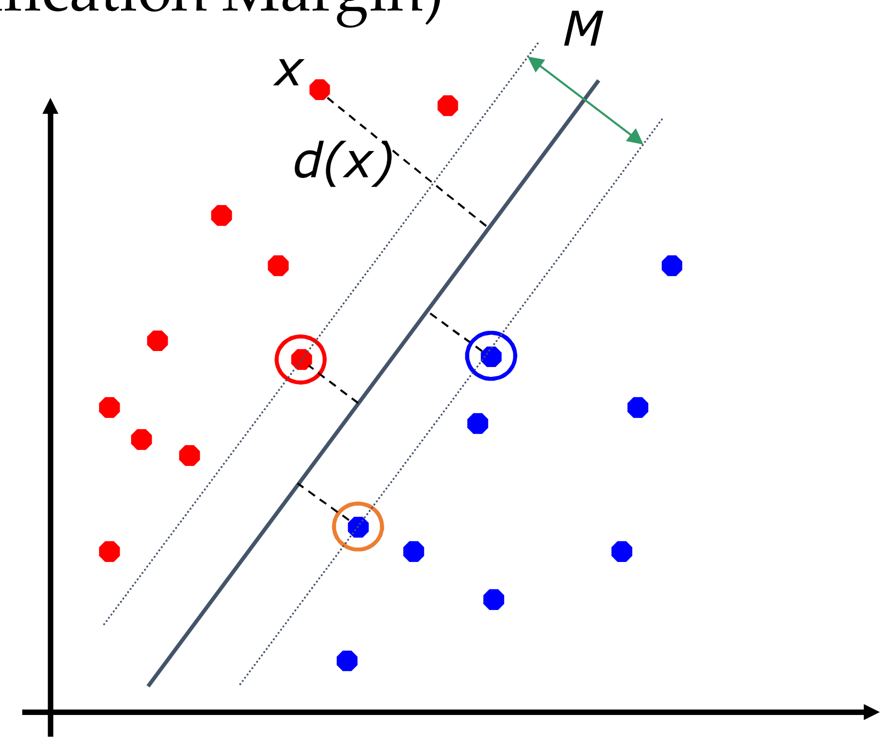
优化目标:

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

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

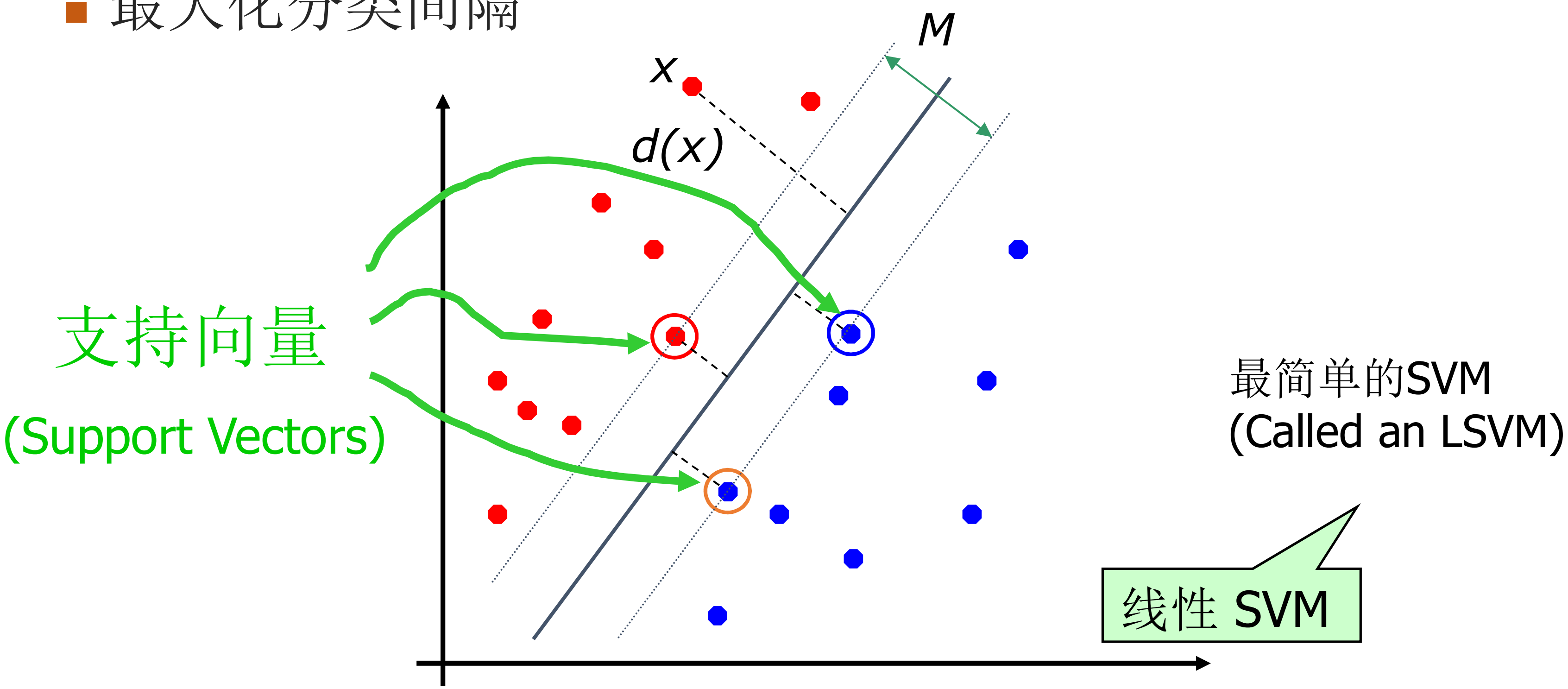
$$s.t.\ y_i(w^T.x_i + b) \geq 1$$

- 分类——支持向量机
- 研究起因
  - 如何找到最优的切分面
    - 分类间隔(Classification Margin)



数据挖掘基础

- 分类——支持向量机
- 研究起因
  - 如何找到最优的切分面
    - 最大化分类间隔

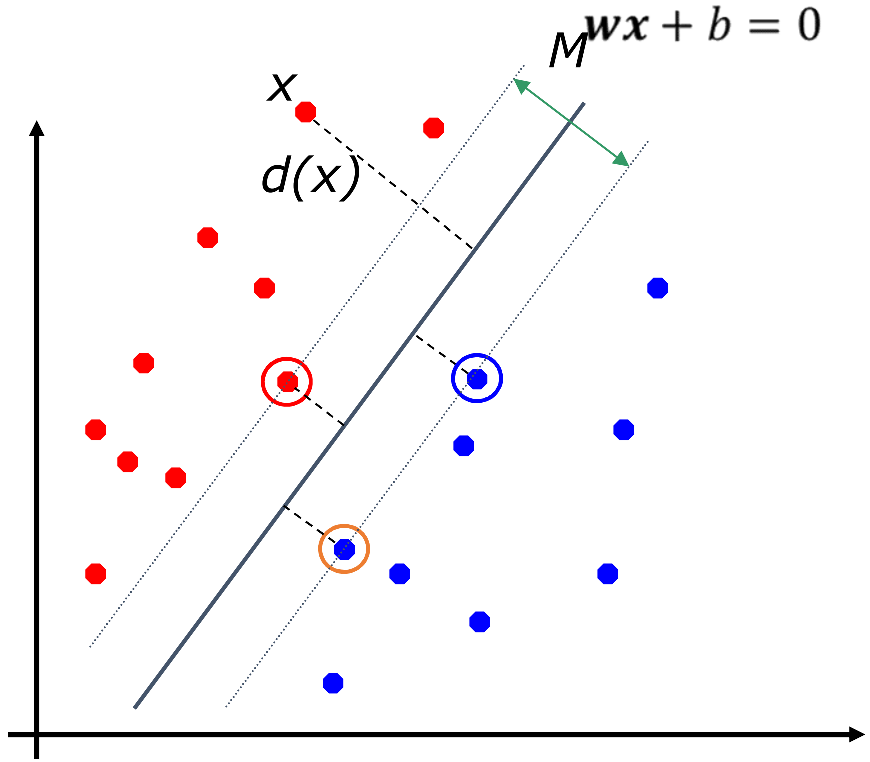


- 分类——支持向量机
- 研究起因
  - 如何找到最优的切分面
    - 最大化分类间隔
      - 直观上最有效
      - 概率的角度，就是使得置信度最小的点置信度最大
      - 即使我们在选边界的时候犯了小错误，使得边界有偏移，仍然有很大概率保证可以正确分类绝大多数样本
      - 很容易实现交叉验证，因为边界只与极少数的样本点有关
      - 有一定的理论支撑(如VC维)
      - 实验结果验证了其有效性

数据挖掘基础

- 给定训练数据$x_i \in X=R^n$，其类标签为$y_i \in Y=\{+1,-1\}$
- 点x_i到平面$w \cdot x + b = 0$的距离：

$$r_i = d(x_i) = \frac{y_i(w \bullet x_i + b)}{\|w\|}$$



- 切分面满足距离：

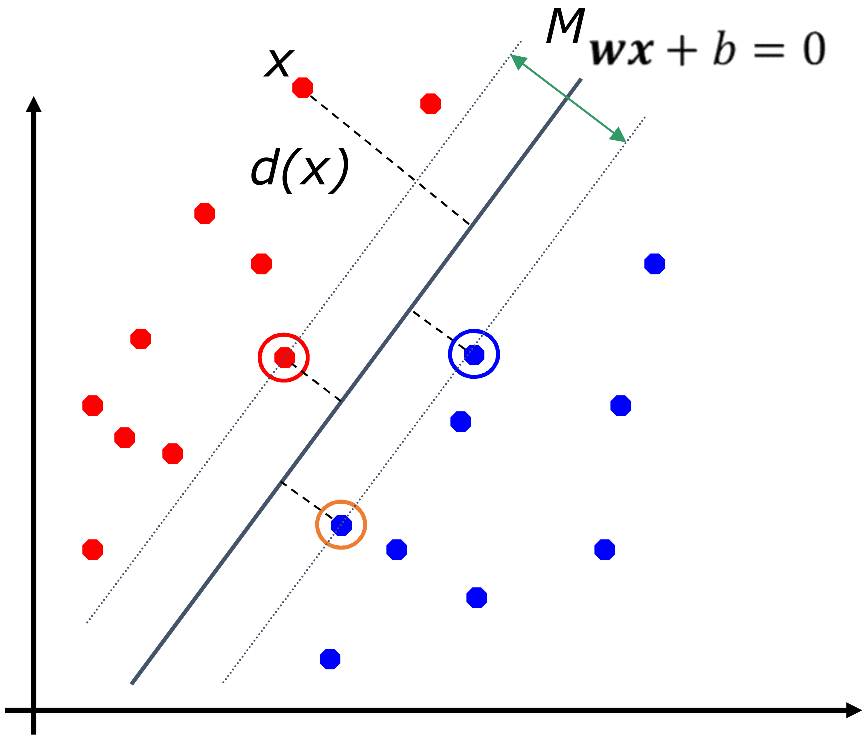
$$r = \min_{i=1,\dots,N} \frac{y_i(w \bullet x_i + b)}{\|w\|}$$

数据挖掘基础

- 目标：最大化 $r = \min\limits_{i=1,\ldots,N} \dfrac{y_i(w \bullet x_i + b)}{\|w\|}$

即为优化问题：

$$\boxed{\begin{aligned} &\max_{w,b} \quad r \\ &s.t. \quad \frac{y_i(w \bullet x_i + b)}{\|w\|} \geq r, i = 1,2,\ldots N \end{aligned}}$$



注：约束保证了所有点到切分面的距离都不比r小

- 间隔转换

$$\max_{w,b} \ r$$
$$s.t. \ \frac{y_i(w \cdot x_i + b)}{\|w\|} \geq r, i = 1,2,\ldots N$$

$$r = \frac{\hat{r}}{\|W\|}$$

$$\max_{w,b} \ \hat{r}$$
$$s.t. \ y_i(w \cdot x_i + b) \geq \hat{r}, i = 1,2,\ldots N$$

r_i: 几何间隔

$\hat{r}$： 函数间隔

函数间隔取值不影响最优化问题的解，因为若将w和b按比例改变为λw和λb，则函数间隔变为λ $\hat{r}$，而超平面不会变化。

数据挖掘基础

- 进一步简化

$$\max_{w,b} \ r$$
$$s.t. \ \frac{y_i(w \bullet x_i + b)}{\|w\|} \geq r, i = 1, 2, \ldots N$$

$$r = \frac{\hat{r}}{\|W\|}$$

$$\max_{w,b} \ \hat{r}$$
$$s.t. \ y_i(w \bullet x_i + b) \geq \hat{r}, i = 1, 2, \ldots N$$

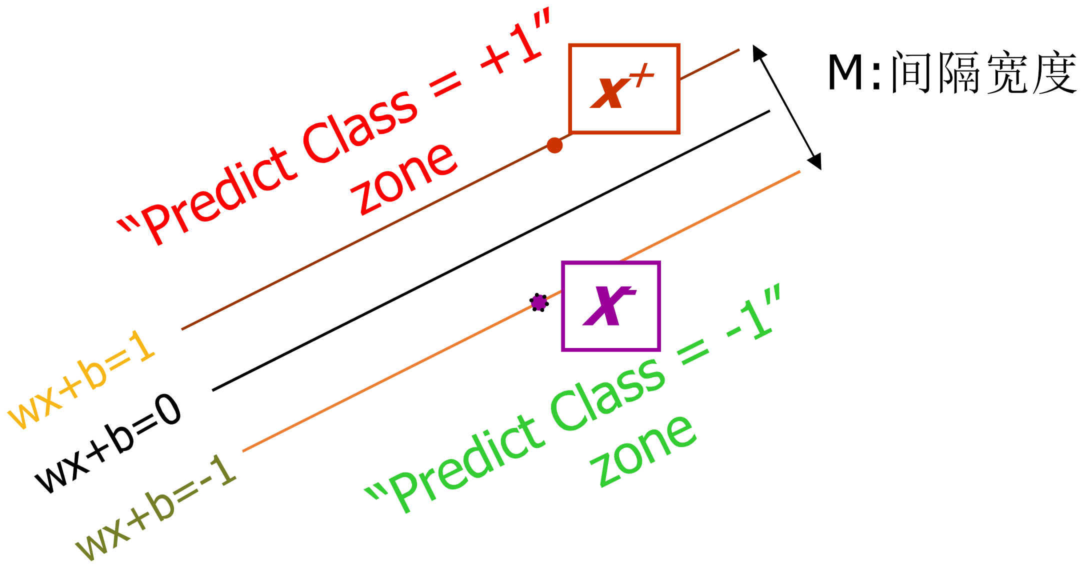
令 $\hat{r} = 1$

$$\max_{w,b} \frac{1}{\|w\|} \xleftrightarrow{\text{等价}} \min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t. \ y_i(w \bullet x_i + b) - 1 \geq 0, i = 1, 2, \ldots N$$

优化目标：平方和系数都是为了求导方便

数据挖掘基础

- 理解



有：

$$w \bullet x^+ + b = +1$$

$$w \bullet x^- + b = -1$$

$$w \bullet (x^+ - x^-) = 2$$

$$\boxed{M = \frac{w}{\|w\|} \bullet (x^+ - x^-) = \frac{2}{\|w\|}}$$

其实就是两边支持向量之间的距离
（即2倍的支持向量到切分面的距离）

数据挖掘基础

- 如果两个数据对象的Cosine相似度为1，则这两个数据对象可以被当作相同的数据
  - A.该论断错误　B.该论断正确
- 朴素（Naive）贝叶斯为什么被叫做“朴素”？
- 如果一个数据集包含两类数据，其中一类占的比例约为0.2%，则_______。
  - 该数据集的信息熵值趋近于1。
  - 该数据集的信息熵值趋近于0。
  - 该数据集的信息熵值趋近于0.3。
  - 该数据集的信息熵值趋近于0.8。

$$Entropy(t) = -\sum_j p(j|t)\log p(j|t)$$

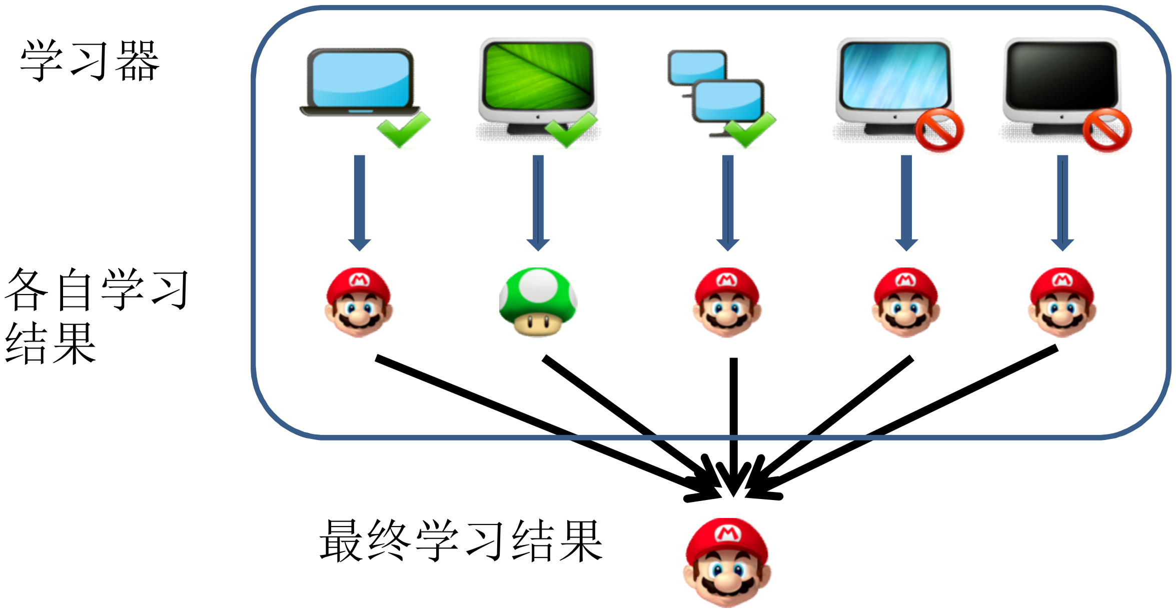
数据挖掘基础

- 下列不属于K近邻方法关键步骤的是_____
  - A. 抽样样本的比例
  - B. 距离函数的选择
  - C. K值的选取
  - D. 分类决策规则的确定
- 如何判断决策树的分裂过程何时结束？
  - A. 每个叶子结点中的所有数据属于同一类时
  - B.每个叶子结点中所有数据有相同的属性值时
  - C:继续分裂不能带来分类效果的提升时
  - D:叶子结点中的数据记录数小于某个阈值时

数据挖掘基础

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
  - 单树最好的分类点：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: 根据均匀概率重复（有放回）抽样

Round 1

x	0.1	0.2	0.2	0.3	0.4	0.4	0.5	0.6	0.7	0.7
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

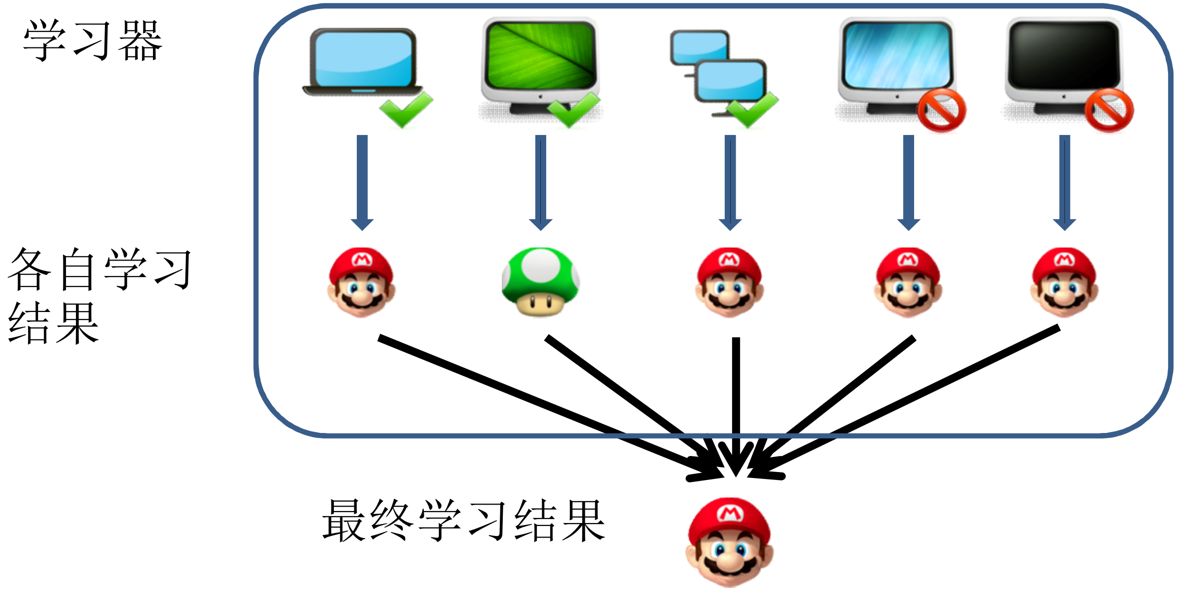
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
	1	1	1	-1	-1	-1	-1	1	1	1

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）
    - 权重包含两层意思：被抽样的概率或者被错分时的权重

- 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

- 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),….(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）

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

$w_j^{(i)}$ 是第j个样本在第i轮的权重

where Z_i is the normalization factor

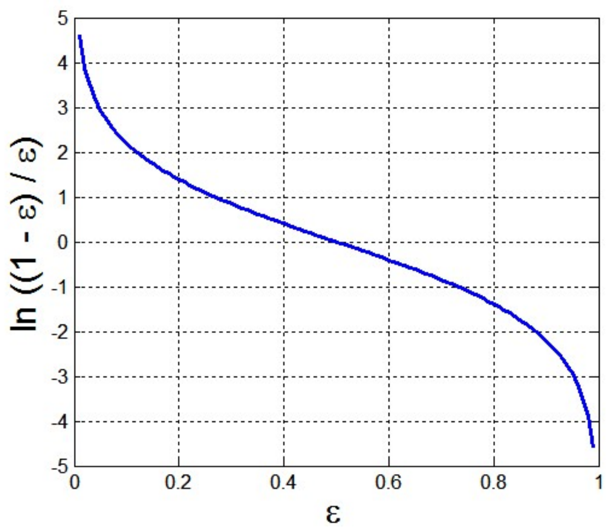
数据挖掘基础

- 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

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

    - Error rate: (*i* = index of classifier, *j*=index of instance)

$$\varepsilon_i = \frac{1}{N}\sum_{j=1}^{N} w_j \delta\left(C_i(x_j) \neq y_j\right)$$



- 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)

- 分类——模型评估方法：混淆矩阵

<table>
<tr><td></td><td colspan="3">PREDICTED CLASS</td></tr>
<tr><td rowspan="3">ACTUAL CLASS</td><td></td><td>Class=Yes</td><td>Class=No</td></tr>
<tr><td>Class=Yes</td><td>a
(TP)</td><td>b
(FN)</td></tr>
<tr><td>Class=No</td><td>c
(FP)</td><td>d
(TN)</td></tr>
</table>

- Most widely-used metric $$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

数据挖掘基础

- 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

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

数据挖掘基础

分类——模型评估方法：Cost-Sensitive Measures

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

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

- 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)

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

F值 **=** 正确率 ***** 召回率 *** 2 / (**正确率 **+** 召回率**)**

（**F** 值即为正确率和召回率的<u>调和平均值</u>）

F=(2/(1/r+1/p))

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正确率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__________。

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

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

$$\text{F-measure }(F_1) = \frac{2rp}{r+p} = \frac{2\times TP}{2\times TP+FP+FN}$$

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

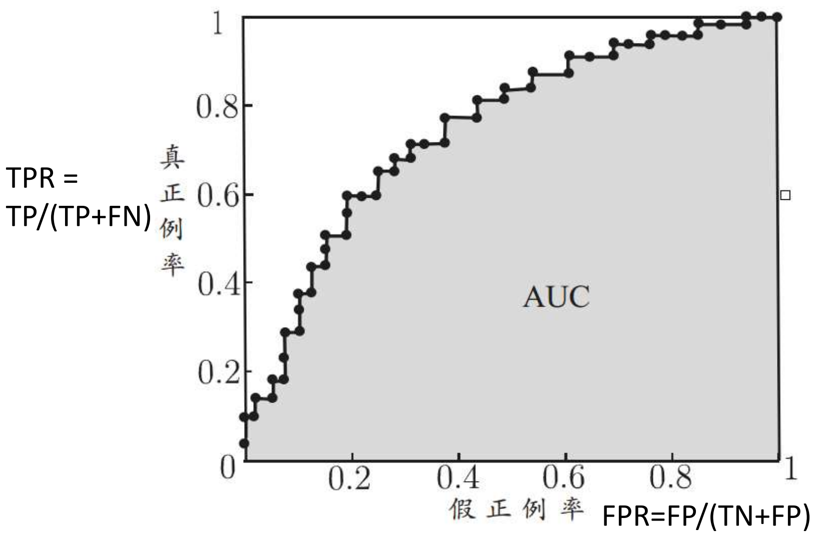
- 分类——模型比较方法：Cost-Sensitive Measures
  - ROC (Receiver Operating Characteristic)与AUC
    - “受试者工作特征”，ROC曲线的面积就是AUC（Area Under the Curve）。AUC用于衡量“二分类问题”机器学习算法性能（泛化能力）
    - Developed in 1950s for signal detection theory to analyze noisy signals
    - Characterize the trade-off between **positive hits** and **false alarms**
    - ROC curve plots TP (on the y-axis) against FP (on the x-axis)
    - 样本中的真实正例类别总数即TP+FN。
      - **真正例率**TPR即True Positive Rate，TPR = TP/(TP+FN)。
    - 样本中的真实反例类别总数为FP+TN。
      - **假正例率**False Positive Rate，FPR=FP/(TN+FP)。

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

真正例率等于 预测为正且实际为正的样本在所有的正样本中的比例
假正例率等于 预测为正但实际为负的样本占所有负样本的比例

- 分类——模型比较方法：Cost-Sensitive Measures
  - ROC (Receiver Operating Characteristic)与AUC

ROC图的绘制：给定 m^+ 个正例和 m^- 个负例，根据学习器预测结果对样例进行排序，将分类阈值设为每个样例的预测值，当前标记点坐标为 (x,y)，当前若为真正例，则对应标记点的坐标为 $(x, y+\frac{1}{m^+})$；当前若为假正例，则对应标记点的坐标为 $(x+\frac{1}{m^-}, y)$，然后用线段连接相邻点.



基于有限样例绘制的 ROC 曲线与 AUC

假设ROC曲线由 $\{(x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m)\}$ 的点按序连接而形成 $(x_1=0,\ x_m=1)$，则：AUC可估算为：

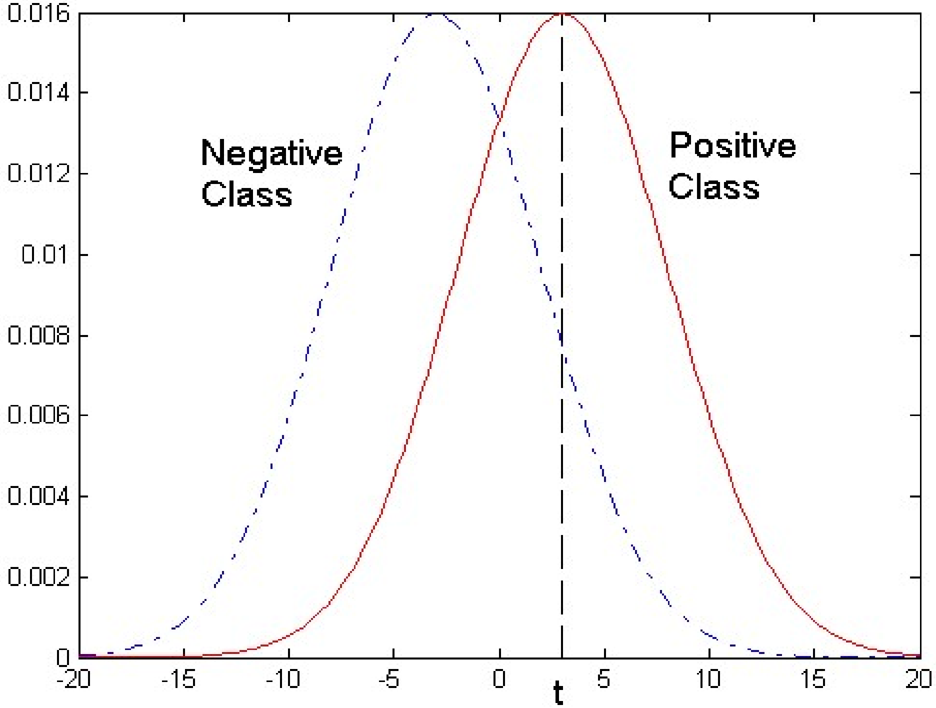
$$\mathrm{AUC} = \frac{1}{2}\sum_{i=1}^{m-1}(x_{i+1}-x_i)\cdot(y_i+y_{i+1})$$

AUC衡量了样本预测的排序质量。

- 分类——模型比较方法：Cost-Sensitive Measures
  - ROC (Receiver Operating Characteristic)与AUC

- 1-dimensional data set containing 2 classes (positive and negative)

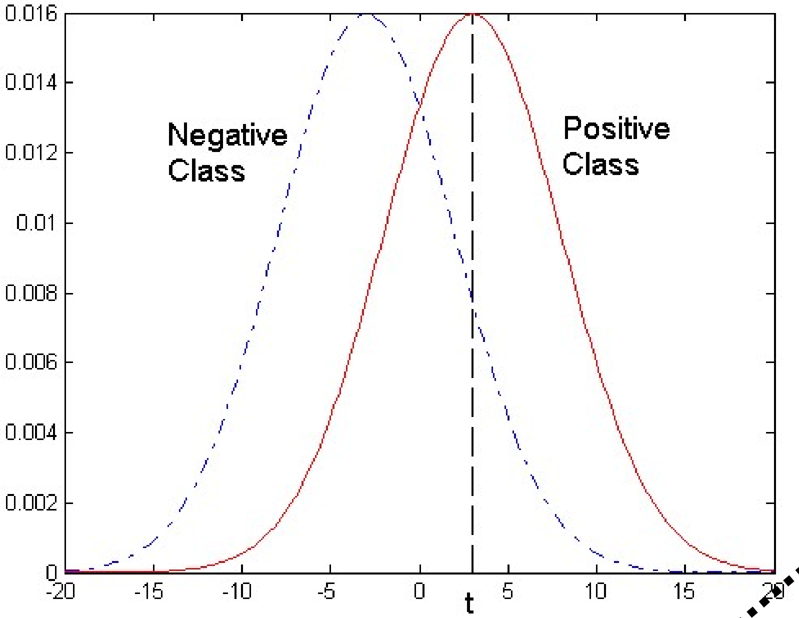
- any points located at x > t is classified as positive

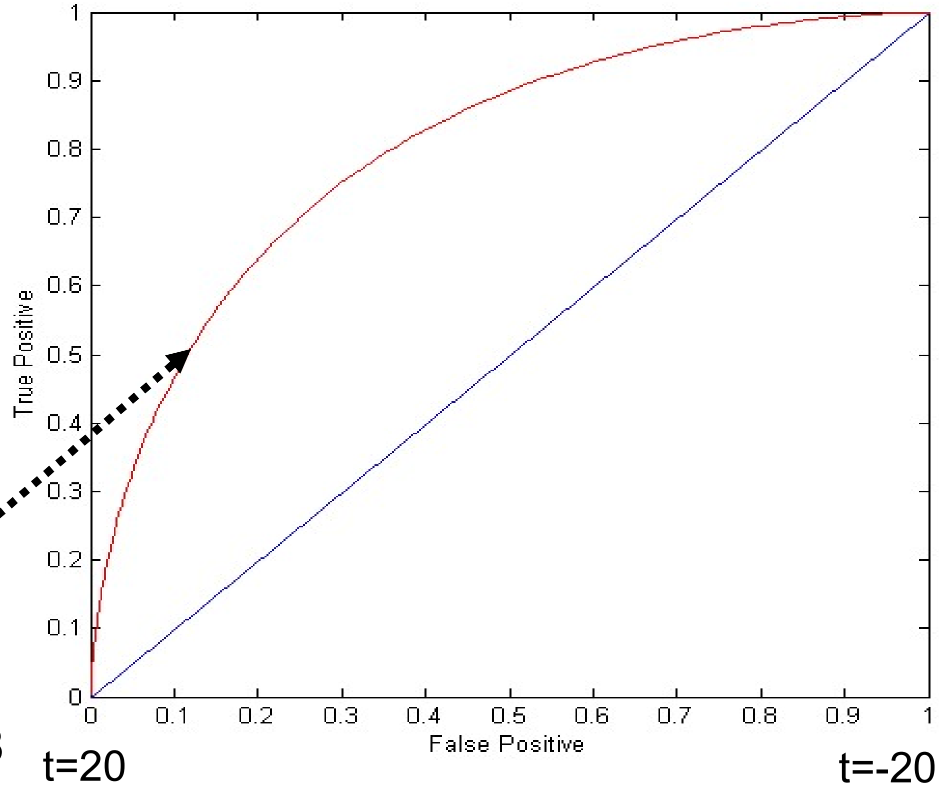


- 分类——模型比较方法：Cost-Sensitive Measures
  - ROC (Receiver Operating Characteristic)与AUC

- 1-dimensional data set containing 2 classes (positive and negative)

- any points located at x > t is classified as positive



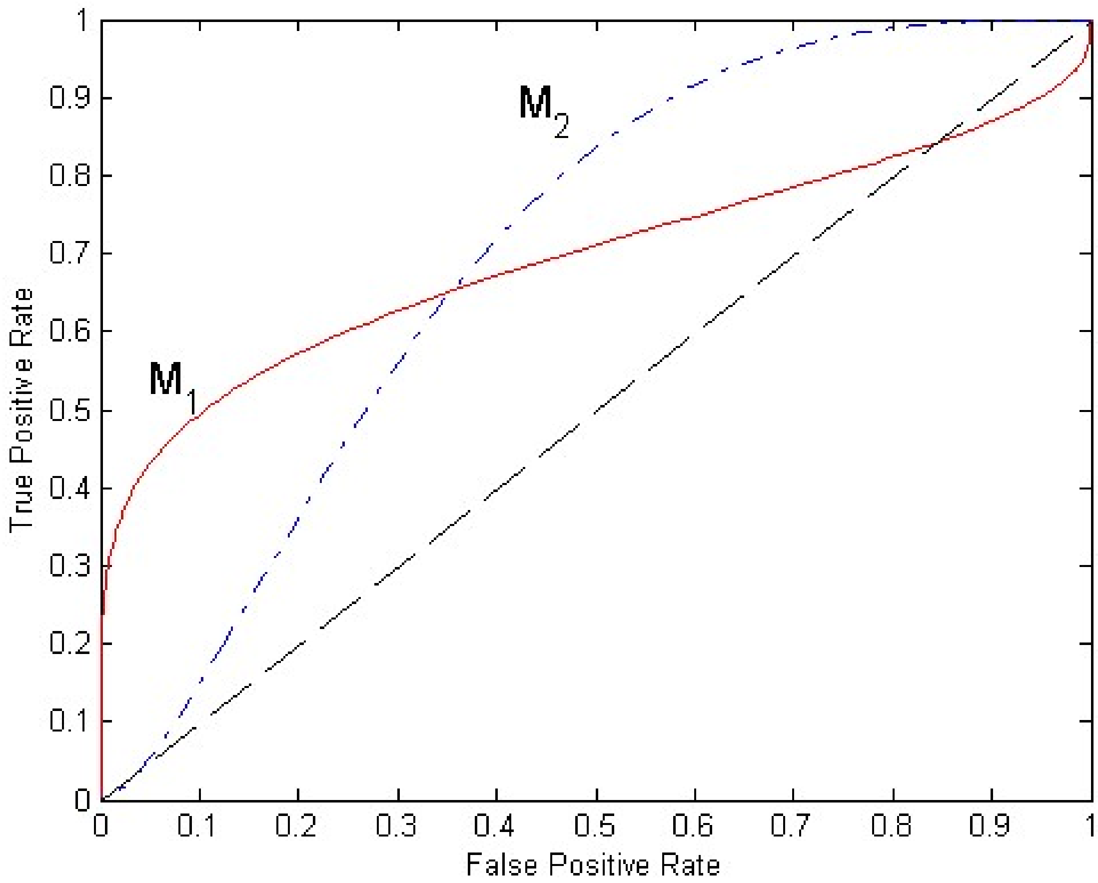


At threshold t:

TP（y）=0.5, FN=0.5, FP(x)=0.12, TN=0.88

数据挖掘基础

- 分类——模型比较方法：Cost-Sensitive Measures
  - ROC (Receiver Operating Characteristic)与AUC



- No model consistently outperform the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5

- 分类——模型比较方法：Test of Significance
- 关于性能比较：
  - 测试性能并不等于泛化性能
  - 测试性能随着测试集的变化而变化
  - 很多机器学习算法本身有一定的随机性

Given two models:

Model M1: accuracy = 85%, tested on 30 instances

Model M2: accuracy = 75%, tested on 5000 instances

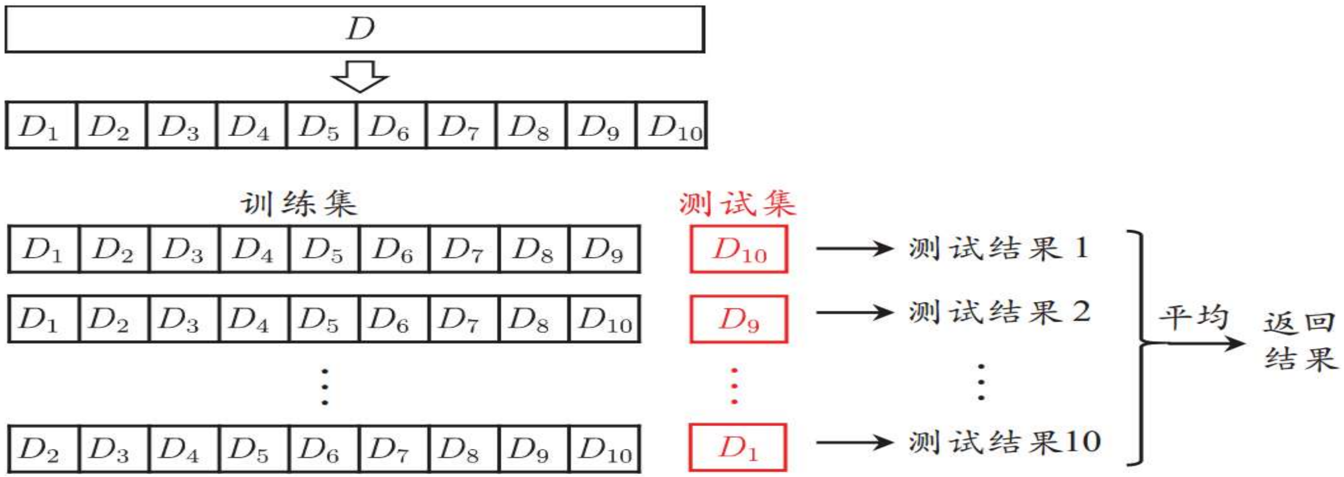
- 进行假设检验，判断差别是否具有统计意义

假设检验为学习器性能比较提供了重要依据，基于其结果我们可以推断出若在测试集上观察到学习器A比B好，则A的泛化性能是否在统计意义上优于B，以及这个结论的把握有多大。

分类——模型验证方法:

交叉验证法（Cross Validation）：

将数据集分层采样划分为k个大小相似的互斥子集，每次用k-1个子集的并集作为训练集，余下的子集作为测试集，最终返回k个测试结果的均值，k最常用的取值是10.



10 折交叉验证示意图