# 分类算法

## KNN(k-Nearlest Neighbor)

### 定义

根据距离函数来计算，待分类样本x，和每个训练样本的距离，以其中最近的k个大多数所属类别来表示其所属类别。关键点，距离的定义。通过调整距离的定义可以达到不同的效果

### 优缺点及适用范围

适合均匀化分布的分类，在文本分类上使用较多，文本分类距离一般定义为cos（θ）；

优点：简单容易实现，重新训练代价低

缺点： 速度慢，每次都要重新计算到所有训练集的距离；当某个样本很大而其他比较少时会影响准确性；对训练集依赖太强；k不好确定；

可行的改进策略

1. 约简去除一些不需要的属性，这里可以用到粗糙集理论
2. 通过聚类，将聚类产生的中心点作为新的训练样本
3. 改变距离的定义，根据实际需求。

### 算法代码

Classify/KNN.py

## 朴素贝叶斯

一般的当 构成完备事件组有

用于分类的定义如下：

1. 设是一个待分类数据
2. 有类别集合
3. 计算
4. 选择出其中最大的即为i的类别

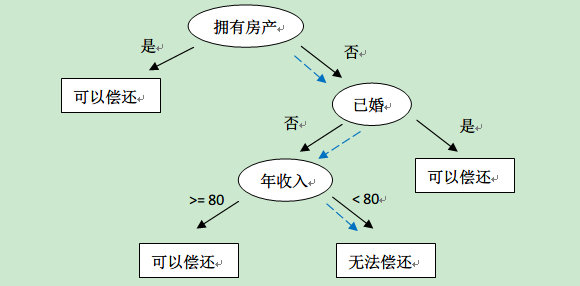
关键点是计算概率，这里假设各个属性是相互独立的

(右边可以通过训练集统计出来)

=

## 决策树

本质上是一个if then 的一个树形判断



决策树构建的基本步骤如下：

1. 开始，所有记录看作一个节点

2. 根据某种分类规则得到最优的划分特征，并创建特征的划分节点——计算最优特征子函数·

3. 根据特征将数据集划分为若干个部分——划分数据集子函数

4. 根据划分结果构建出新的节点，作为树生长出新的分支。

4. 检验是否符合递归条件

5. 将新划分数据作为输入，递归执行上述步骤

### ID3信息熵测度

熵表示任何一种能量在空间中分布的均匀程度。能量分布的越均匀，熵就越大。

香农提出了“信息熵”的概念，解决了对信息的量化度量问题。信息熵这个词是香农从热力学中借用过来的。热力学中的热熵是表示分子状态混乱程度的物理量。香农用信息熵的概念来描述信源的不确定度。

定义信息量： p是事件u发生的概率

信息熵：

      设D为用类别对训练元组进行的划分，则D的[熵](http://en.wikipedia.org/wiki/Entropy)（entropy）表示为：

http://latex.codecogs.com/gif.latex?info(D)=-\sum%20%5em_%7bi=1%7dp_ilog_2(p_i)

      其中pi表示第i个类别在整个训练元组中出现的概率，可以用属于此类别元素的数量除以训练元组元素总数量作为估计。熵的实际意义表示是D中元组的类标号所需要的平均信息量。

      现在我们假设将训练元组D按属性A进行划分，则A对D划分的期望信息为：

http://latex.codecogs.com/gif.latex?info_A(D)=\sum%20%5ev_%7bj=1%7d\frac%7b|D_j|%7d%7b|D|%7dinfo(D_j)

      而信息增益即为两者的差值：

http://latex.codecogs.com/gif.latex?gain(A)=info(D)-info_A(D)

      ID3算法就是在每次需要分裂时，计算每个属性的增益率，然后选择增益率最大的属性进行分裂。

### C4.5

ID3算法存在一个问题，就是偏向于多值属性，例如，如果存在唯一标识属性ID，则ID3会选择它作为分裂属性，这样虽然使得划分充分纯净，但这种划分对分类几乎毫无用处。ID3的后继算法C4.5使用[增益率](http://en.wikipedia.org/wiki/Information_gain_ratio)（gain ratio）的信息增益扩充，试图克服这个偏倚。

C4.5算法首先定义了“分裂信息”，其定义可以表示成：

http://latex.codecogs.com/gif.latex?split\_info_A(D)=-\sum%20%5ev_%7bj=1%7d\frac%7b|D_j|%7d%7b|D|%7dlog_2(\frac%7b|D_j|%7d%7b|D|%7d)

      其中各符号意义与ID3算法相同，然后，增益率被定义为：

http://latex.codecogs.com/gif.latex?gain\_ratio(A)=\frac%7bgain(A)%7d%7bsplit\_info(A)%7d

      C4.5选择具有最大增益率的属性作为分裂属性，其具体应用与ID3类似，不再赘述。

### CART

随机森林是通过很多的决策树组合而成的，随机森林采用CART算法。