DataMining

P1—P8:

**Introduction**

1、数据挖掘领域前沿发展---国际会议

**International Conference on Data Mining**

**International Conference on Data Engineering**

**International Conference on Machine Learning**

**International Joint Conference on Artificial Intelligence**

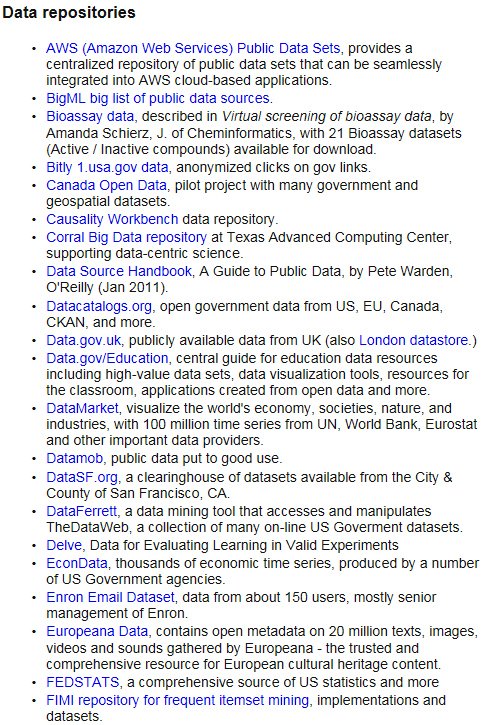
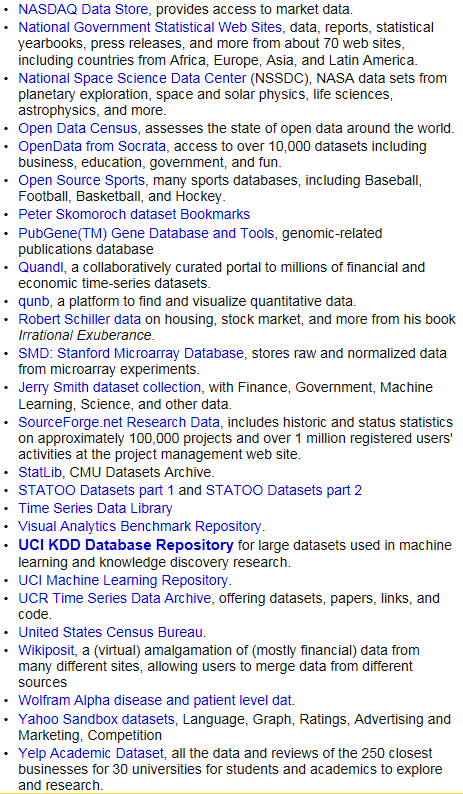
**Pacific-Asia Conference on Knowledge Discovery and Data Mining**

**ACM SIGKDD Conference on Knowledge Discovery and Data Mining**

**期刊：**

 [](http://www.google.com.au/url?sa=i&rct=j&q=ieee+computer+society&source=images&cd=&cad=rja&docid=EG1tghtrs_DUdM&tbnid=KHwKdrVv-ulsjM:&ved=0CAUQjRw&url=http://ewh.ieee.org/r10/w_australia/computer/&ei=NEIgUtj2LsSiigern4G4BA&bvm=bv.51495398,d.aGc&psig=AFQjCNGoKGKAkMoQycL5P4dHgGrmOBUDYQ&ust=1377932152157603)

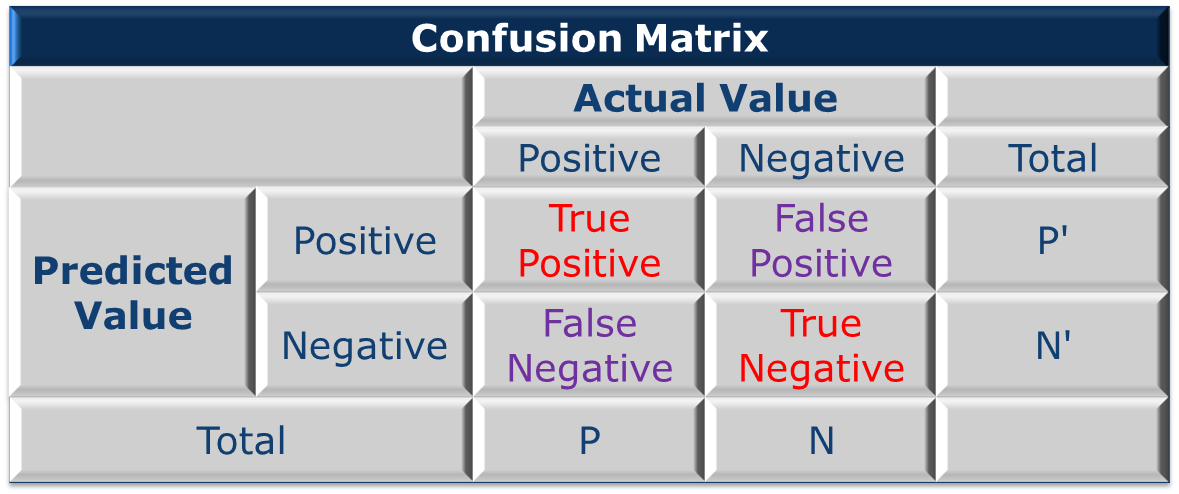
公开数据：



***From Data To Intelligence***



***混淆矩阵Confusion Matrix：***

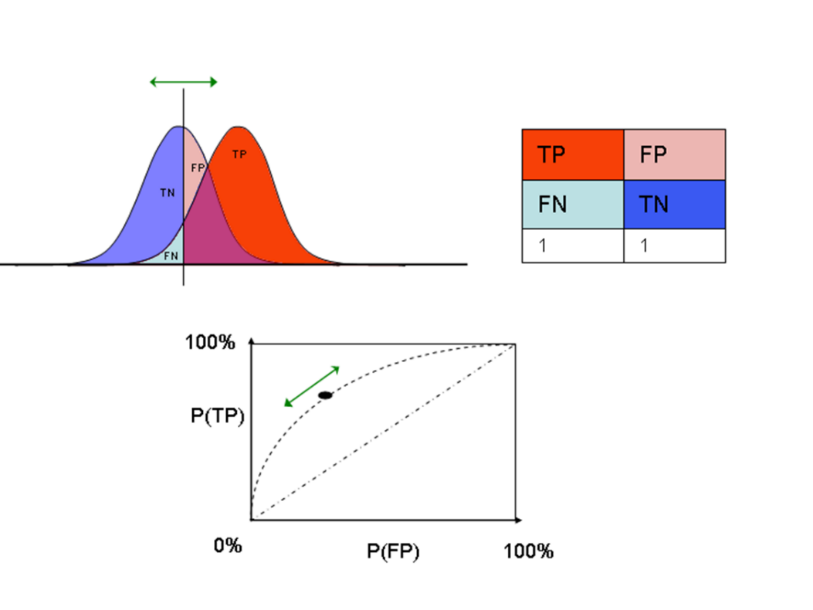


TPR=TP/(TP+FN)

TNR=TN/(TN+FP)

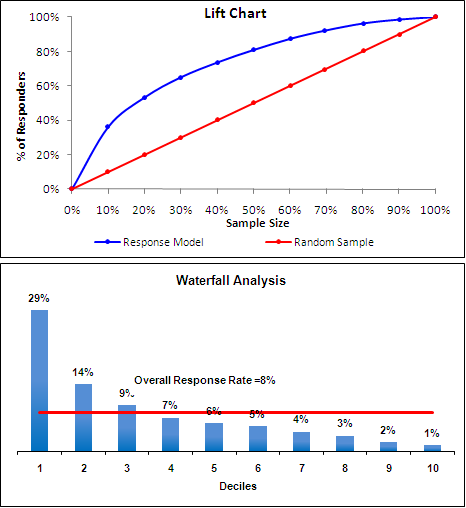
Accuracy=(TP+TN)/(P+N)

***Receiver Operating Characteristic ROC曲线***



AUC：Area Under Curve [0,1], 越接近1越好,ROC曲线覆盖的面积

***Lift Analysis 提升度分析***

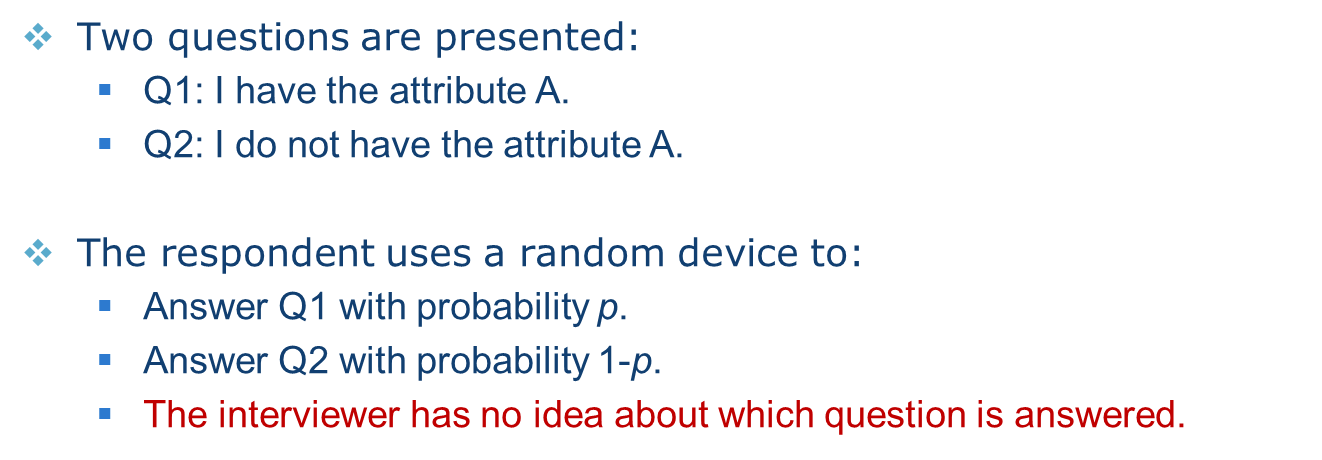


***Eg: 100个人中有8%的人会购买，经过模型处理按照购买可能性排序后 前10%的人中有29%的人会购买。在资源有限情况下，不需要对100个人进行筛选，只需要从前20%的人群中进行筛选就可以筛选出绝大部分愿意购买商品的目标客户。。***

***DM Techniques - Clustering***



***Privacy Protection***

******



P\*(True): 问卷中选项True的比例

P(True): 真实情况中have A的比例

p: 选择Q1问题来回答的概率

***Cloud Computing***

* Pay As You Go
* Software as a Service
* Platform as a Service
* Infrastructure as a Service

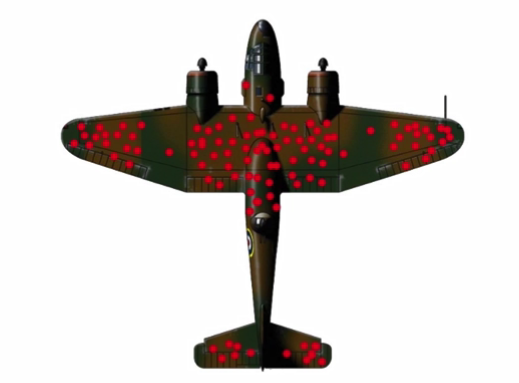
通俗来讲，需要用的时候才去购买或租用，即Pay As You Go, 将software / platfor / infrastructure当成一种服务，而不是自己必须拥有的资产

* Algorithm： Always start with simple ones.

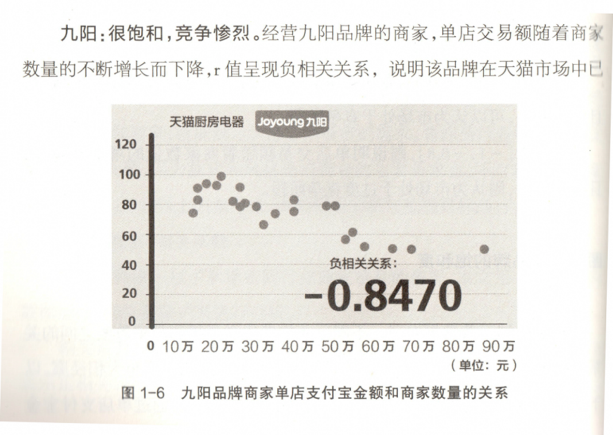
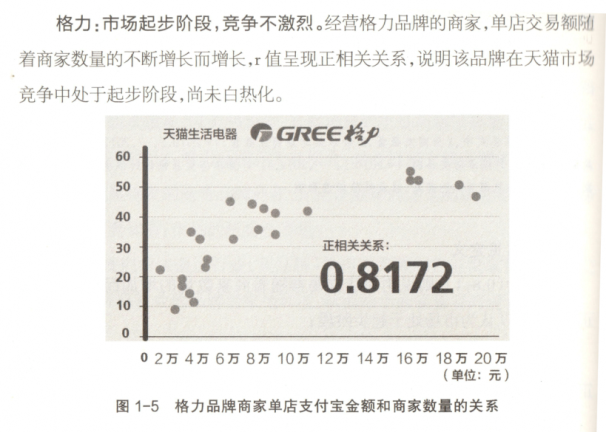
***Survivorship Bias***

***红色点为飞机中弹区域，那么，应该在飞机什么位置加装防护钢板？***

* ***实际在驾驶舱、油箱、发动机部位加装，而不是中弹密集区***
* ***原因：驾驶舱 油箱 发动机一旦中弹，飞机基本就没了；实际存活下来的飞机即样本数据本身是幸存者的偏差数据，致命位置中弹的数据都没有，因此偏差数据不可靠。***



避免单向思维：从不同角度可能有截然相反的解释

缺少数据的时间维度，两个曲线都可以从左往右或从右往左看，两种视角下的解释截然不同。

**Data Preprocessing**

***How to handle missing data?***

* Ignore
  + Remove samples/attributes with missing values
  + The easiest and most straightforward way
  + Work well with low missing rates
* Fill in the missing values manually
  + Recollect the data
  + Domain knowledge（领域知识）
  + Tedious/Infeasible
* Fill in the missing values automatically
  + A global constant
  + The mean or median
  + Most probable values
* More art than science

***Anomaly（异常点）vs. Outlier（离群点）***

***Local Outlier Factor***

***一定是考虑相对距离，不是绝对距离***

***Eg.***

***Attribute Types***

* Continuous
  + Real values: Temperature, Height, Weight …
* Discrete
  + Integer values: Number of people …
* Ordinal
  + Rankings: {Average, Good, Best}, {Low, Medium, High} …
* Nominal
  + Symbols: {Teacher, Worker, Salesman}, {Red, Green, Blue} …

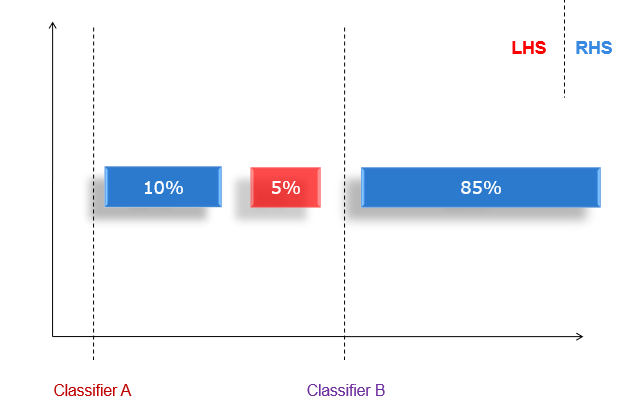
不同的编码会导致空间结构的不同，如果{Red, Green, Blue}编码为0,1,2，则相当于默认了Red和Green的距离比Red和Blue的距离更近；

可以考虑独热编码

* String
  + Text: “Tsinghua University”, “No. 123, Pingan Avenue” …

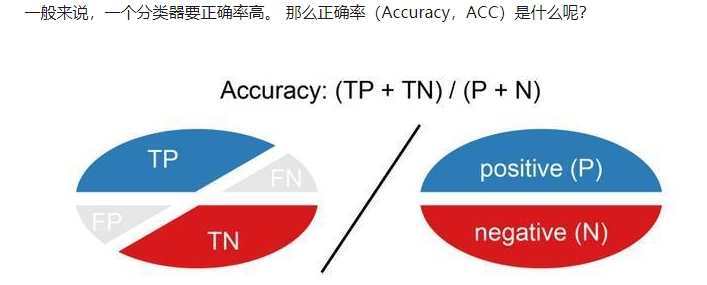
***Imbalanced Datasets---不平衡样本***

***---- 总体准确率不适用于不平衡数据集 ----***



Classifier A: 准确率95%

Classifier B: 准确率85%

但是A很容易将负样本错误地分类为正样本，在医学领域 这样会产生很严重的后果，如将癌症患者诊断为健康。从实际效果来看，虽然B准确率比A低，但B的实际效果比A好

Precision: 查准率=正确预测的正样本数 / 所有预测为正样本的样本数

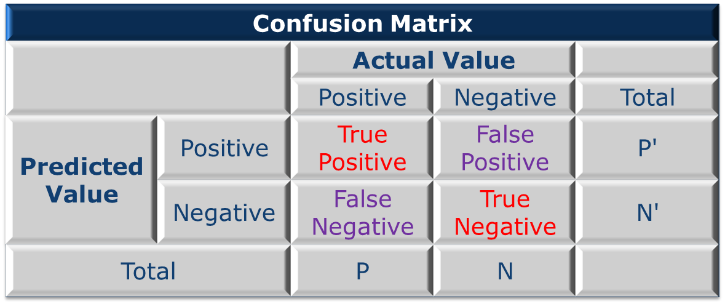
Recall: 查全率=正确预测的正样本数 / 实际正样本数

TP: 正确的正样本（实际pos，预测为pos），将正样本positive正确分类为正样本

FN: 错误的负样本（实际pos, 预测为neg），将正样本错误分类为负样本

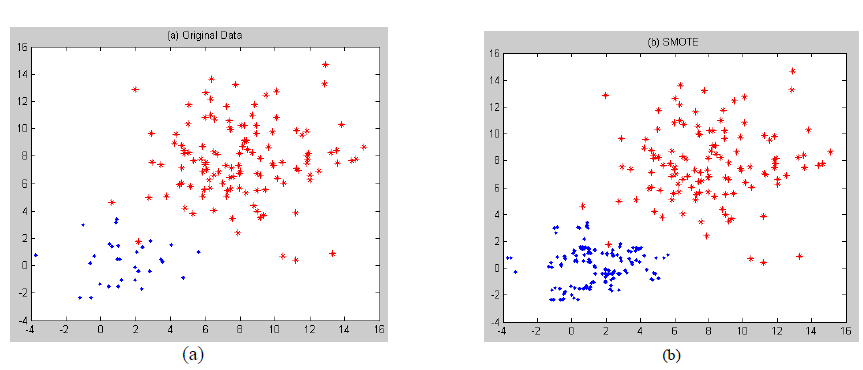
FP: 错误的正样本（实际neg，预测为pos），将负样本错误分类为正样本

TN: 正确的负样本（实际neg，预测为neg，将负样本正确分类为负样本



***Over-Sampling***

***不均衡样本中少的数据如蓝色点，在其周围随机生成更多的点作为蓝点样本来均衡蓝点和红点的比例***



***Boundary Sampling***

对于大量数据导致计算压力大，可以只采样数据的边缘点作为代表性样本参与计算。如下图，所有的点中，边缘的点在分类中即可代表该区域所有的点参与分类运算，极大节省运算压力和时间



***Normalization 归一化 标准化***

* **Min-max normalization:**

---> [, ]

范围缩放，有明确上下限的数据可以采用这种方法

* Let income range $12,000 to $98,000 be normalized to [0.0, 1.0]. Then $73,600 is mapped to:
* **Z-score normalization (μ: mean, σ: standard deviation):**

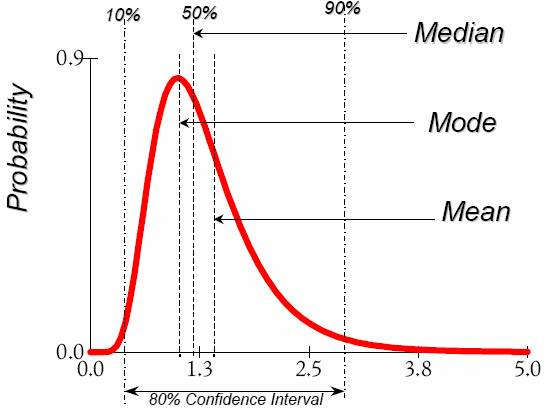
一般偏离均值3 以外的点就已经很少，离群点检测可以用

* Let μ = 54,000, σ = 16,000. Then

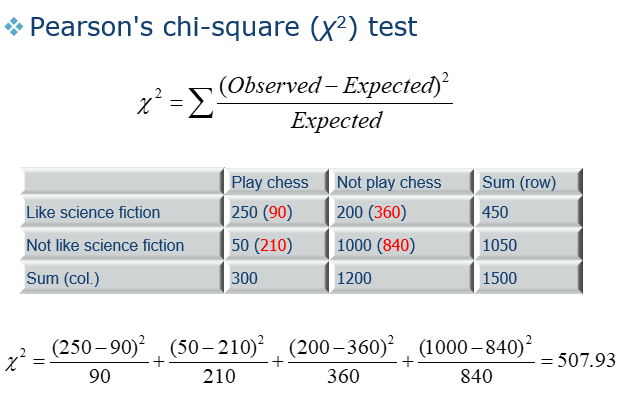
（13600偏离均值 1.225个方差σ）

***Data Description***

* **Mean**
  + Arithmetic mean
* **Median**
* **Mode**
  + The most frequently occurring value in a list
  + Can be used with non-numerical data.
* **Variance**
  + Degree of diversity



* **correlation**
* Pearson’s product moment correlation coefficient
* If rA, B > 0, A and B are positively correlated.
* **If rA, B = 0, no linear correlation between A and B.（非线性相关 而不是不相关）**
* If rA, B < 0, A and B are negatively correlated.
* **Pearson's chi-square (*χ*2) test 卡方检验**



调查1500人，其中下棋300人，不下棋1200人； 喜欢科幻450人，不喜欢科幻1050人；

蓝色为真实观测的值observed，如250表示即下棋又喜欢科幻实际有250人，红色为预期的值expected.

---> 假设下棋和喜欢科幻不相关，则喜欢科幻中下棋的人的比例与总人数中对应比例相同，即喜欢科幻的450中，预期有300/1500\*450=90个人下棋，360个人不下棋；同理不喜欢科幻的人中，预期有300/1500\*1050=210个人下棋，840个人不下棋

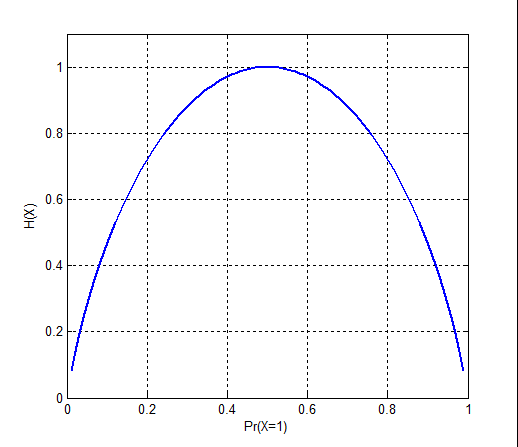
卡方检验值 *χ*2 = np.sum( (observed - expected)\*\*2 / expected ) = 507.93

(查表，很大值，说明两者之间存在相关性)

***Feature Selection***

***Entropy 熵***

***熵计算公式****：*  ***熵越大，不确定性越高***

** **

1. 当不确定性为0.5(p=0.5)时, 熵最大，即不确定性最高：

假设有属性

由图可知：non-smoker(占总人数60%)中女性的概率：0.8， 男性的概率为：0.2

Smoker(占总人数40%)中女性的概率为0.05, 男性的概率为0.95

不抽烟人群中熵：

（越大，不确定性越高）

抽烟人群中熵：

(越小，不确定性越低)

由于 < , 则说明在抽烟人群中，对性别判断的更加准确

属性X整体上的效能：

即不抽烟人的系数0.6 \* 不抽烟人的熵 + 抽烟人的系数0.4 \* 抽烟人的熵

---> 原始熵**,** 属性X后的熵

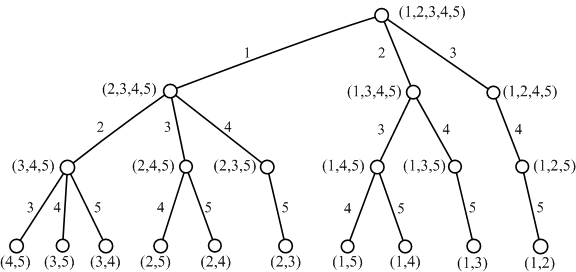
**Information Gain (信息增益)：**

**Gain(S, X) = - = 1 – 0.5477 = 0.4532**

当新增了一个属性后，对系统原来的不确定性降低有多少。可根据每一个属性对系统熵的信息增益大小来决定哪些属性是优先考虑的属性，选择信息增益更大的属性。

***Feature Subset Search --- Branch and Bound***

***从N个属性中如何选择出最优的K个属性***



* Top K Individual Features
* Sequential Forward Selection
* Sequential Backward Selection
* Optimization Algorithms

Simulated Annealing 模拟退火

Tabu Search 禁忌搜索

Genetic Algorithms 遗传算法

***Feature Extraction***

***Principal Component Analysis （PCA）主成分分析***

***2D Example\_1***

* *Data:* ***Gaussian Distribution***
* *Variance:* ***Information （数据散布越光 信息越多）***
* *Ellipse:* ***Major Axis*** *vs. Minor Axis*
* ***Select attribute corresponding to the Major Axis****.（优先选择information多的属性）*

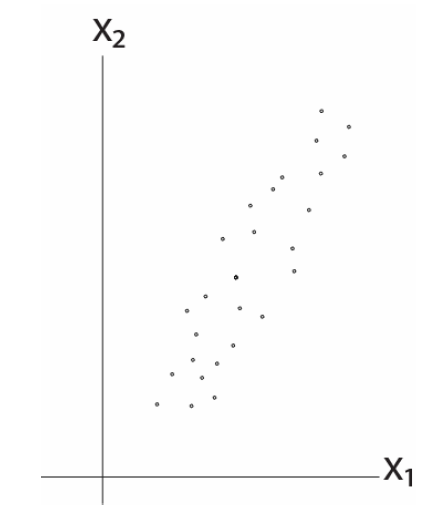
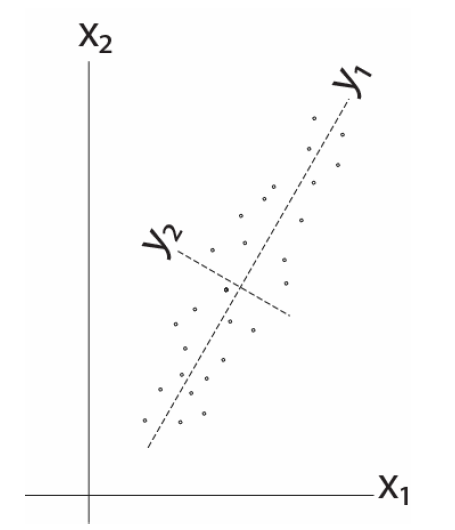
Minor

Major

Feature X1

Feature X2

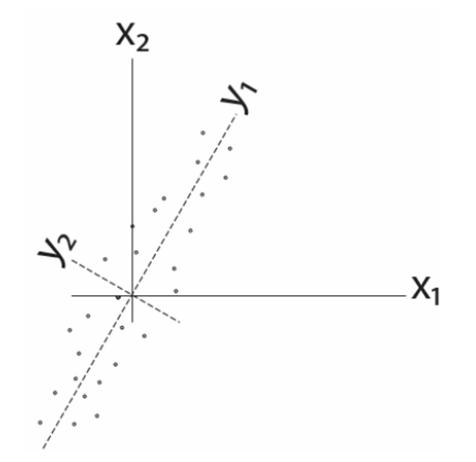
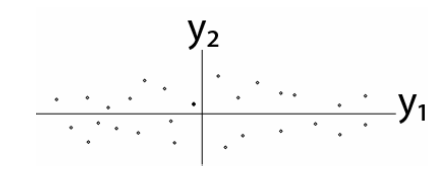
***2D Example\_2 （实际数据可能x1 x2存在相关，如下例）***

***5 ***

*x1和x2存在相关性correlation, 需消除两者的相关性后再根据散布information来选择。*

*先把数据中心点去掉，即****去中心****，让数据中心与坐标轴原点重合；*

***坐标轴旋转****成 y1和y2，* ***去除两者的线性关系****，再根据散布选特征*

*** ***

***协方差矩阵：***

*======remove correlation=====>*

*Math Calculate:*

*---> ---> ----*

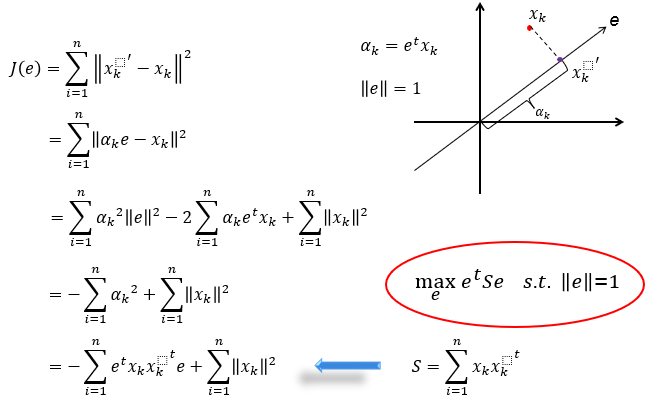
*-> ---> =*

*->*

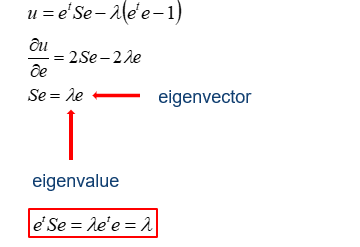
其中 (特征分解)中D是对角矩阵。

若想S(Y)为对角矩阵，则PQ =I (单位矩阵) ,由于Q是正交矩阵，则 P = Q-1 = QT

***Lagrange Multipliers 拉格朗日乘数法***

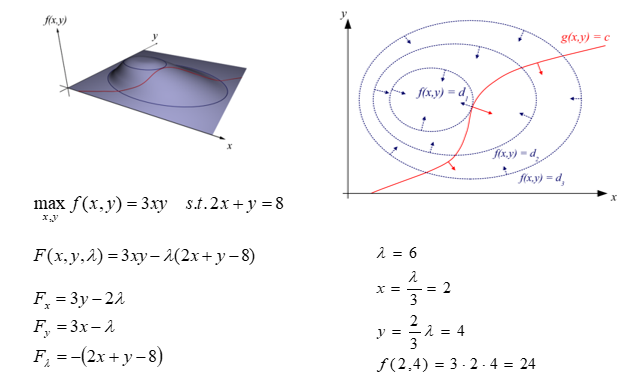


**拉格朗日乘数法：**

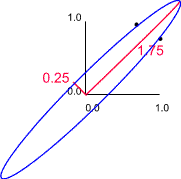
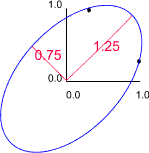


**求函数在满足 下的条件极值，可以转化为函数**

**的无条件极值问题。**



**PCA缺点：会有信息丢失**

Eg:  

相比第一个图，对第二个图进行PCA 丢失的信息会比图1要多，且丢失的信息有可能是重用的信息，因此PCA的使用不是在任何情况下都适合。

**特征值与奇异值：**

**REFER:** [**https://blog.csdn.net/xiaocong1990/article/details/54909126**](https://blog.csdn.net/xiaocong1990/article/details/54909126)

**特征值：**

如果说一个向量v是方阵A的特征向量，将一定可以表示成下面的形式：

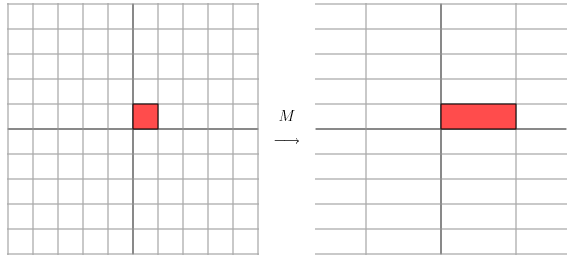
λ就被称为特征向量v对应的特征值，一个矩阵的一组特征向量是一组正交向量。特征值分解是将一个矩阵分解成下面的形式：

其中Q是这个矩阵A的特征向量组成的矩阵，Σ是一个对角阵，每一个对角在线的元素就是一个特征值。

一个矩阵其实就是一个线性变换，因为一个矩阵乘以一个向量后得到的向量，其实就相当于将这个向量进行了线性变换，如下图所示：

http://images.cnblogs.com/cnblogs_com/LeftNotEasy/201101/20110119222632500.png

它其实对应的线性变换是下面的形式：

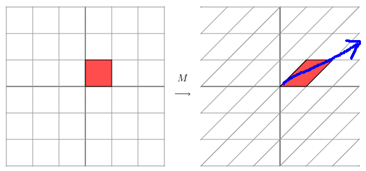


 因为这个矩阵M乘以一个向量(x,y)的结果是：

http://images.cnblogs.com/cnblogs_com/LeftNotEasy/201101/201101192226334470.png

上面的矩阵是对称的，所以这个变换是一个对x，y轴的方向一个拉伸变换（每一个对角在线的元素将会对一个维度进行拉伸变换，当值>1时，是拉长，当值<1时时缩短）。

当矩阵不是对称的时候，假如说矩阵是下面的样子：

http://images.cnblogs.com/cnblogs_com/LeftNotEasy/201101/201101192226336454.png 

这其实是在平面上对一个轴进行的拉伸变换（如蓝色的箭头所示），在图中，蓝色的箭头是一个最主要的变化方向（变化方向可能有不止一个），如果我们想要描述好一个变换，那我们就描述好这个变换主要的变化方向就好了。反过头来看看之前特征值分解的式子，分解得到的Σ矩阵是一个对角阵，里面的特征值是由大到小排列的，这些特征值所对应的特征向量就是描述这个矩阵变化方向（从主要的变化到次要的变化排列）。

总结：

特征值分解可以得到特征值与特征向量，特征值表示的是这个特征到底有多重要，而特征向量表示这个特征是什么，可以将每一个特征向量理解为一个线性的子空间。

局限：**特征值变换的矩阵必须是方阵**

**奇异值分解：**

奇异值分解是一个能适用于任意的矩阵（M, N）的一种分解的方法，不局限于方阵。

假设A是一个N \* M的矩阵,那么得到的矩阵

U是一个N \* N的方阵（里面的向量是正交的，U里面的向量称为左奇异向量），

Σ是一个N \* M的矩阵（除了对角线的元素都是0，对角在线的元素称为奇异值），

VT(V的转置)是一个N \* N的矩阵，里面的向量也是正交的，V里面的向量称为右奇异向量）

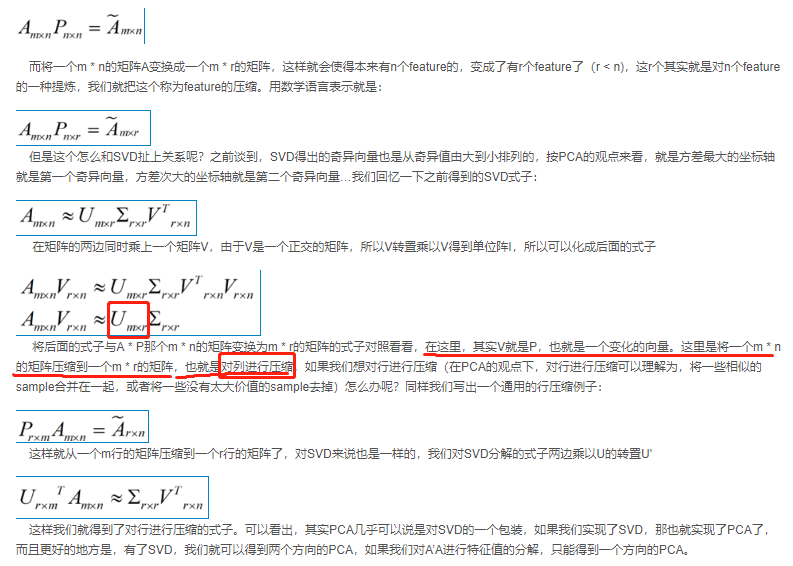
可以用前r大的奇异值来近似描述矩阵:

 r是一个远小于m、n的数

**奇异值与主成分分析PCA:**

PCA的问题其实是一个基的变换，使得变换后的数据有着最大的方差。方差的大小描述的是一个变量的信息量，我们在讲一个东西的稳定性的时候，往往说要减小方差，如果一个模型的方差很大，那就说明模型不稳定了。但是对于我们用于机器学习的数据（主要是训练数据），方差大才有意义，不然输入的数据都是同一个点，那方差就为0了，这样输入的多个数据就等同于一个资料了.

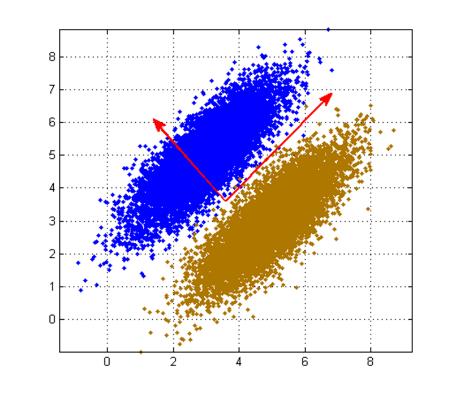
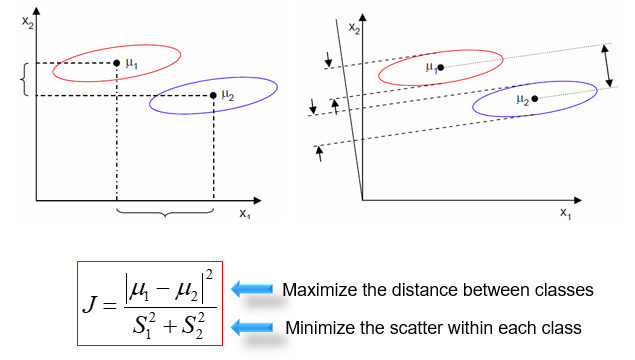
 PCA的全部工作简单点说，就是对原始的空间中顺序地找一组相互正交的坐标轴，第一个轴是使得方差最大的，第二个轴是在与第一个轴正交的平面中使得方差最大的，第三个轴是在与第1、2个轴正交的平面中方差最大的，这样假设在N维空间中，我们可以找到N个这样的坐标轴，我们取前r个去近似这个空间，这样就从一个N维的空间压缩到r维的空间了，但是我们选择的r个坐标轴能够使得空间的压缩使得数据的损失最小。



针对一行一样本，一列一特征的数据：

对列压缩即对特征进行了降维，由n维降至r维，故是对特征进行降维后的m行样本的数据；

**PCA是unsupervised 方法， 处理无卷标数据降维**，不考虑label，即class information。如下图左所示，如果两个样本有明显的label，则可能本来能分开的数据，采用PCA后却无法分开。

由于PCA实际计算中只涉及到输入数据概率密度分布函数（Pdf）的二阶特性（协方差矩阵），所以解出的各主分量只互相正交（不相关），但并不满足相互独立。而且信号的大部分重要特征往往包含在Pdf的高阶统计特性中，所以只有多变量观测数据是由高斯分布的源信号构成，PCA方法才有效。

非线性PCA（NLPCA）即将高阶累积量引入标准的PCA中，是由芬兰学者Karhunen和Oja首先提出并将其应用于ICA。它的可以完成对输入信号的盲分离。高阶累积量是以隐含的方式引入计算的，采用自适应迭代方法便于工程实现。**标准的PCA基于信号的协方差矩阵仅能处理高斯信号**，而NLPCA可以处理非高斯信号。

总结：

1、处理数据集是线性的，对于非线性问题，有核化PCA可以进行处理，该方法主要是利用非线性权值对线性PCA进行扩展。

2、PCA是使用中值和方差来进行概率分布的描述，而这种方法仅适用于样本呈指数分布的情形，所以PCA假设数据是呈高斯分布的。

3、PCA隐含着假设：数据具有较高的信噪比，所以具有最大方差的方向被作为主元，而方差小的方向被作为噪声，这是由低通滤波的特性决定的。两个服从高斯分布的随机变量，不相关和独立为充要条件

4、主元正交：PCA假设主元之间是正交的，这样的假设有利于采用线性代数的方法进行求解。

**LDA (Linear Discriminant Analysis) 线性判别分析**

处理**有卷标数据降维**问题的方法可以用 LDA (Linear Discriminant Analysis)，在降维的同时，尽可能多地保留分类判别信息class discriminant information。

LDA降维后投影**可分性的度量：Fisher Criterion**

如上图右所示：

J的分子衡量的是不同类的中心之间的距离，越大越好，越大则不同类距离越远；

J的分母衡量的是类的散布情况，越小越好，越小则类内部的数据越集中。

同一种类别数据的投影点尽可能的接近，而不同类别的数据的类别中心之间的距离尽可能的大

因此，J越大，可分性越好

***LDA目标： max J***

向量基础：

1. 向量内积：A与B的内积等于A到B的投影长度乘以B的模

设向量B的模为1，则A与B的内积值等于A向B所在直线投影的向量长度， 即，该值即为向量A在基向量B上的投影

1. 要准确描述向量x，首先要确定一组基（w1,w2），然后给出x分别在一组基所在的各个直线上的投影值

**坐标系转换**：

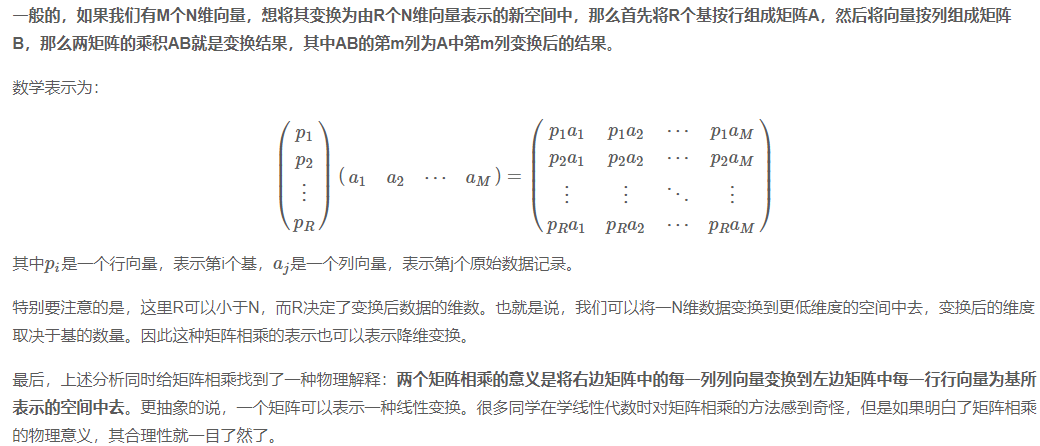
=====> 详细参考 ： <http://blog.codinglabs.org/articles/pca-tutorial.html>

已知原基向量组成的坐标系中的向量为X, 新的坐标系基向量为W，W=[w1, w2],则

转换后的新坐标系中的向量为, 则

(w1,w2为一组基向量)

即：将(x1, x2)变换为新基上的坐标，就是用(x1, x2)与第一个基做内积运算，作为第一个新的坐标分量，然后用(x1, x2)与第二个基做内积运算，作为第二个新坐标的分量



**LDA和PCA对比(refer:** [**https://blog.csdn.net/u012679707/article/details/80529252**](https://blog.csdn.net/u012679707/article/details/80529252)**)**

    LDA用于降维，和PCA有很多相同，也有很多不同的地方，因此值得好好的比较一下两者的降维异同点。

　　　首先我们看看相同点：

　　　　1）两者均可以对数据进行降维。

　　　　2）两者在降维时均使用了矩阵特征分解的思想。（**求特征值、特征向量**）

　　　　3）两者都假设**数据符合高斯分布**。

　　　我们接着看看不同点：

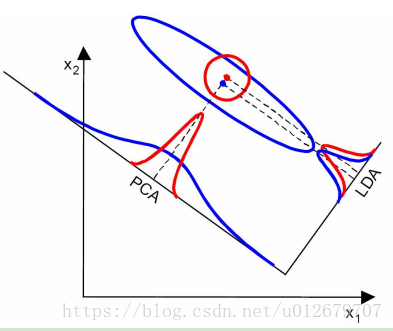
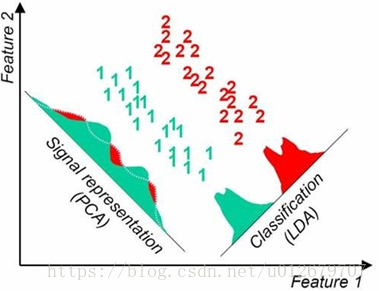
　　　　1）LDA是有监督的降维方法，而PCA是无监督的降维方法

　　　　2）LDA降维最多降到类别数k-1的维数，而PCA没有这个限制。

　　　　3）LDA除了可以用于降维，还可以用于分类。

　　　　4）LDA选择分类性能最好的投影方向，而PCA选择样本点投影具有最大方差的方向。

　　　　这点可以从下图形象的看出，在不同资料分布下LDA和PCA降维的优势不同。二者各有优缺。

**LDA算法的主要优点有**：

　　　　1）在降维过程中可以使用类别的先验知识经验，而像PCA这样的无监督学习则无法使用类别先验知识。

　　　　2）LDA在样本分类信息依赖均值而不是方差的时候，比PCA之类的算法较优。

　 LDA算法的主要缺点有：

　　　　1）LDA不适合对非高斯分布样本进行降维，PCA也有这个问题。

　　　　2）LDA降维最多降到类别数k-1的维数，如果我们降维的维度大于k-1，则不能使用LDA。当然目前有一些LDA的进化版算法可以绕过这个问题。

　　　　3）LDA在样本分类信息依赖方差而不是均值的时候，降维效果不好。

　　　　4）LDA可能过度拟合数据。

**LDA算法流程**：

输入：数据集D={(x1,y1),(x2,y2),...,((xm,ym))}D={(x1,y1),(x2,y2),...,((xm,ym))},其中任意样本xixi为n维向量，yi∈{C1,C2,...,Ck}yi∈{C1,C2,...,Ck}，降维到的维度d。

　　　　输出：降维后的样本集$D′$

　　　　1) 计算类内散度矩阵Sw

　　　　2) 计算类间散度矩阵Sb

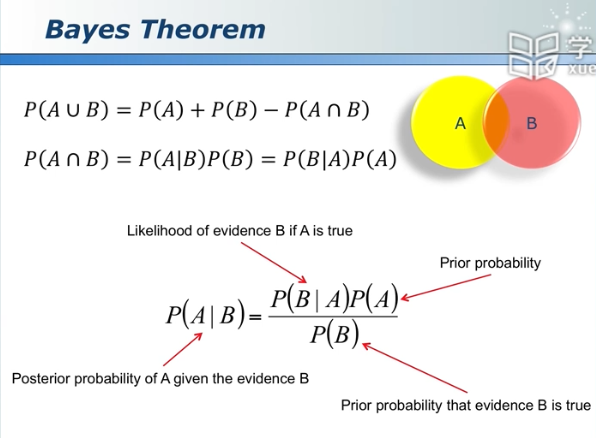
　　　　3) 计算矩阵Sw−1Sb

　　　　4）计算Sw−1Sb的最大的d个特征值和对应的d个特征向量(w1,w2,...wd)，得到投影矩阵W (python: eigvals, eigvects = np.linalg.eig(Sw−1Sb))

　　　　5)  对样本集中的每一个样本特征xi,转化为新的样本zi=WTxi

　　　　6) 得到输出样本集D′={(z1,y1),(z2,y2),...,((zm,ym))}

**Bayes and DecisinonTree**



Eg：

* H=“Having a headache”
* F=“Coming down with flu”
* P(H)=1/10; P(F)=1/40; P(H|F)=1/2
* What does this mean?
* One day you wake up with a headache …
* Since 50% flu cases are associated with headaches …
* I must have a 50-50 chance of coming down with flu! 🡪 wrong!!

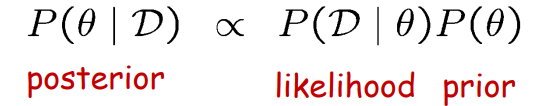
Truth is :

**Naïve Bayes Classifier**（**MAP: Maximum A Posterior**）

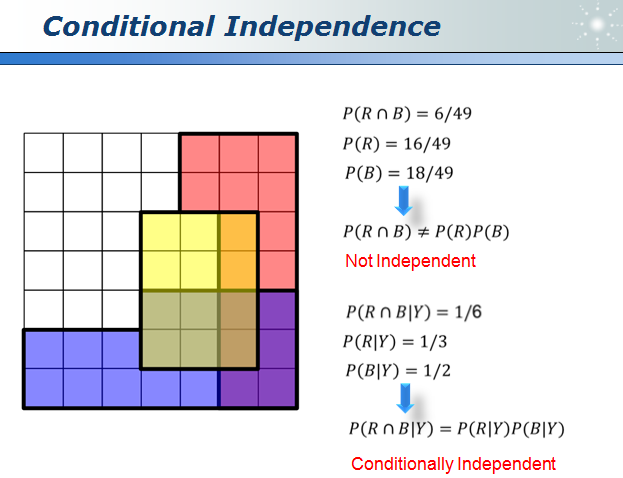
Assumming: 服从条件独立(Conditionally Independent)，即发生的条件下，a1,a2,……,an之间是相互独立的，可近似为：

**最大后验概率估计MAP ：**

**最大似然估计MLE:**

MAP & MLE: 

* 频率学派 - Frequentist - Maximum Likelihood Estimation (MLE，最大似然估计)
* 贝叶斯学派 - Bayesian - Maximum A Posteriori (MAP，最大后验估计)



根据贝叶斯公式， 事件A和事件B发生（A和B不一定相互独立）的条件概率关系满足：

可得：

上一行公式中，其中，

P(A,B)：事件A和事件B的联合概率，也可表示为P(AB)或P(A);

P(A)：事件A发生的先验概率，Prior probability that evidence B is true

P(B)：事件B发生的先验概率，Prior probability that evidence B is true

P(B|A)：事件A发生情况下B发生的似然，Likelihood of evidence B if A is true

P(A|B)：事件B发生情况下A发生的后验概率，Posterior probability of A given evidence B

1）假设事件A和事件B相互独立（即Independent），则有：

可得：

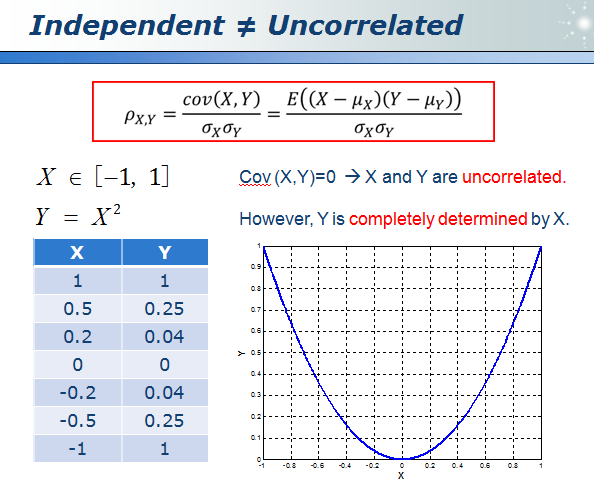
--- Independent

1. 假设事件A和B在条件G存在的前提下相互独立（即Conditionally Independent），则有：

可得：

**---** Conditionally Independent

独立并不等于不相关***Independent ≠ Uncorrelated***，如下例：

****

**Estimating – 离散属性**

**Laplace（拉普拉斯平滑）：**

其中：

代表第j个特征的第m个选择，代表第j个特征可选的特征个数

比如:

可选j特征范围为[‘帅’，‘富’，‘高’，‘矮’，‘’穷，‘挫’]，aj2表示j属性的第2个选择，即aj2=‘富’；

j特征范围为[‘’帅，‘富’，‘高’，‘矮’，‘’穷，‘挫’]，则j特征可选的特征个数为6, 即Sj=6

**注意：**

由于Navie Bayes中后验概率的计算是所有概率的乘法运算，如果由于样本量不足导致其中一项属性的概率为0，则会导致最后计算结果为0，因此引入laplace平滑来避免0值的出现。

实际应用：文本分类、推荐系统等

**Decision Making**

* **ID3（基于信息增益）**
* **C4.5（基于信息增益比）**
* **CART（gini指数）**

Simpler trees are generally preferred

信息熵：

信息增益：

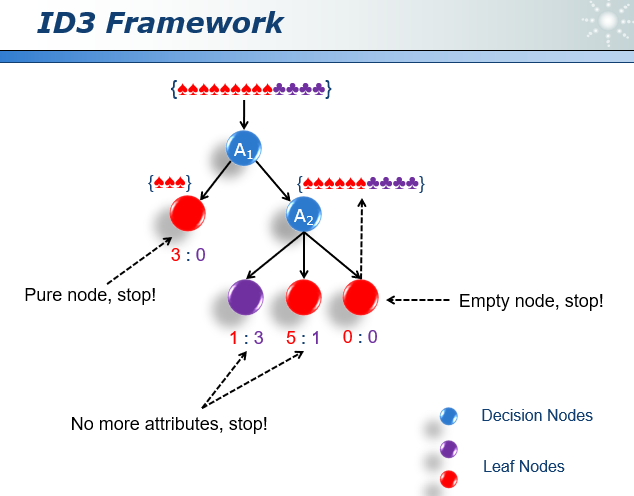
**例如：**

**ID3 （Iterative Dichotomizer 3）**

**ID3采用信息增益作为节点划分指标**

----> How to build a shortest tree from a dataset?

* **ID3(Examples, Target\_attribute, Attributes)**
  + Create a *Root* node for the tree.
  + If *Examples* have the same target attribute T, return *Root* with label=T.
  + If *Attributes* is empty, return *Root* with label=the most common value of *Target\_attribute* in *Examples*.
  + A 🡨 the attribute from *Attributes* that best classifies *Examples*.
  + The decision attribute for *Root* 🡨 A.
  + For each possible value vi of A
    - Add a new tree branch below *Root*, corresponding to A= vi.
    - Let *Examples* (*vi*) be the subset of Examples that have value vi for A.
    - If *Examples* (*vi*) is empty
      * Below this new branch add a leaf node with label=the most common value of *Target\_attribute* in *Examples*.
    - Else below this new branch add the subtree
      * **ID3(Examples(vi), Target\_attribute, Attributes-{A})**
* Return *Root*



如果树的深度过大，会导致过拟合OverFitting,解决方法如下：

1. Stop growing the tree earlier.（提前终止树生长）
2. Allow the tree to overfit the data and then post-prune the tree（**剪枝**）

Validation Error由下降转为上升的拐点处，停止剪枝

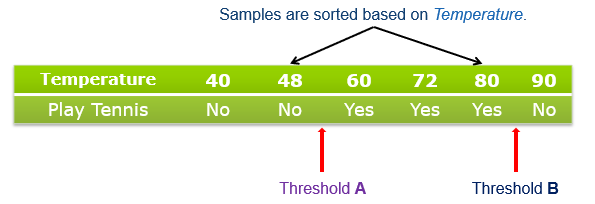
增益率：

判定树算法C4.5采用增益率作为划分标准。

例如，将顾客ID作为节点划分，能产生更纯的划分, ID作为节点能产生更大的信息增益，但却不是一个有预测性的属性,因为每一个ID只有1个属性值, 具有唯一性。即使在不太极端情形下，也不会希望产生大量输出的测试条件，因为输出数量越大，与每个划分相关联的记录太少，以致不能作出可靠的预测。

熵和Gini指针在划分时不考虑属性测试条件产生的输出数量，增益率在信息增益的基础上将输出数量也考虑了进去。

连续属性的划分：



**Neural Networks**

**Support Vector Machines**

**线性支持向量机：**

一个线性分类器的决策边界或划分超平面可以写成如下形式：

任何位于决策边界上的样本如 **和** 都必须满足如下公式：

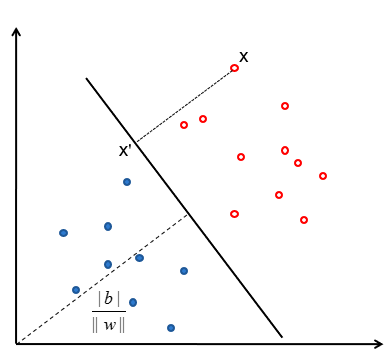
两方程相减得：

因为 **和** 都在决策边界上, 是一个平行于决策边界的向量，由于向量点击的结果为0， 则的方向必然是垂直于决策边界。

称为法向量，决定了超平面的方向，b为位移项（截距项），决定了超平面与原点之间的距离，显然，划分超平面可被法向量和位移b决定，下面将该平面记为（w,b），则**样本中任意到超平面（w, b）的距离**为：

原点(0,0)到该超平面的距离为：

推导过程如下：



（决策边界上的点一定满足： ）

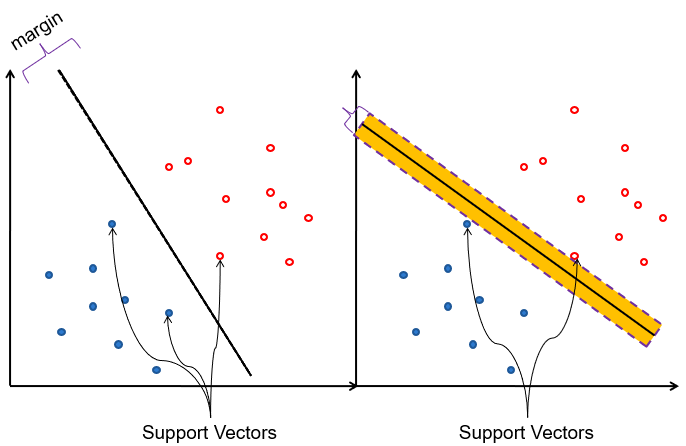
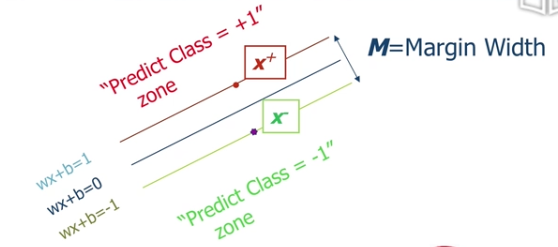
点到超平面的距离：   


间隔（Margin）：

两个异类支持向量到超平面的距离之和。依赖于分离超平面的法向量w, 等于 2/||w||。 法向量指向的一侧为正类，另一侧为负类。

支持向量(Support Vectors)：

线性可分情况下，训练数据集样本点中与分离超平面距离最近的样本点的实例称为支持向量（即间隔边界上的点）。

**决定分离超平面时，只有支持向量起作用，其他实例点并不起作用**，移动或删除其他实例点，解都不会改变，正是由于支持向量在确定超平面中的决定作用，所以将该分类模型称为支持向量机。

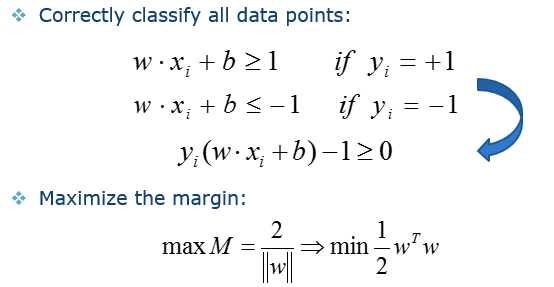
Margin计算如下（上图右）：

Margin由和两个超平面之间的距离给定，两个超平面上任一点到决策边界的距离为：， 则

线性支持向量机本质上与感知器一一样，但SVM增加了一个要求：SVM的决策边界的margin必须是最大的。即Max Margin.

令， 上式等价于：

SVM目标函数优化问题



SVM优化函数和约束条件：

**Min:**

**Subject to:**

**线性可分：SVM优化函数求解 ------- 拉格朗日乘数:**

<https://blog.csdn.net/BIT_666/article/details/79865225?utm_source=blogxgwz2>

SVM目标函数和约束条件：

对每条约束添加拉格朗日乘子αi≥0，则该问题的拉格朗日函数可写为：

其中α=(α1,α2,...,αm)，令L(w,b,α)对w和b求偏导为0可得:

将上式带入拉格让日函数中w和b消去:

从而得到对偶问题：

解出α后，带入w和b关于α的公式即可求出。

求解max(L)的问题省略(二次规划问题)，可见上文连结

**线性不可分情况：SVM**

Margin --> soft margin:

允许一定训练错误的决策边界，即决策边界不再满足 给定的所有约束。

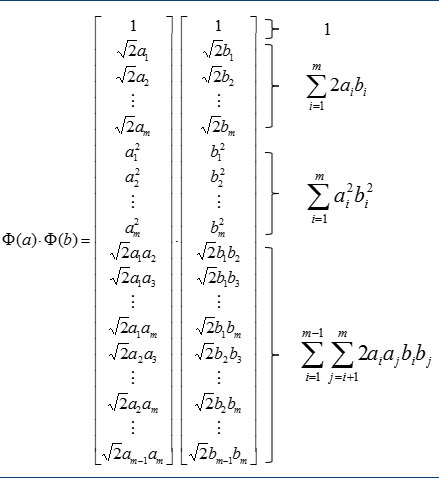
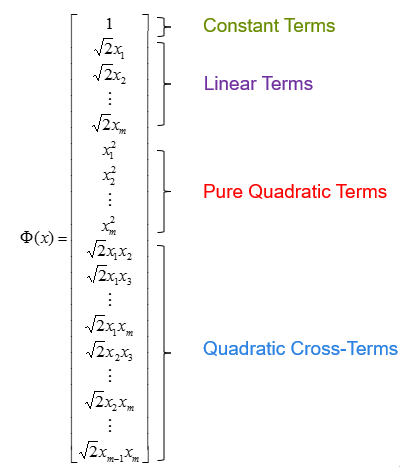
在优化约束中引入松弛变数(slack variable) :

)

过程：

**非线性支持向量机：**

***Quadratic Basis Functions***



将x映射为 向量点乘运算:

可得：  **（O(m) 和O(M2)）**

映射后的维度：

核函数：

使用原属性集计算变换后的空间中的相似度的函数K成为核函数（Kernel Function）

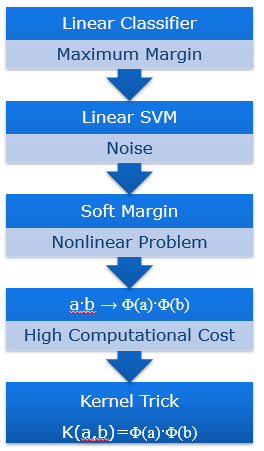
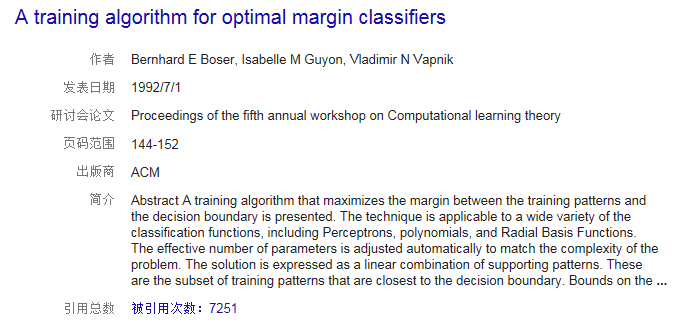
升维变换后线性可分，但是升维后空间计算的复杂度高，利用核函数可以在升维后线性可分的前提下，利用原属性集计算变换空间的相似度。如：(a⋅b+1)^2=Φ(a)⋅Φ(b)

常用核函数：

Polynomial中可以映射到md/2的维度，Gaussian可以映射到无穷维，超高纬度下的计算存在维度灾难，通过核函数可以直接在原空间计算高纬度空间的相似度。

核技术有助于处理如何实现非线性SVM的一些问题：

1. 非线性SVM中使用的核函数必须满足Mercer定理的数学原理（Mercer原理确保核函数总是可以用某高纬空间中两个输入向量的点积表示）。SVM核变换后的空间成为再生核希尔伯特空间（RKHS）；
2. 相对于使用变换后的属性，使用核函数计算点积的开销更小；
3. 计算是在原空间中进行，维灾难问题就可以避免。

参考文献资料：

N. Cristianini and J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press, 2000.

C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition”. Data Mining and Knowledge Discovery, vol. 2, pp. 121-167, 1998.

H. Lodhi et al., “Text Classification Using String Kernels”. The Journal of Machine Learning Research, vol. 2, pp. 419-444, 2002.

Online Resources

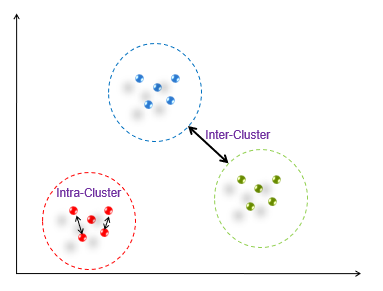
http://www.kernel-machines.org/

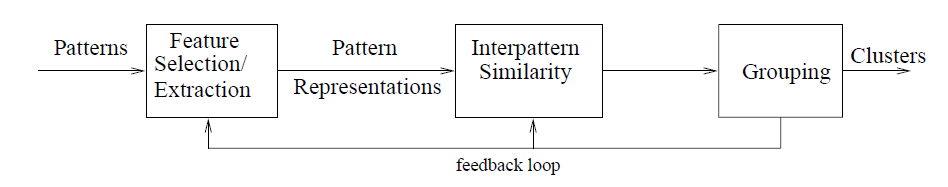
http://www.support-vector-machines.org/

http://www.tristanfletcher.co.uk/SVM%20Explained.pdf

<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

**Clustering**





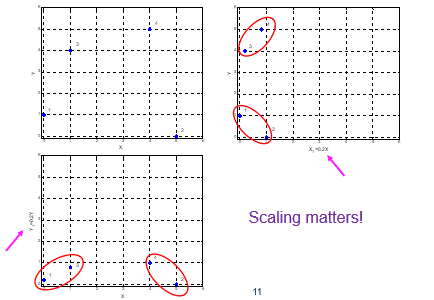
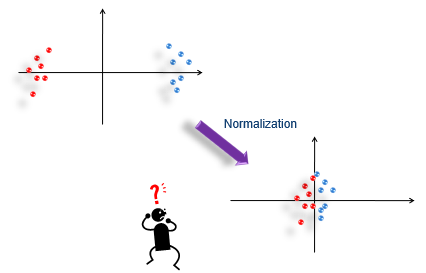
***Requirements***

* Scalability
* Ability to deal with different types of attributes
* Ability to discover clusters with arbitrary shape (任意形状的资料，如S型分布的)
* Minimum requirements for domain knowledge
* Ability to deal with noise and outliers
* Insensitivity to order of input records
* Incorporation of user-defined constraints
* Interpretability and usability

Practical Considerations:

**Scalar matters!!!**

坐标缩放或归一化处理会影响到聚类的结果，如下图左1，子图2为将子图1资料的x轴缩放后的聚类结果，子图3是将子图1资料的y轴缩放后的聚类结果；两者缩放标准不一样导致聚类结果完全不同。

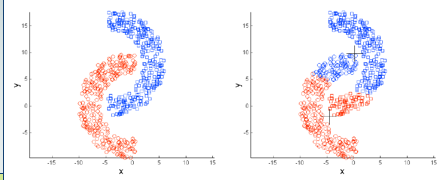
 

***Evaluation***聚类评判标准：

方法1：

所有簇 簇内的点到类中心的距离之和 的总和。相当于衡量的是簇内的数据散布情况。

Attention!! 该evaluation实际包含一个隐藏的条件：假设数据类似呈球状分布。

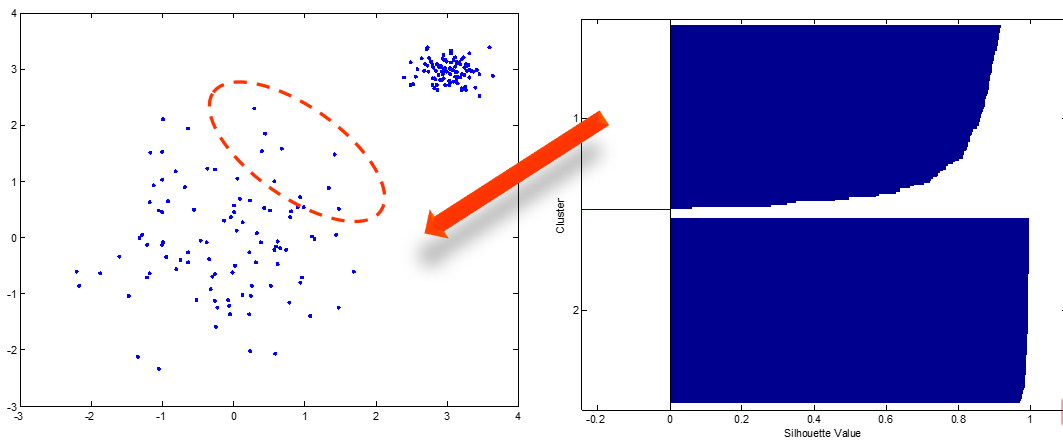
例如：图1中使用方法1的evaluation，结果更倾向于图1右边的聚类；

图2中使用方法1，结果更倾向于图2右边的聚类。而实际上，这两组数据更合适的聚类结果应该是图左边的情况。

方法2：Silhouette（推荐）轮廓系数--表示聚类划分内密外疏的程度

* A method of interpretation and validation of clusters of data.
* A succinct graphical representation of how well each data point lies within its cluster compared to other clusters.
* a(*i*): average dissimilarity of *i* with all other points in the same cluster（一个点与同一个簇内的其他点之间的平均距离）
* b(*i*): the lowest average dissimilarity of *i* to other clusters（一个点与其他簇内的点之间的平均距离）
* 轮廓系数的值介于[-1,1]区间，1表示完美聚类，-1表示错误聚类，0表示聚类重迭。

注：s(i)有可能为负数，即簇A内的某一个点在A的边缘，且距离簇B比较近，如下图



K-Means

Steps of K-Means algorithm

* Determine the value of K.
* Choose K cluster centres randomly.
* Each data point is assigned to its closest centroid.
* Use the mean of each cluster to update each centroid.
* Repeat until no more new assignment.
* Return the K centroids.

J. MacQueen (1967): "Some Methods for Classification and Analysis of Multivariate Observations", Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, vol.1, pp. 281-297.

**优缺点：**

* Pros
  + Simple and works well for regular disjoint clusters.（对于球状分布数据或互斥资料效果很好）
  + Converges relatively fast.（快可至5~6步即收敛）
  + Relatively efficient and scalable *O*(*t*·*k*·*n*)
  + *t*: iteration; *k*: number of centroids; *n*: number of data points
* Cons
  + Need to specify the value of K in advance.

Difficult and domain knowledge may help.（需要先验知识）

* + May converge to local optima.

In practice, try different initial centroids.

* + May be sensitive to noisy data and outliers.

Mean of data points …（均值对异常点和噪声很敏感）

* + Not suitable for clusters of Non-convex shapes(凸面)

Sequential Leader Clustering

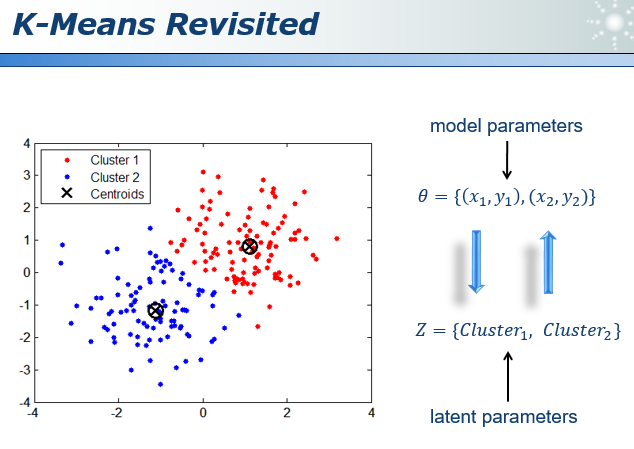
* A very efficient clustering algorithm.
  + No iteration
  + A single pass of the data
* No need to specify K in advance.
* Choose a cluster threshold value.
* For every new data point:
  + Compute the distance between the new data point and every cluster's centre.
  + If the minimum distance is smaller than the chosen threshold, assign the new data point to the corresponding cluster and re-compute cluster centre.
  + Otherwise, create a new cluster with the new data point as its centre.
* Clustering results may be influenced by the sequence of data points.

***Gaussian Mixture混合高斯模型***

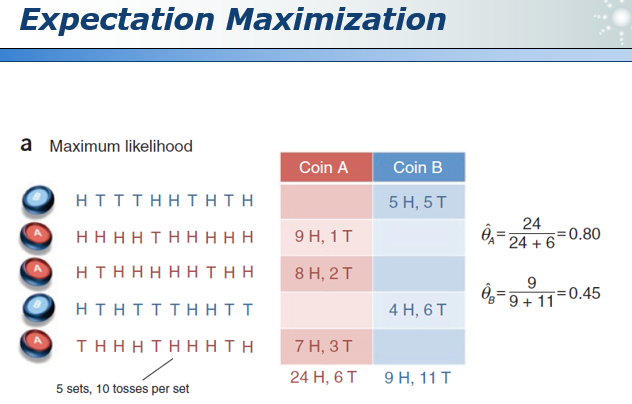
***K-Means Revisited***

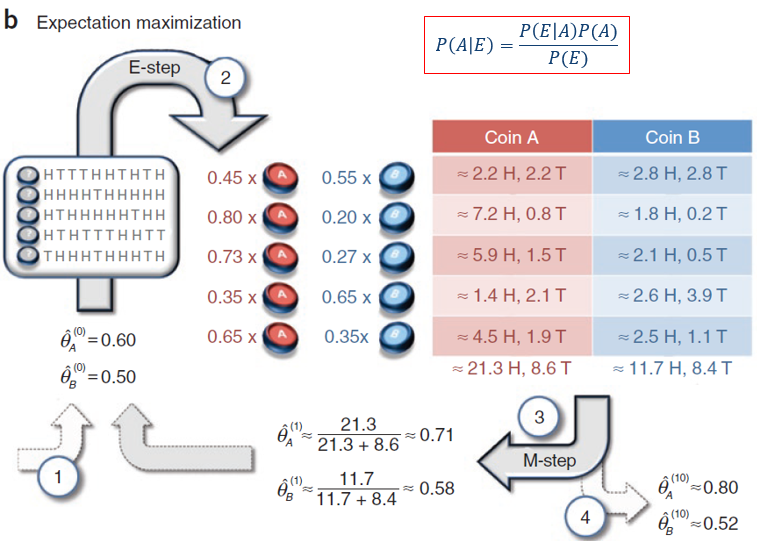
model parameters：

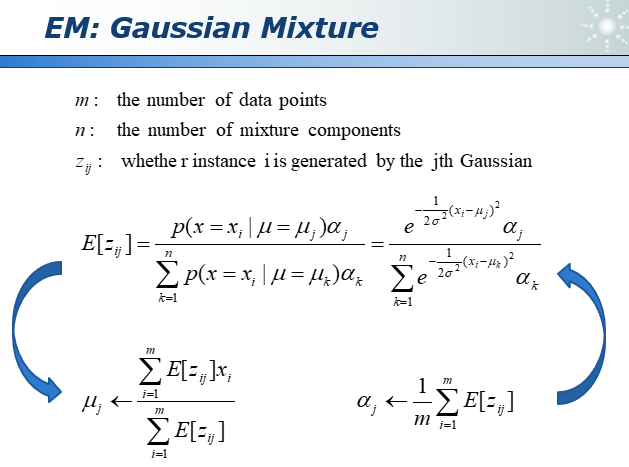
latent parameters：



**EM算法*Expectation Maximization*：**





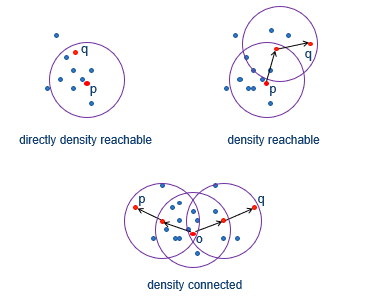


**Density Based Methods**

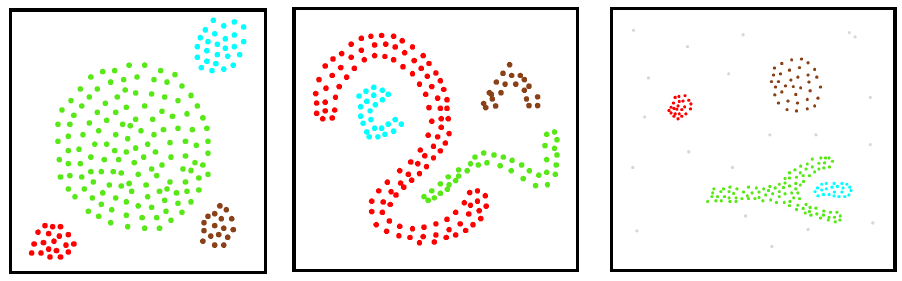
* Generate clusters of arbitrary shapes.(任意形状数据聚类)
* Robust against noise.（鲁棒性好）
* No K value required in advance.（不需要提前设定K值）
* Somewhat similar to human vision.

**DBSCAN:**

* Density-Based Spatial Clustering of Applications with Noise
* Density: number of points within a specified radius
* Core Point: points with high density
* Border Point: points with low density but in the neighbourhood of a core point
* Noise Point: neither a core point nor a border point



* A cluster is defined as the maximal set of density connected points.(对螺旋形/S形等都有很不错的聚类效果, 如下图s)
* Start from a randomly selected unseen point P.
* If P is a core point, build a cluster by gradually adding all points that are density reachable to the current point set.
* Noise points are discarded (unlabelled).



**Hierarchical Clustering（分层聚类）**

* Bottom-up Method
* Assign each data point to a cluster.
* Calculate the proximity matrix.
* Merge the pair of closest clusters.
* Repeat until only a single cluster remains.
* How to calculate the distance between clusters?
* Single Link
  + Minimum distance between points
* Complete Link
  + Maximum distance between points

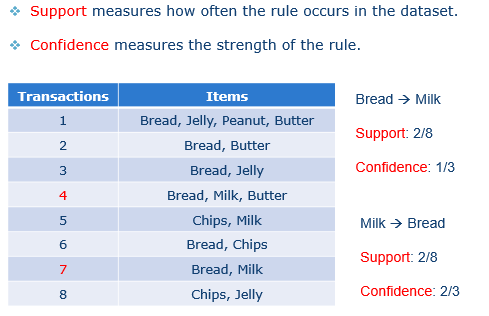
***Reading Materials***

* Text Books
  + R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification,* Chapter 10, 2nd Edition, John Wiley & Sons.
  + J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Chapter 8, Morgan Kaufmann.
* Survey Papers
  + A. K. Jain, M. N. Murty and P. J. Flynn (1999) “Data Clustering: A Review”, *ACM Computing Surveys*, Vol. 31(3), pp. 264-323.
  + R. Xu and D. Wunsch (2005) “Survey of Clustering Algorithms”, *IEEE Transactions on Neural Networks*, Vol. 16(3), pp. 645-678.
  + A. K. Jain (2010) “Data Clustering: 50 Years Beyond K-Means”, *Pattern Recognition Letters*, Vol. 31, pp. 651-666.
  + Online Tutorials
  + <http://home.dei.polimi.it/matteu>
  + cc/Clustering/tutorial\_html
  + http://www.autonlab.org/tutorials/kmeans.html
  + http://users.informatik.uni-halle.de/~hinnebur/ClusterTutorial

**Association Rule**

***Support & Confidence of Association Rule***

* The support of an association rule X🡪Y is the percentage of transactions that contain X and Y.
* The confidence of an association rule X🡪Y is the ratio of the number of transactions that contain {X, Y} to the number of transactions that contain X.
* It can be represented equally as
* Conditional probability: P(Y|X)



***Frequent Itemsets and Strong Rules***

* A frequent (large) itemset is an itemset with support larger than σ.
* A strong rule is a rule that is frequent and its confidence is higher than Φ.

Association Rule Problem

* Given I, D, σ and Φ, to find all strong rules in the form of X🡪Y.

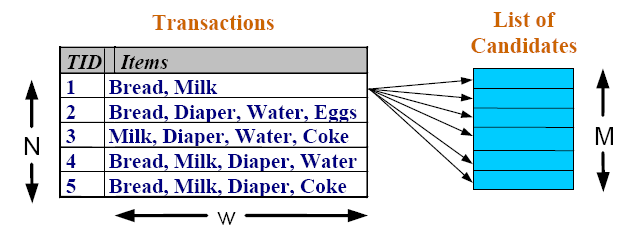
Attention:

A rule with high confidence is not necessarily plausible(貌似可信的).

* + 置信度需要与先验概率对比，若大于阈值的置信度反而小于先验概率，则说明该规则并不可信
  + 两个item出现概率相差很大，比如10个买A的人有8个，只有1个买B，则P(A|B)=100%(买B的人一定买A概率)，买A的人先验概率为P(A)=80%，即置信度P(A|B)>P(A); 但是“买B的人一定买A”这样的说法就没意义，因为B和A并不存在因果关系（又比如，犯罪率上升，且冰淇淋销量也上升，不能认为两者存在因果关系）。

Association **≠** Causality （相关性并不等价于存在因果关系）

***Itemset Calculation***



自由度O(NWM): M = 2d – 1 （-1：减去空集）-- 过于庞大计算量

推荐算法：Apriori Algorithm(数据挖掘十大算法之一)

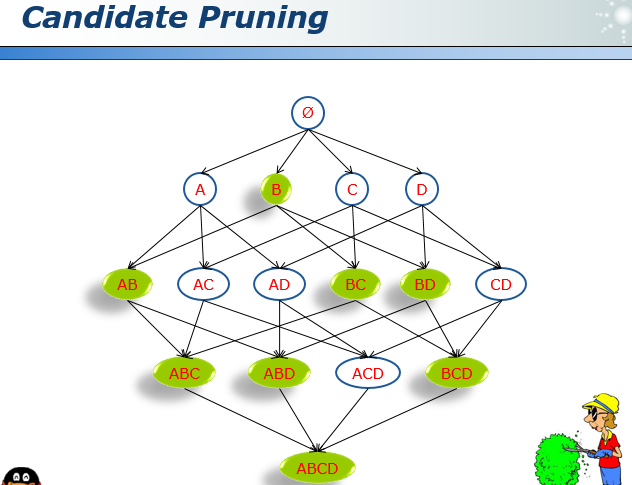
***The Apriori Method***

* One of the best known algorithms in Data Mining
* Key ideas
  + A subset of a frequent itemset must be frequent.
    - {Milk, Bread, Coke} is frequent 🡪 {Milk, Coke} is frequent

（{Milk, Coke}是{Milk, Bread, Coke}的子集）

* + The supersets of any infrequent itemset cannot be frequent.
    - {Battery} is infrequent 🡪 {Milk, Battery} is infrequent

（{Milk, Battery}是{Battery}的超集）



回顾：

子集定义

对于两个集合A与B，如果集合A的任何一个元素都是集合B的元素，我们就说集合A包含于集合B，或集合B包含集合A，也说集合A是集合B的子集。如果集合A的任何一个元素都是集合B的元素，而集合B中至少有一个元素不属于集合A，则称集合A是集合B的真子集。空集是任何集合的子集。 任何一个集合是它本身的子集.空集是任何非空集合的真子集.

超集定义

如果一个集合S2中的每一个元素都在集合S1中，且集合S1中可能包含S2中没有的元素，则集合S1就是S2的一个超集。

Itemset剪枝（pruning）:如上图，如果B是infrequent, 则所有包含B的itemset都不频繁

***General Procedure：***

* Generate itemsets of a particular size.
* Scan the database once to see which of them are frequent.
* Use frequent itemsets to generate candidate itemsets of size=size+1.
* Iteratively find frequent itemsets with cardinality from 1 to k.
* Avoid generating candidates that are known to be infrequent.
* Require multiple scans of the database.
* Efficient indexing techniques such as Hash function & Bitmap may help.

***Apriori Algorithm：***

***C*k**: Candidate itemset of size k（长度为k的itemset）

***L*k**: Frequent itemset of size(长度为k的itemset中的频繁itemset)



L1为频繁项集，

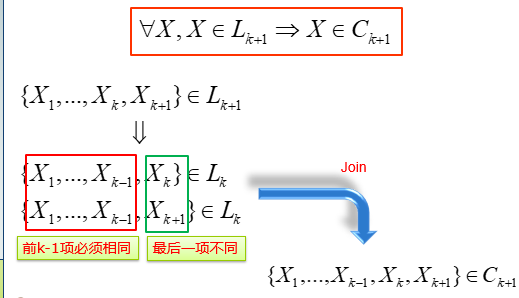
Candidates: 通过L1生成C2, C2中itemsize计数，去掉不频繁项集(即筛选出频繁项集)，生成L2；通过L2生成C3, C3中itemsize计数，去掉不频繁项集(即筛选出频繁项集)，生成L4, 依次类推

（频繁项集的子集一定是频繁项集；不频繁项集的超集一定是不频繁项集）

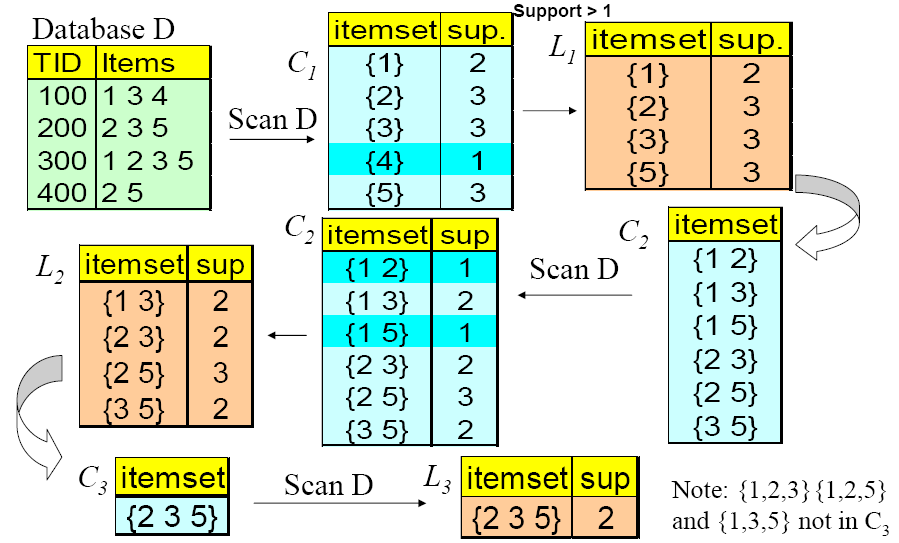
关键：Lk ----> Ck

其中：每一个项集都必须是先排序过的， 升序， ordered list

推导：



举例1：



L1,L2,L3是ZUOYOU 的频繁项集，由于关联规则中的规则至少需要2个item，因此L1去掉，最后筛选出来的频繁项集为L2,L3 {{1 3}, {2 3}, {2 5}, {3 5}, {2 3 5}}

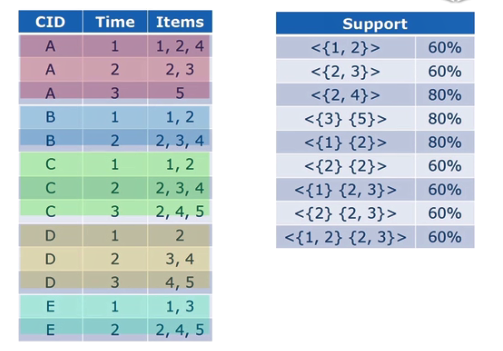
***Sequential Pattern***

* t=<t1 t2 … tm> is a subsequence of s=<s1 s2 … sn> if there exist integers

1 j1<j2< … <jmn such that t1sj1, t2sj2,…, tmsjm.

|  |  |  |
| --- | --- | --- |
| **s** | **t** | **Y/N** |
| <{2, 4} {3, 6, 5} {8}> | <{2} {3, 6} {8}> | Yes |
| <{2, 4} {3, 6, 5} {8}> | <{2} {8}> | Yes |
| <{1, 2} {3, 4}> | <{1} {2}> | No |
| <{2, 4} {2, 4} {2, 5}> | <{2} {4}> | Yes |

***Support of Sequence***



* Core Papers
  + J. Han, J. Pei, Y. Yin and R. Mao (2004) “Mining frequent patterns without candidate generation: A frequent-pattern tree approach”. *Data Mining and Knowledge Discovery*, Vol. 8(1), pp. 53-87.
  + R. Agrawal and R. Srikant (1995) “Mining sequential patterns”. In *Proceedings of the Eleventh International Conference on Data Engineering* (ICDE), pp. 3-14.
  + R. Agrawal and R. Srikant (1994) “Fast algorithms for mining association rules”. In *Proceedings of the 20th International Conference on Very Large Data Bases* (VLDB), pp. 487-499.
  + R. Agrawal, T. Imielinski, and A. Swami (1993) “Mining association rules between sets of items in large databases”. In *Proceedings of the ACM SIGMOD International Conference on Management of Data* (SIGMOD), pp. 207-216.

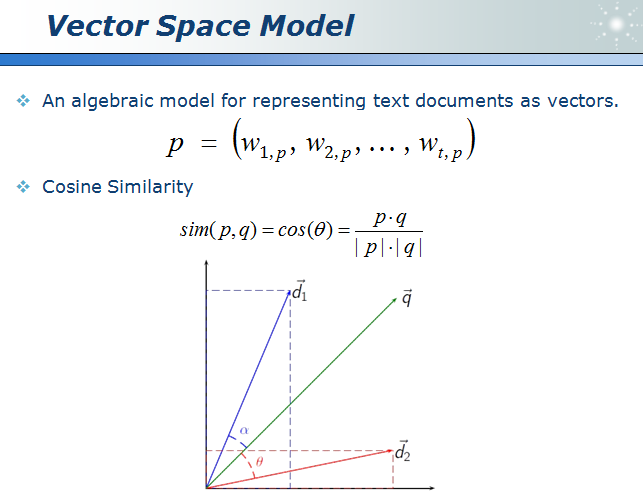
**Recommendation Algorithms**

* TF-IDF
* Vector Space Model
* Latent Semantic Analysis
* PageRank
* Collaborative Filtering
* **Content-Based Filtering**
  + Focuses on the characteristics of items.
  + Recommends items similar to those that a user liked in the past.
* **Collaborative Filtering**
  + Predicts what users will like based on their similarity to other users.
  + Similar to asking the opinions of friends.
  + Does not rely on machine analysable contents.

**Tf-idf 词频-逆文档矩阵**

* Multiple query words

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Doc 1** | **Doc 2** | **Doc 3** | **Doc 4** |
| the | 20 | 10 | 15 | 8 |
| best | 0 | 1 | 0 | 2 |
| car | 3 | 5 | 0 | 0 |



* Synonymy (近义词)
  + Different words, same meaning
  + Car, Vehicle, Automobile
  + Small cosine values 🡪 unrelated
  + Poor recall （正确预测的正样本数 / 实际的正样本数）
* Polysemy（多义词）
  + One word, different meanings
  + Apple Computer vs. Apple Juice
  + Large cosine values 🡪 related
  + Poor precision
* Let’s work in a more informative space.
  + Merge dimensions with similar meanings.
  + Singular Value Decomposition

如何解决上述近义词或多义词或stopwords等的影响---》 隐含语义分析

**Latent Semantic Analysis （LSA）**

**Refer:** [**https://blog.csdn.net/lxy\_ts/article/details/16902337**](https://blog.csdn.net/lxy_ts/article/details/16902337)

其中：

X: M\*N 一行一单词word 一列一文檔doc;

T: M\*r正交矩阵; TTT=E

S: r\*r 对角矩阵;

D: n\*r 正交矩阵; DDT=E

r=rank(X) 秩

TS: (M\*r)(r\*r)🡪(M\*r), M行个word，r列个doc

则，T 是 XXT的特征向量（XXT: X与XT的点乘， dot products）

Rows of TS：(m\*r) (r\*r) 🡪 (m\*r)

表征的是coordinates of terms(terms即words) 单词的坐标

则，D 是XTX的特征向量（XTX: XT与X的点乘， dot products）

Rows of DS：　(n\*r) (r\*r) 🡪 (n\*r)

表征的是coordinates of docments(即文檔) 文档的坐标

分解之后（S表示的是特征值大小）进行k降维，类似PCA，如rank即纬度前k个值对应的T的列向量和D对应的行向量：

取k=2, 即认为T前两列和D前两行保留了X的最大信息量，通过T[:, :1]和D[:1, :]进行X的恢复，得到



TS (m\*2)矩阵，TS矩阵中每一行代表一个单词(index为单词名)，单词的坐标即为TS矩阵中每一行的2个数值，

如第3行单词索引为’user’，对应矩阵坐标为（-1.4，0.3）

上图中，判断两个点之间是否接近不是采用欧氏距离，而是采用余弦相似度，即从原点(0,0)位置与点连直线，看夹角是不是接近来判断两个点是否接近。

给定文档列向量q，求文档坐标：

]

0060

.

0

,

4864

.

0

[

]

0024

.

0

,

1456

.

0

[

]

0

,

0

,

0

,

0

,

0

,

0

,

1

,

0

,

0

,

0

,

0

,

1

[

"

"

1





















*S*

*q*

*q*

*T*

*S*

*q*

*q*

*response*

*human*

*Query*

*T*

*T*

*T*

*k*

*k*

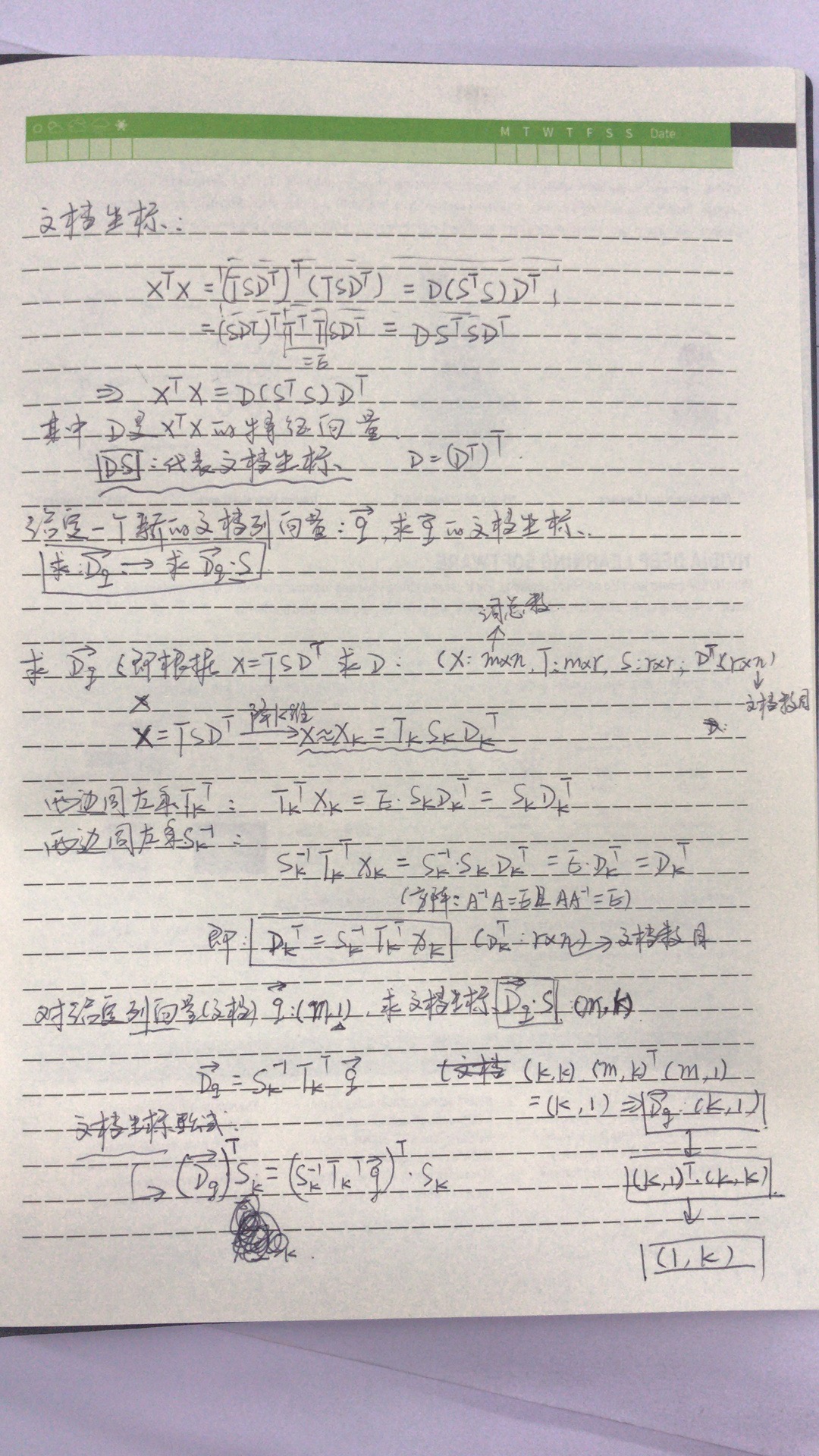
*T*





根据求出的文档坐标，迭代计算与潜在语义空间中的每一个文档的相似度(余弦相似度)来得到和给定查询最相似的文档。

文档坐标推导如下：



***PageRank***

大概是根据指向该连结的连结数和这些连结的pagerank权重作为指标

如图：

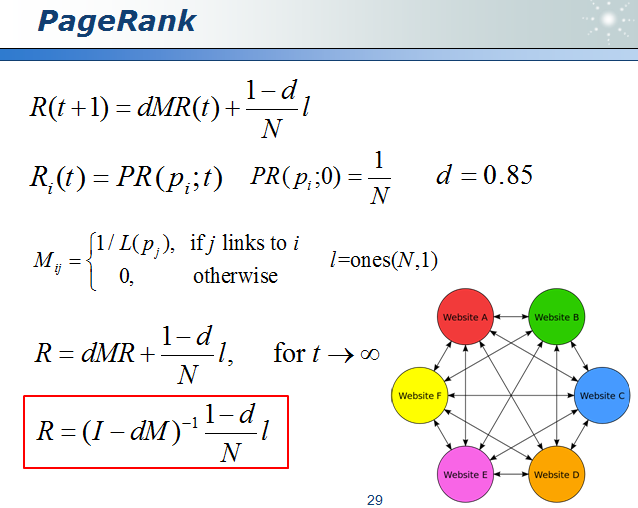
PR(B)/2中分母为2是因为pageB指向A和C2个page, 因此pageB对pageA的pagerank的支持度只有B自身rank值的二分之一

* 即：The importance of vote from a page depends on its own PageRank and the number of outbound links.

其中：

PR(Pj:t)表示t时刻page-j的pagerank值

L(Pj)表示page-j 的outlink总数，即page-j指出的连结的总数



***Collaborative Filtering（CF） 协同过滤***

* User-based CF
* Item-based CF
* Model-based CF

核心思想：根据兴趣或品味相同的其他人的窗体里rately high的内容进行推荐

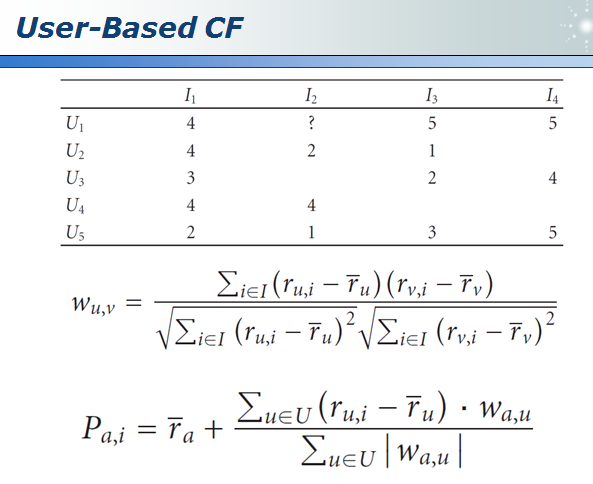
* Core Idea:
  + People get the best recommendation from others with similar tastes.
* Workflow:
  + Creates a rating or purchase matrix.
  + Finds similar people by matching their ratings.
  + Recommends items that similar people rate highly.

缺陷或注意事项：

* + 无可参考对象
  + 恶意打分现象如恶意好评等
  + 冷启动

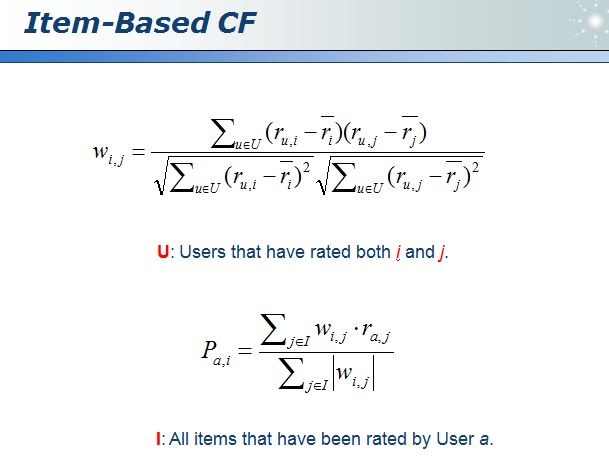
打分矩阵

基于用户的协同过滤：

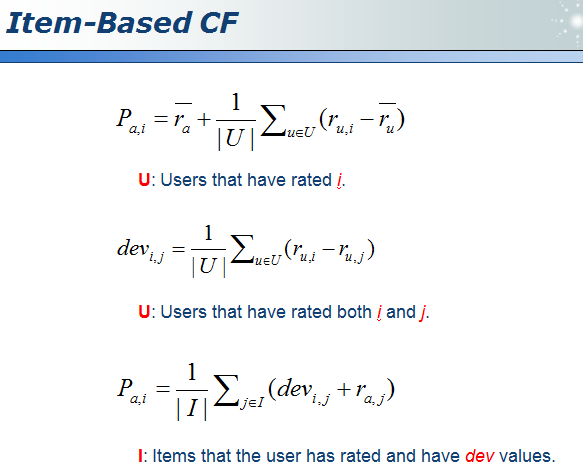


用戶相似度

基于内容的协同过滤：



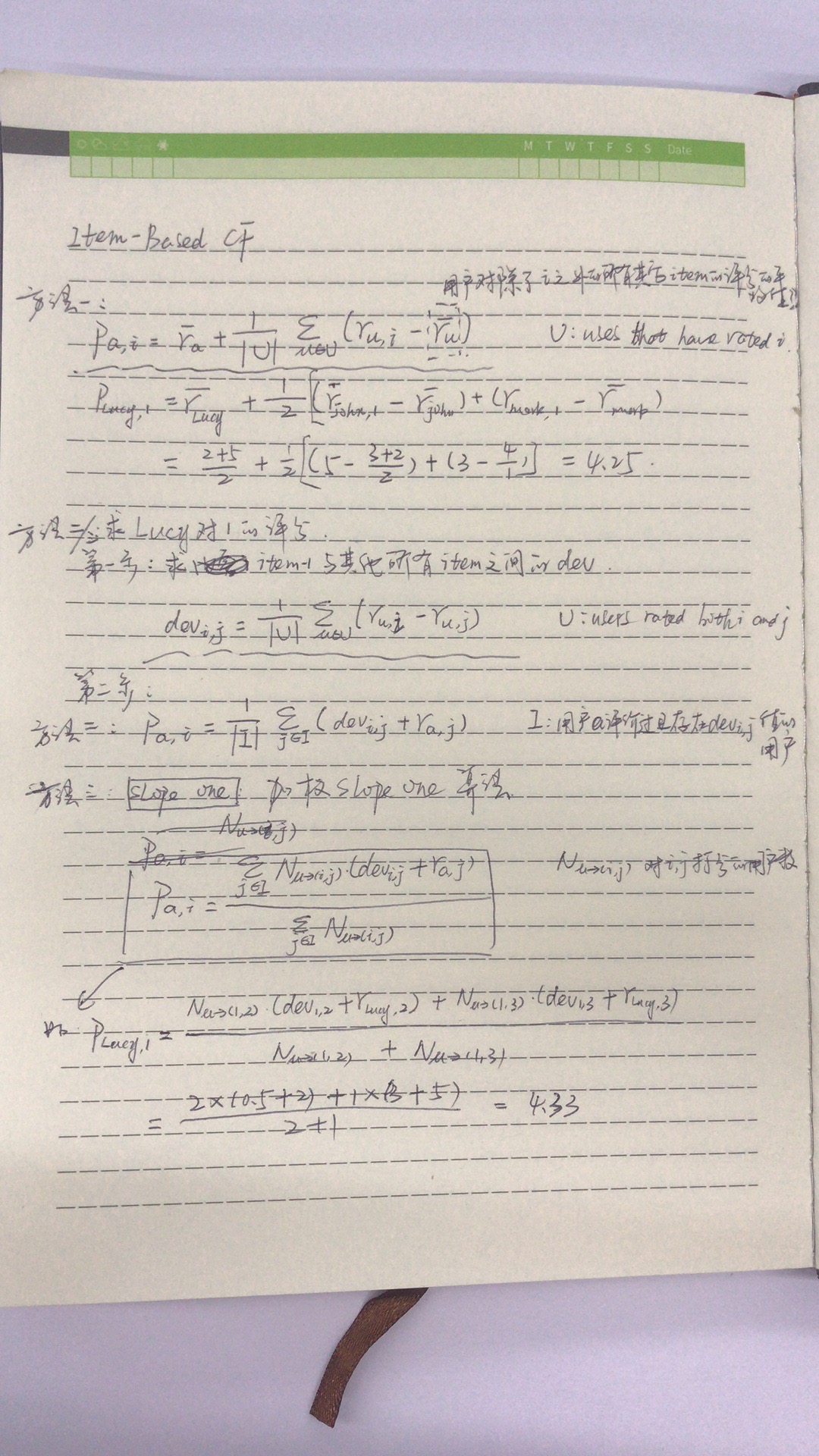
打分可以有多中方式，如下图列出的两种方式就不同



上图公式1中 u对除了item-i 之外的其他所有item的打分的平均分??

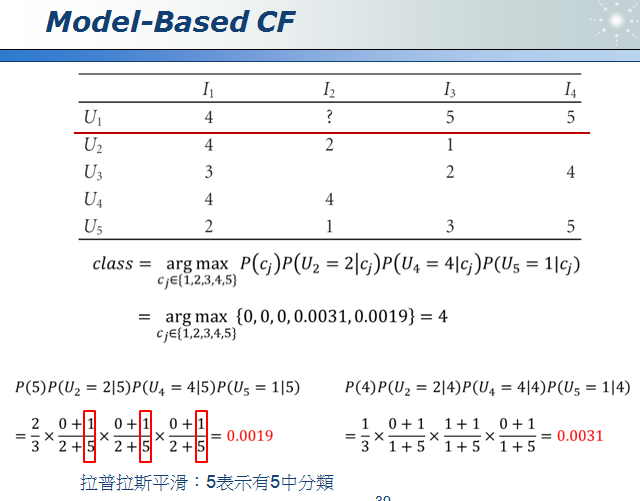
后面的例子是这样的

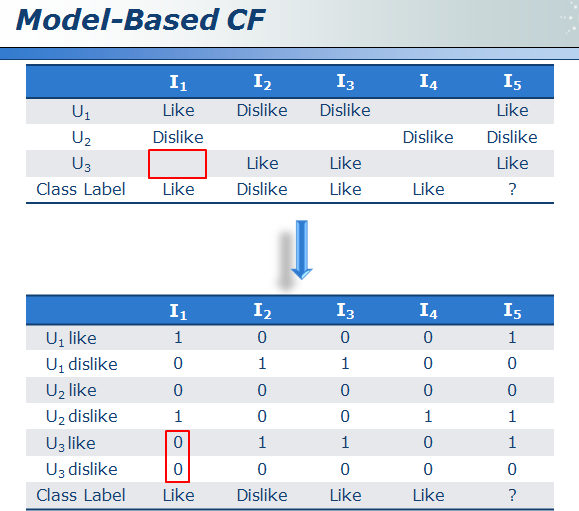
举例：重点关注 基于item的加权slope one算法



基于模型的协同过滤：





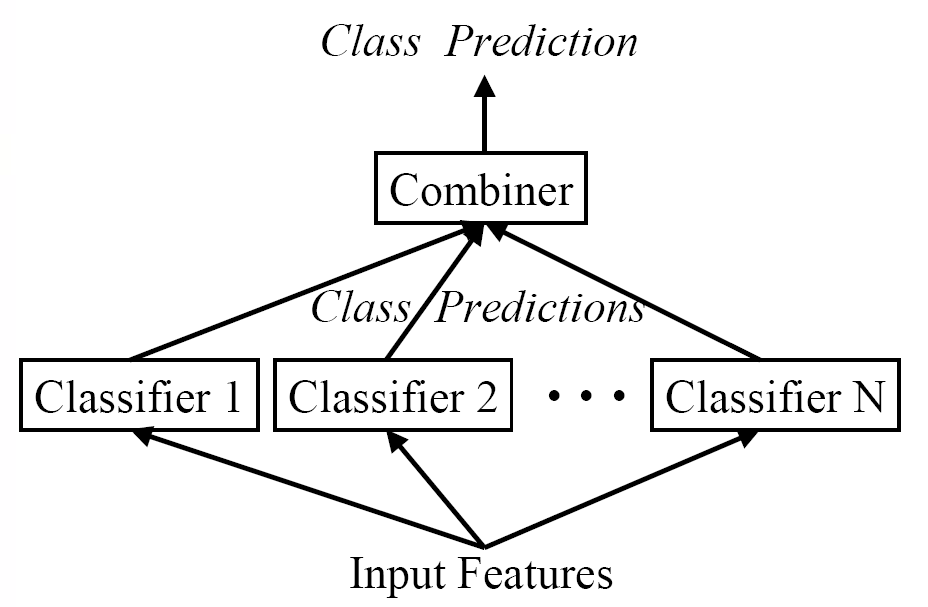
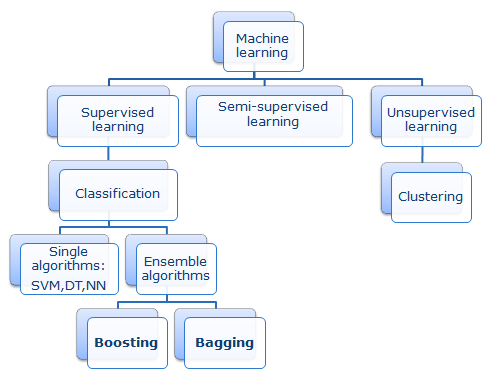


如上图所示，对于like or dislike的二元分类且含有空值的模型，可以将每个使用者的like和dislike分为两个子用户，然后二值化。如选择like或dislike则相应子用户为1，空值则like和dislike都为0

Reading Materials:

* P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “Grouplens: an Open Architecture for Collaborative Filtering of Netnews”, in Proceedings of the ACM Conference on Computer Supported Cooperative Work, pp. 175–186, 1994.
* D. Billsus and M. Pazzani, “Learning Collaborative Information Filters”, in Proceedings of the 15th International Conference on Machine Learning, pp. 46-54, 1998.
* B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms”, in Proceedings of the 10th international Conference on World Wide Web, pp. 285-295, 2001.
* X. Su and T. Khoshgoftaar, “A Survey of Collaborative Filtering Techniques”, Advances in Artificial Intelligence, Article ID: 421425, 2009.
* L. Page, S. Brin, R. Motwani, and T. Winograd, “The PageRank Citation Ranking: Bringing Order to the Web”, Technical Report, Stanford InfoLab, 1999.
* S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, “Indexing by Latent Semantic Analysis”, JASIS, vol. 41(6), pp. 391-407, 1990.
* E. Nathan and A. Pentland, “Reality Mining: Sensing Complex Social Systems”, Personal and Ubiquitous Computing, vol. 10(4), pp. 255-268, 2006.

Ensemble\_Learning



***Combiners***

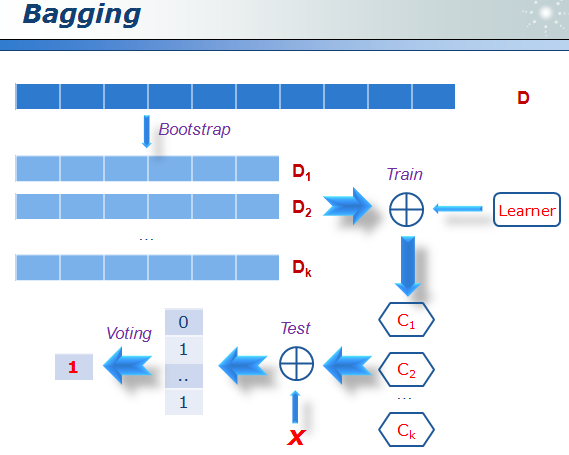
* How to combine the outputs of classifiers?
* Averaging
* Voting
  + Majority Voting
    - Random Forest
  + Weighted Majority Voting
    - AdaBoost
* Learning Combiner
  + General Combiner
    - Stacking
  + Piecewise Combiner
    - RegionBoost

Diversity:

* The key to the success of ensemble learning
  + Need to correct the errors made by other classifiers.
  + Does not work if all models are identical.
* Different Learning Algorithms
  + DT, SVM, NN, KNN …
* Different Training Processes
  + Different Parameters
  + Different Training Sets
  + Different Feature Sets
* Weak Learners
  + Easy to create different decision boundaries.
  + Stumps(树桩) …

***Bagging***

***Bootstrap Samples — 采样方法：有放回采样***



***Random Forests***

* Developed by Prof. Leo Breiman
  + Inventor of CART
  + www.stat.berkeley.edu/users/breiman/
  + Breiman, L.: Random Forests. *Machine Learning* 45(1), 5–32, 2001
  + Bootstrap Aggregation (Bagging)
  + Resample with Replacement
  + Use around two third of the original data.

样本至少被选中一次的概率：，趋于无穷大时，值趋近于2/3

* A Collection of CART-like Trees
  + Binary Partition
  + No Pruning
  + Inherent Randomness
  + Majority Voting

***RF Main Features***

* Generates substantially different trees（生成不同的树，树的权重都一样）:
  + Use random bootstrap samples of the training data.
  + Use random subsets of variables for each node.
* Number of Variables（选择部分样本，通常）
  + Square Root (K)
  + K: total number of available variables
  + Can dramatically speed up the tree building process.
* Number of Trees（通常不低于500棵树）
  + 500 or more
* Self-Testing
  + Around one third of the original data are left out.
  + Out of Bag (OOB)
  + Similar to Cross-Validation

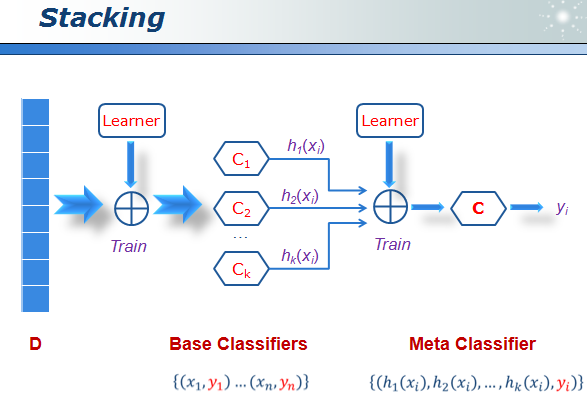
***RF Advantages***

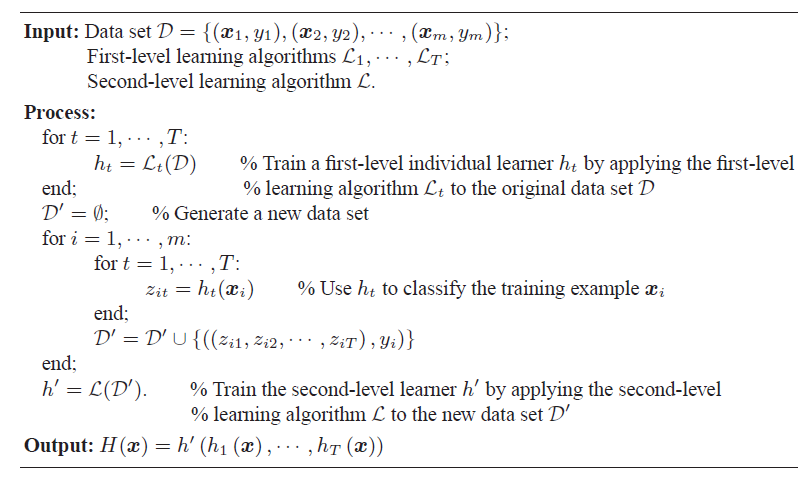
* All data can be used in the training process.
  + No need to leave some data for testing.
  + No need to do conventional cross-validation.
  + Data in OOB are used to evaluate the current tree.
* Performance of the entire RF
  + Each data point is tested over a subset of trees.
  + Depends on whether it is in the OOB.
* High levels of predictive accuracy
  + Only a few parameters to experiment with.
  + Suitable for both classification and regression.
* Resistant to overtraining (overfitting).
* No need for prior feature selection.

***Stacking(bagging的升级版，权重不一样，通过学习获得)***

将base classifiers的输出作为meta classifier的输入

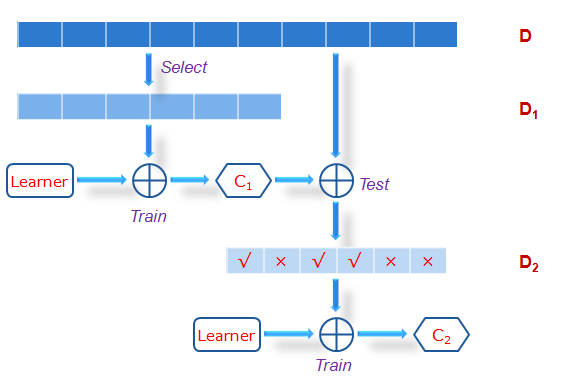
第一层base classifiers的训练实际相当于学习权重

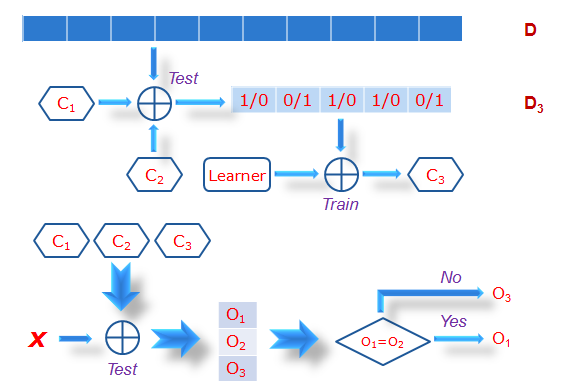




***Boosting***

核心思想：





D1：从所有训练样本中选取的训练样本

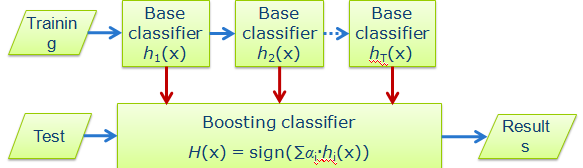
D2：C1分类器分错的训练样本和部分分类正确的样本组成的新样本

D3：C1和C2分类不一致的样本作为新的训练样本

* C1、C2、C3是存在顺序的，串行而不是并行（bagging中不同base classifiers是并行的）

机理步骤：

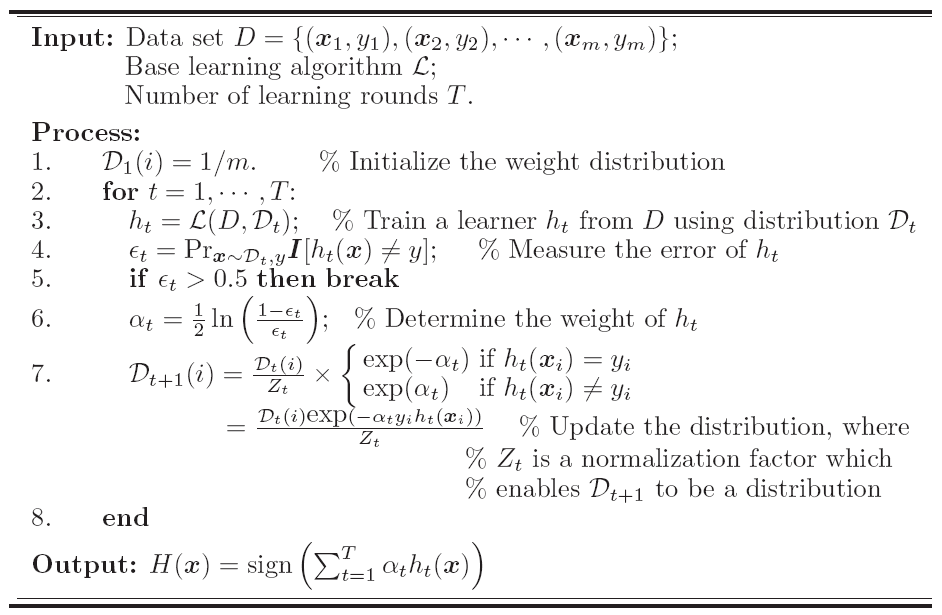
* 训练样本中选取D1用来训练得到分类器C1（弱分类器即可）
* 将所有样本D作为C1的输入进行测试，从C1的Test结果中抽取部分正确分类和错误分类的样本组合成新的训练样本D2（提高错误分类样本的权重）
* 将D2作为样本训练分类器C2
* 将训练样本D作为输入对C1和C2进行Test, 将两个分类器分类结果不一致的样本提取出来组合成新的训练样本D3
* D3作为训练样本训练分类器C3（若有更多C，以此类推）
* 对于新的输入X，分别利用C1、C2和C3进行test, 得到输入O1、O2和O3，如果O1与O2一致，则直接输出O1或O2；若不一致，则输出O3





***AdaBoost – boosting的典型算法***

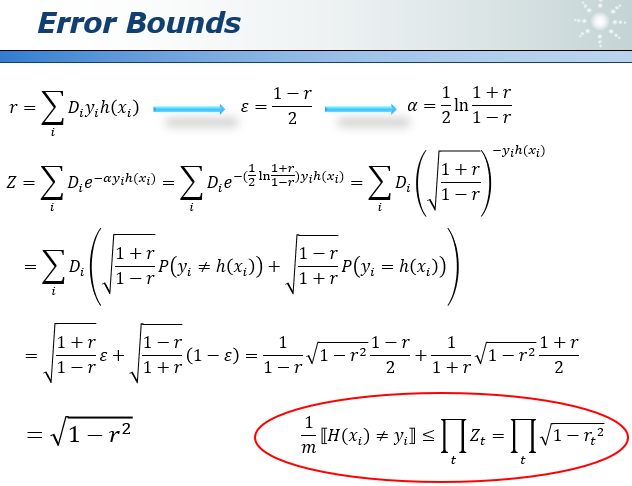
Adaboost属于十大数据挖掘算法中的一种





推导过程太难了！！！

Adaboost参数alpha公式（参数通常是迭代学习得来的，而adaboost的参数alpha是有明确的计算公式）：



上图蓝色框技巧，由于 -y\*h(x)=1 or -1, 因此可按蓝色框进行拆分

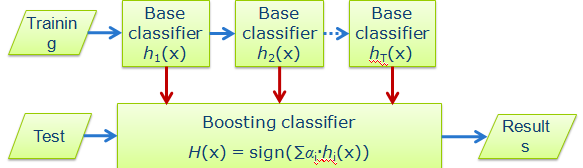
对二元分类p(y=h(x)) = 1 – p(y!=h(x))

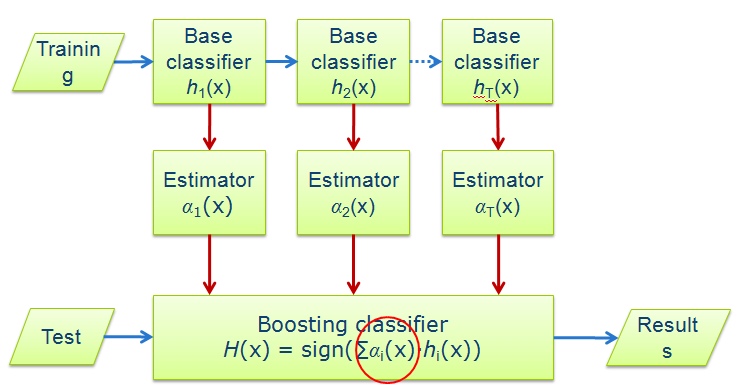
从上图可以看出Z=sqrt(1-r^2), 带入Z，则红色框<=号右边的error bound就可以转化为sqrt(1-r^2)，其中r必定在区间[-1,1]之间,且sqrt(1-r^2)在区间[0,1], 即error bound是由t个[0,1]之间的数字相乘，当t趋于无穷，error bound趋向于0.

因此adaboost的误差最终是趋于0的。

Adaboost的误差上界有严格的公式推导证明，且参数alpha是有公式可以推导计算，这是其他很多算法没有的优点

* Advantages
  + Simple and easy to implement
  + Almost no parameters to tune(基础分类器—50个差不多，弱分类器即可，比random guess好一点儿就行)
  + Proven upper bounds on training set
  + Immune to overfitting
* Disadvantages
  + Suboptimal values（局部最优）
  + Steepest descent
  + Sensitive to noise
* Future Work
  + Theory
  + Comprehensibility
  + New Framework

***Adaboost***

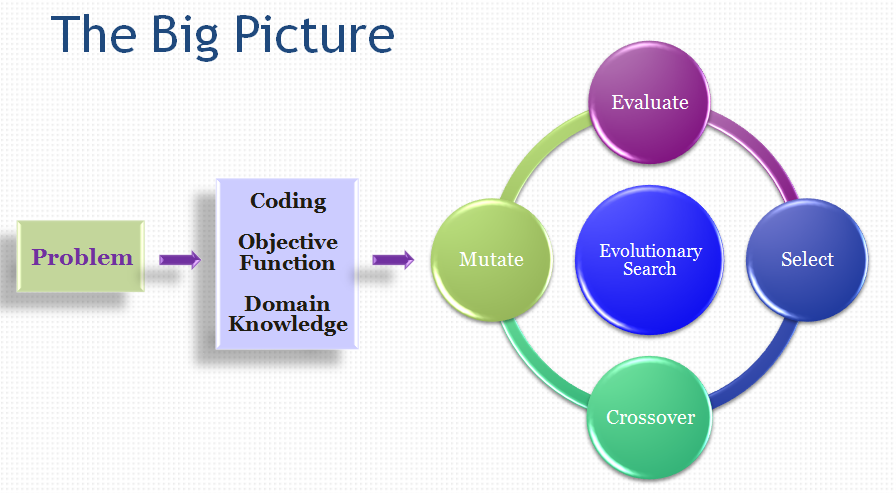
***RegionBoost***

动态权重

***Review***

* What is ensemble learning?
* What can ensemble learning help us?
* Two major types of ensemble learning:
  + Parallel (Bagging)
  + Sequential (Boosting)
* Different ways to combine models:
  + Average
  + Majority Voting
  + Weighted Majority Voting
* Some representative algorithms:
  + Random Forests
  + AdaBoost
  + RegionBoost(测试误差比adaboost表现好，虽然训练误差表现差)

***Evolutionary Algorithms 进化计算***



**Top Journals**:

* IEEE Transactions on Evolutionary Computation
* Evolutionary Computation Journal (MIT Press)

**Major Conferences**:

* IEEE Congress on Evolutionary Computation (CEC)
* Genetic and Evolutionary Computation Conference (GECCO)
* Parallel Problem Solving from Nature (PPSN)

The Limitation of Computation

* FLoating point OPerations per Second
  + GFLOPS (gigaFLOPS): 109 (每秒10亿次运算)
  + TFLOPS (teraFLOPS): 1012 (每秒1万亿次运算)
  + PFLOPS (petaFLOPS): 1015 (每秒1千万亿次运算)
* Intel Core i7 980 XE: ~100 GFLOPS
* Bremermann's Limit
  + c2/*h*=9×1016/6.62606896×10−34 ≈1.36×1050 bits·s-1·kg-1
  + Is it fast enough?

并行计算，解决局部最优解问题

**Genetic Algorithms**

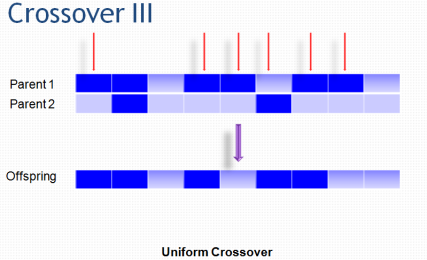
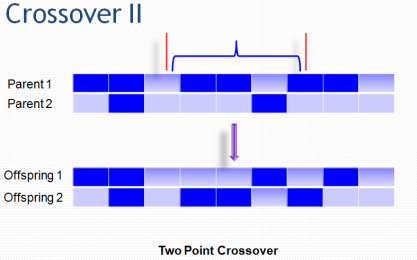
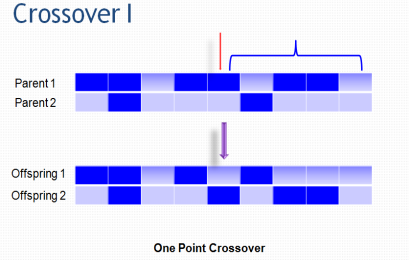
**Basic Components**

* Representation
  + How to encode the parameters of the problem?
  + Binary Problems
    - 10001 00111 11001 …..
  + Continuous Problems
    - 0.8 1.2 -0.3 2.1 …..
    - Genetic Operators
  + Crossover
    - Exchange genetic materials between two chromosomes.
  + Mutation:
    - Randomly modify gene values at selected locations.
* Selection Strategy
  + Which chromosomes should be involved in reproduction?
  + Which offspring should be able to survive?

***Selection***

…

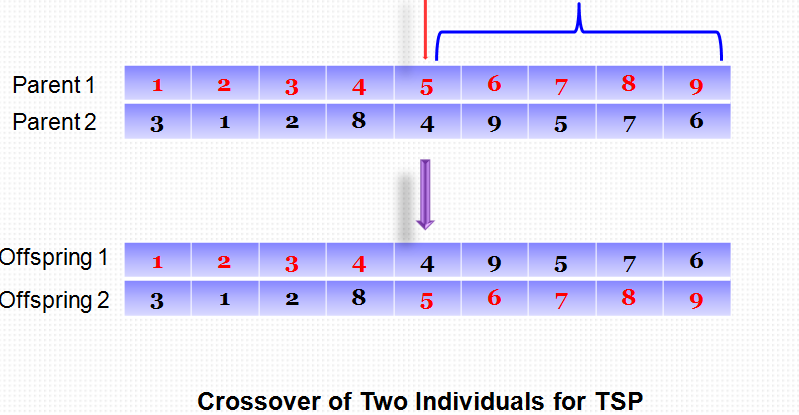
***Crossover 杂交***



单点：指定位置互换

两点：两点间的片段互换

合并: 如1和2全深色为深，一深一浅0.5概率深0.5概率浅，全浅则浅

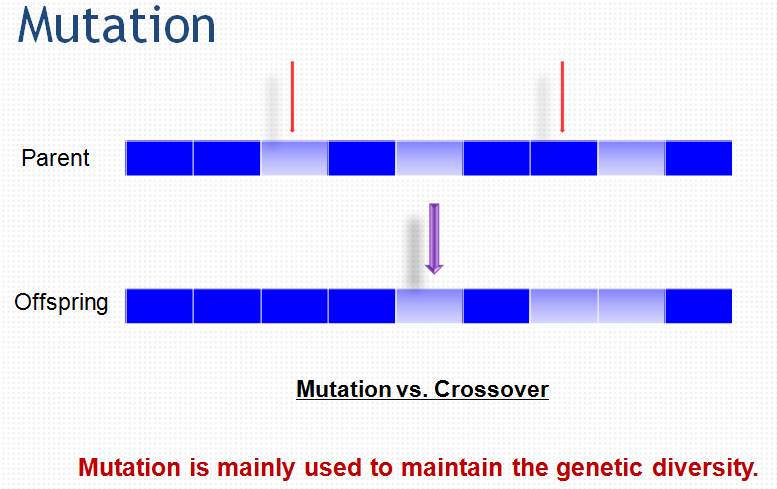


可能出现的问题：杂交互换后的子代不是individual, 存在重复和缺失片段，如上图中crossover后子代offspring1存在两个4，缺少8

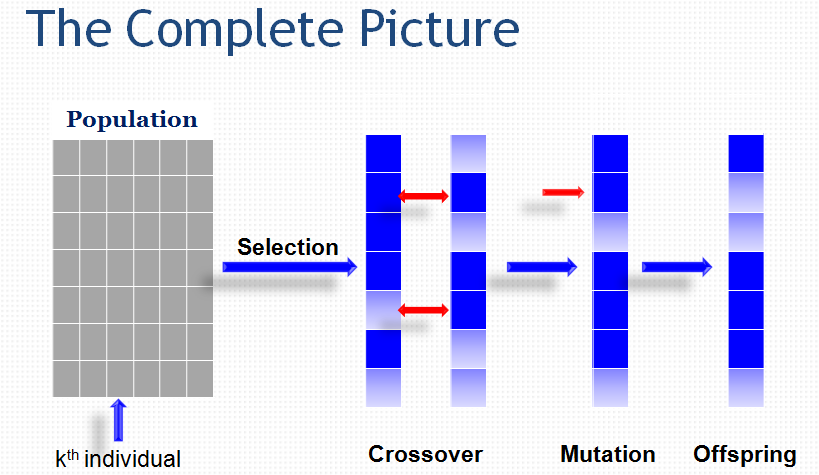
处理方式1：算法将重复的4换成缺失的8

处理方式2：重新设计一种crossover算法

***Mutation 变异***



* **Selection**
  + Bias the search effort towards promising individuals.
  + Loss of genetic diversity
* **Crossover**
  + Create better individuals by combining genes from good individuals.
  + Building Block Hypothesis
  + Major search power of GAs
  + No effect on genetic diversity
* **Mutation**
  + Increase genetic diversity.
  + Force the algorithm to search areas other than the current focus.



**GA Framework：**

**Initialization:** Generate a random population P of M individuals

**Evaluation:** Evaluate the fitness *f*(*x*)of each individual

**Repeat until the stopping criteria are met:**

**Reproduction:** Repeat the following steps until all offspring are generated

**Parent Selection:** Select two parents from P

**Crossover:** Apply crossover on the parents with probability *P*c

**Mutation:** Apply mutation on offspring with probability *Pm*

**Evaluation:** Evaluate the newly generated offspring

**Offspring Selection:** Create a new population from offspring and P

**Output:** Return the best individual found

**Parameters**

* Population Size
  + Too big: Slow convergence rate
  + Too small: Premature convergence（不成熟收敛）
* Crossover Rate
  + Recommended value: 0.8
* Mutation Rate
  + Recommended value: 1/L
  + Too big: Disrupt the evolution process
  + Too small: Not enough to maintain diversity
* Selection Strategy
  + Tournament Selection
  + Truncation Selection (Select top T individuals)
  + Need to be careful about the selection pressure.

Randomness

Inherently stochastic algorithms（随机算法）

Independent trials are needed for performance evaluation.

Why does it work?

Easy to understand & implement (No maths required!)

Very difficult to analyse mathematically.

Converge to global optimum with probability 1 (infinite population).

Schema Theorem