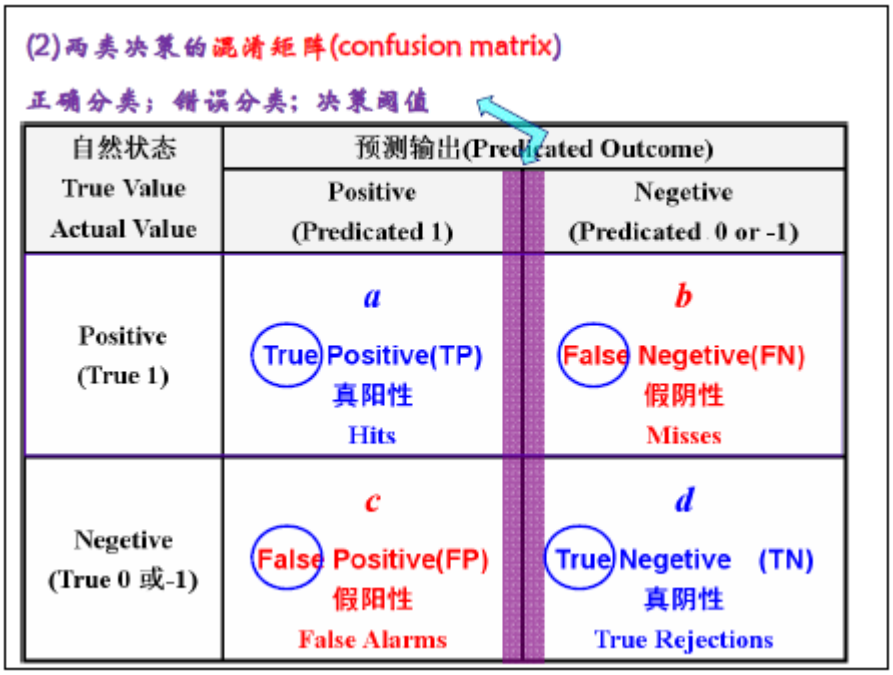
某个类别的，利用混淆矩阵评价分类模型的性能，查准率 查全率 f1 分值，各个类别平均预测的正确率

混淆矩阵：confusion matrix

混淆矩阵的每一列代表了预测类别 ， 每一列的总数表示预测为该类别的数据的数目；

每一行代表了数据的真实归属类别 ， 每一行的数据总数表示该类别的数据实例的数目。



混淆矩阵举例：

TP = True Postive = 真阳性； FP = False Positive = 假阳性

FN = False Negative = 假阴性； TN = True Negative = 真阴性

比如我们一个模型对 15 个样本进行预测，然后结果如下
注： 0 为阳性 1 为阴性
预测值：1 1 1 1 1 0 0 0 0 0 1 1 1 0 1

真实值：0 1 1 0 1 1 0 0 1 0 1 0 1 0 0

	预测值：0	预测值：1	总计
真实值：0	4 TP	4 FN	8
真实值：1	2 FP	5 TN	7
总计	6	9	

准确度(Accuracy) = (TP+TN) / (TP+TN+FN+TN) (即对角线上的值，就是阳性和阴性真实值跟预测值都一样的 / 所有的样本数)

在上面的例子中，准确度 = (5+4) / 15 = 0.6

精度(precision，或者 PPV, positive predictive value) = TP / (TP + FP)

在上面的例子中，精度 =4/（4+2）=0.667

F1-值(F1-score) = 2*TP / (2*TP+FP+FN)

（F1 值 = 正确率 * 召回率 * 2 / (正确率 + 召回率))

在上面的例子中，F1-值 = 2*4/（2*4+2+4）

/*对角线上的数据为预测数据跟真实数据一样*/
为 0 的准确率：
4/（4+2）
为 1 的准确率：
5（5+4）

如果是分类问题　什么是分类，对分类做定义 举几个例子

1. 分类问题的一般描述

给定带有类别标记的训练样本集$\{(x_i, y_i), i=1,\cdots,N\}$.

其中：

x_i----第i个观测样本的特征向量，$x_i=[x_{i1},\ldots,x_{id}]^T \in R^d$

y_i----第i个观测样本的类别标号$\begin{cases} C=2, & y_i \in Y=\{1,2\} \\ C>2, & y_i \in Y=\{1,2,\ldots,C\} \end{cases}$

要求：

基于上述样本集，设计分类模型--**分类模型的监督式学习**;
对特征空间的任意观测x进行类别决策--**模型的使用**

按照不同的特征对数据进行划分成不同的类别。
二分类：比如说根据人脸识别出男性或者女性
多分类：
iris 数据集，根据数据的特征，把该样本划分为不同的品种。
/*具体看分值多少，分值多的话，就多写点，少的话，就简写*/

猫狗兔识别技术:，譬如有一个 1000 个样本的训练集，是 1000 张照片，里面有 200 张是猫，200 张是狗，600 张是兔子，一共分成三类。我们将每个照片向量化后，加上它的标签

“猫”——“0”
“狗”——“1”
“兔子”——“2”

这相当于一个 x 和 y 的对应关系，把它们输入到训练集去训练（但是这个地方的标签 0、1、2 并不是实数定义，而是离散化的标签定义）。经过多轮训练之后，分类器将逻辑关系调整到了一个相对稳定的程度，然后用这个分类器再对这 200 张猫，200 张狗，600 张兔子进行分类的时候。发现：

200 张猫的图片中，有 180 张可以正确识别为猫，而有 20 张误判为狗。

200 张狗的图片可以全部判断正确为狗。

600 张兔子的图片中，有 550 张可以正确识别为兔子，还有 30 张被误判为猫，20 张误判为狗。

K 近邻模型 给了已知标签的样本集 对任何一个样本集进行预测给了 k 紧邻模型 预测的过程 K 代表什么 什么因素会影响 K 金林模型的分类性能

3. K近邻分类的三个基本要素

距离度量方式

超参数K值

决策规则

K 近邻，就是画圈的，在规定的范围内，那个类别的数据离样本数据的个数多，就预测该样本为那个类别

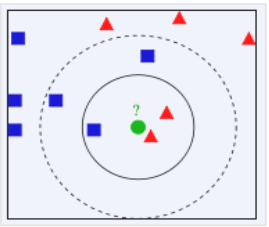
K 最近邻(k-Nearest Neighbor，KNN)分类算法，是一个理论上比较成熟的方法，也是最简单的机器学习算法之一。该方法的思路是：在特征空间中，如果一个样本附近的 k 个最近(即特征空间中最邻近)样本的大多数属于某一个类别，则该样本也属于这个类别。

案例介绍

编辑

如右图所示，有两类不同的样本数据，分别用蓝色的小正方形和红色的小三角形表示，而图正中间的那个绿色的圆所标示的数据则是待分类的数据。也就是说，现在， 我们不知道中间那个绿色的数据是从属于哪一类（蓝色小正方形or红色小三角形），下面，我们就要解决这个问题：给这个绿色的圆分类。

我们常说，物以类聚，人以群分，判别一个人是一个什么样品质特征的人，常常可以从他/她身边的朋友入手，所谓观其友，而识其人。我们不是要判别上图中那个绿色的圆是属于哪一类数据么，好说，从它的邻居下手。但一次性看多少个邻居呢？从上图中，你还能看到：



- 如果K=3，绿色圆点的最近的3个邻居是2个红色小三角形和1个蓝色小正方形，少数从属于多数，基于统计的方法，判定绿色的这个待分类点属于红色的三角形一类。
- 如果K=5，绿色圆点的最近的5个邻居是2个红色三角形和3个蓝色的正方形，还是少数从属于多数，基于统计的方法，判定绿色的这个待分类点属于蓝色的正方形一类。

于此我们看到，当无法判定当前待分类点是从属于已知分类中的哪一类时，我们可以依据统计学的理论看它所处的位置特征，衡量它周围邻居的权重，而把它归为(或分配)到权重更大的那一类。这就是K近邻算法的核心思想。

KNN算法中，所选择的邻居都是已经正确分类的对象。该方法在定类决策上只依据最邻近的一个或者几个样本的类别来决定待分样本所属的类别。

2.2 K近邻分类算法的描述

输入：

(1) 训练样本集 $D=\{(x_i,y_i),i=1,\cdots,N\}\subset R^d\times Y$，并且有：

x_i----第i个训练样本的特征向量，$x_i\in R^d$

y_i—第i个训练样本的类别标号 $\begin{cases}C=2, & y_i\in Y=\{1,2\}\\ C>2, & y_i\in Y=\{1,2,\ldots,C\}\end{cases}$

(2) 观测样本x

输出： 观测样本x所属的类别y.

2.2 K近邻分类算法的描述 - 续

STEP0. 训练集D的输入部分预处理，并记录预处理的使用参数

STEP1. 指定**距离度量**，并**选择K值**

STEP2. 训练集D内找到预处理的样本x的前**K个近邻**，**记为**$N_K(x)$

$N_K(x) = N_{K,1}(x) \cup \cdots \cup N_{K,C}(x)$

$N_{K,i}(x)$----x的前K个近邻中，属于第i类的部分

STEP3. 结合指定的**分类规则**，对x的类别y 进行预测

$$\hat{y} = \underset{i \in Y=\{1,\ldots,C\}}{\operatorname{argmax}} \sum_{x_j \in N_k(x)} I(y_j = i)$$

注：指示函数$I(y_j = i) = \begin{cases} 1, if\ y_j = i \\ 0, if\ y_j \neq i \end{cases}$

什么因素会影响 K 金林模型的分类性能 ？

在给定训练集的前提下，
不同的距离度量方式、
不同的K值、
不同的决策规则，

会导致不同的分类结果

假定在树的构建过程中, 怎么用训练样本集来评价当前三种 节点不纯度的方法 三种方法都计算一下不纯度

(1)有关概念

- **纯结点(数据集)、不纯结点(数据集)**

若到达某结点的**训练样本集**只含一类样本，则该结点为**纯(pure)结点**，或为**同质(homogenous) 结点**

否则，为**不纯(impure)**、或**异构(heterogeneous)结点**。

- **结点的不纯度(impurity，杂度)**

关于**决策树结点不纯程度**的度量.

(2)结点不纯度的典型度量方式

设到达某结点的训练样本集D含K个不同类别，$D=D_1 \cup \cdots \cup D_K$

类别集合$Y=\{\omega_1,\ldots,\omega_K\}$　　$K=|Y|$

样本容量$N=|D|=\sum_{j=1}^{|Y|}|D_j|=\sum_{j=1}^{K}N_j$

第j类出现的概率 $P_j \approx \frac{|D_j|}{|D|}=\frac{N_j}{N}$

$\sum_{j=1}^{K}P_j=1$

One

A. 熵不纯度(entropy impurity)

$$I_{Entropy}(D)=-\sum_{i=1}^{K}P_i\log_2 P_i$$

约定：　$0\log 0=0$

各类别等概率出现：$I_{Entropy}(D)=\sum_{i=1}^{K}\frac{1}{K}\log_2 K=\log_2 K$

只出现一个类别：　$I_{Entropy}(D)=0$

Two

B. Gini不纯度(Gini impurity)/方差不纯度

$$I_{Gini}(D)=\sum_{j=1}^{K}\sum_{\substack{i=1\\ i\neq j}}^{K}P_iP_j=1-\sum_{j=1}^{K}P_j^2$$

各类别等概率出现：$I_{Entropy}(D)=1-\sum_{i=1}^{K}\frac{1}{K^2}=\frac{K-1}{K}$

只出现一个类别：　$I_{Entropy}(D)=0$

Three

C. 误差不纯度

$$I_{Error}(D)=1-\max_{j\in\{1,\dots,k\}}P_j$$

各类别等概率出现：$I_{Entropy}(D)=1-\frac{1}{K}=\frac{K-1}{K}$

只出现一个类别：$I_{Entropy}(D)=0$

如果现在节点已经选好了特征，后面已经分成了几个分支 计算一下这种划分 决策树的绝对增益 用信息商，三种规则 节点不纯度的三种计算方式

假定节点已经分裂了 怎么计算节点的绝对增益

绝对增益

例：ID3决策树内每个非叶结点的特征选择，采用最大“绝对信息增益”准则，选特征

$$a^*=\arg\max_{a\in A}Gain(D,a)$$

B. 信息增益率(Information Gain Ratio)—相对增益

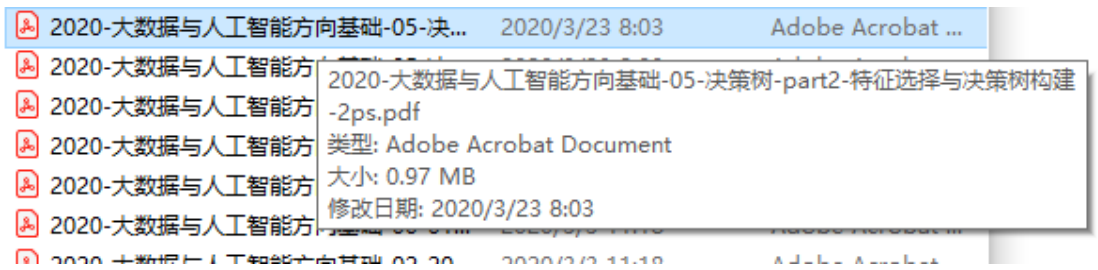
$$Gain\_ratio(D,a)=\frac{Gain(D,a)}{IV(a)}$$

特征a对训练集D的**绝对信息增益**$Gain(D,a)$

$$Gain(D,a)=I_{Entropy}(D)-\sum_{i=1}^{m}\frac{|D^{(i)}|}{|D|}I_{Entropy}(D_i)$$

$$=-\sum_{j=1}^{K}\frac{|D_j|}{|D|}\log_2\frac{|D_j|}{|D|}-\sum_{i=1}^{m}\frac{|D^{(i)}|}{|D|}\left[-\sum_{j=1}^{K}\frac{|D_j^{(i)}|}{|D^{(i)}|}\log_2\frac{|D_j^{(i)}|}{|D^{(i)}|}\right]$$

(没学好，跳过)具体的知识点应该在这个 ppt 中



两个集成模型，一个随机森林，一个 bagging 树

模型评价　混淆矩阵评价模型　根据混淆矩阵计算别的指标

（ https://gitee.com/lijianmin1/statistical_learning/blob/master/%E5%86%B3%E7%AD%96%E6%A0%91/Bagging_and_%E9%9A%8F%E6%9C%BA%E6%A3%AE%E6%9E%97.ipynb）

bagging 使用了 boostrap 的思想，从 m 个样本的训练集中有放回地抽取 m 次，获得第一个样本集，用于训练第一个基学习器，以此类推可获得 k 个样本集供基学习器训练。这里训练得到的模型都是各个独立的，最后，用测试数据，经过每个模型预测，最后采用投票的方式获得预测的结果。

其实，随机森林就是 bagging 树演化过来的，不同的是，在训练数据的时候，bagging 使用的样本的所有特征，随机森林，可以随机的抽取特征进行训练（可以自己设定使用多少特征），后面的其实都是一样的。

模型的评价:

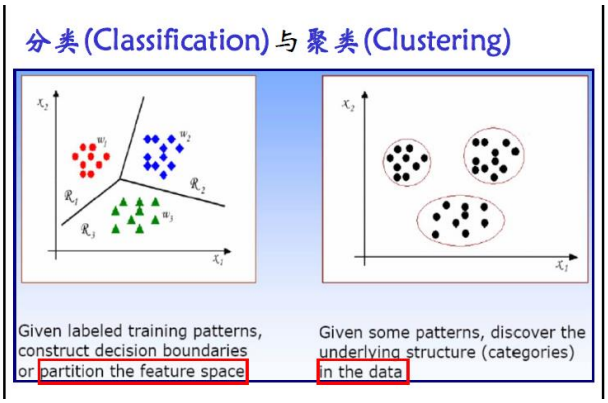
(1)基于投票法的类别决策

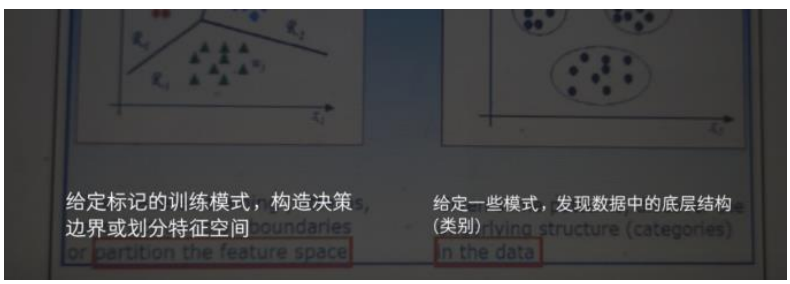
(2)基于预测概率的类别决策

投票法就是，比如一个测试样本，根据每个模型所得到的结果（0,1），进行投票，看看那个类别得到的票多，就预测这个样本是那个类别。

预测概率的决策（其实就是投票法的一种变形，根据每个模型预测这个样本可能是每个类别的概率（0~1 之间的数）），投票累加概率和，那个类别的所得票数多，就预测这个样本为那个类别。）

要了解大数据与人工智能 知道 什么是分类 什么是聚类 能够简单的一两句举例说明做一个定义





聚类和分类区别

原创 rocling 2019-06-06 16:59:21 4920 收藏

1. 产生的结果相同（将数据进行分类）
2. 聚类事先没有给出标签（无监督学习）

聚类与分类的定义

原创 meaworld 2013-01-21 14:05:53 5044 收藏 展开

1.聚类的概念：

有一堆数据，讲这堆数据分成几类称为聚类。

举个例子，

比如有一堆水果，我们按着不同的特征分为：苹果，橘子，香蕉三类叫做分类。

2.分类的概念：

在聚类的前提下，拿来一个新水果，我们按着他的特征，把他分到橘子或者香蕉那类中，叫做分类。

3.训练集和测试集

一般就是把数据分成10份，9:1

9份作为训练数据，来学习一个模型；

1份作为测试数据，来测试这个模型。

什么是监督式学习 非监督式学习 举例说明

监督学习和非监督学习的差别之一就在于：有没有目标值的差别

而另一个区别就在于：学习过程有没有人工干预

- **监督学习**

当一个孩子逐渐认识事物的时候，父母给他一些苹果和橘子(目标值)，并且告诉他苹果是什么样的，有哪儿些特征(特征值)，橘子是什么样的，有哪儿些特征(特征值)。经过父母的不断介绍，这个孩子已经知道苹果和橘子的区别，如果孩子在看到苹果和橘子的时候给出错误的判断，父母就会指出错误的原因（人工干预），经过不断地学习，再见到苹果和橘子的时候，孩子立即就可以做出正确的判断。

- **非监督学习**

同样的一个孩子，在一开始认识事物的时候，父母会给他一些苹果和橘子，但是并不告诉他哪儿个是苹果，哪儿个是橘子，而是让他自己根据两个事物的特征自己进行判断，会把苹果和橘子分到两个不同组中，下次再给孩子一个苹果，他会把苹果分到苹果组中，而不是分到橘子组中。

监督式用于分类和回归模型的学习 需要训练集学习模型的时候 需要样本的输入部分和别的部分

聚类 采用没有标签的数据集完成 没有用到其他的

（对哒，基本就是这个意思）

模型的评价 两类方式 以分类为例，分类比较多对于分类来说 有两大类评价的方式 假定用训练集写好了分类模型 让模型对测试集挨个测试

得到混淆矩阵 基于混淆矩阵的模型评价 混淆矩阵：两类别的分类问题 两个类别同时都是我们关注的类别 比如性别分类 多类别的分类 每个都是关注的

特定的分类 比如人脸检测 给定两个或者两个以上的内容 同时感兴趣的类别 这样的分类模型得到的混淆矩阵 c 行 c 列的，混淆矩阵每一个元素统计的是样本的数量

，

基于混淆矩阵做的评价 算出来参与决策的样本数量，利用混淆矩阵能够估计测试样本集总体预测的错误率和正确率 主对角线的元素都是正确(预测出来的结果跟真实结构一样)的样本数量，

估计某一个特定类别的查准率查全率以及 f1 分值。算出每一个类别的正确率 c 个类别的预测正确率的平均值

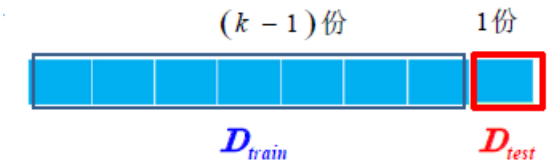
（重复了，大概就是这意思）

分类是已知标签的，聚类是未知标签的！

K 折交叉验证：（可用来进行模型选择或模型评价）

①单轮 K 折交叉验证

把数据集 D 随机打乱，均分成 K 等份。依次从头到尾遍历每个等份中的样本，并以该等份中的样本作为测试集，以其他所有等份中的样本作为训练集，评估本轮交叉验证的正确率(或错误率)。完成遍历后，以 K 轮交叉验证的平均正确率(或错误率)作为最终交叉验证的总体正确率（或错误率）。



②多轮 K 折交叉验证即将单轮 K 折交叉验证重复多次。
K 折交叉验证可能涉及到分层随机打乱
分层随机打乱：在保证每一折中每类样本数量一致的前提下进行样本的随机打乱。

三种监督式学习模型（监督式即样本标签已知的情况下进行模型学习）
监督式学习：基于已知标签的数据集进行模型的学习，基于该模型对未知样本的输出做出预测。
（考分类问题的概率比较大）
一、KNN 模型（K 近邻模型）
流程：
1. 对训练样集 D 进行标准化预处理，并记录预处理的使用参数。
   标准化预处理通常采用 0 均值、1 方差的方式：
   对于每个样本，用该样本每个特征的值减去该特征的均值除以该特征的标准差。
2. 指定距离度量，并选择 K 值。
   距离度量通常采用欧氏距离：（欧式距离就是通常意义上说的距离）

C. 欧式距离（L_2距离） $d_2(\boldsymbol{x}_i,\boldsymbol{x}_j)=\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_2=\left(\sum_{k=1}^{d}|x_{ik}-x_{jk}|^2\right)^{\frac{1}{2}}$

K 值的选择通常用到 m 折交叉验证(跟之前说的 K 折交叉验证一样)

***STEP*1.训练集随机打乱，均分成m等份，每一份的训练样本数目$\frac{N}{m}$.**

***STEP*2.对于每个备选K值，**

2-1. for $i=1,\ldots,m$ *do*

拿出第i份作为验证集，其余m-1份构成估计集

利用估计集，对验证集的每个样本进行类别预测，得验证集预测错误率$Err_i(K$

2-2.估计对应于该备选K值的
- 平均错误率 $\mu_{Err(K)}=\frac{1}{m}\sum_{i=1}^{m}Err_i(K)$
- 标准差 $\sigma_{Err(K)}=\sqrt{\frac{1}{m}\sum_{i=1}^{m}\left[Err_i(K)-\mu_{Err(K)}\right]^2}$

表示为$\mu_{Err(K)}\pm\sigma_{Err(K)}$

***STEP*3.取最小$\mu_{Err(K)}$对应的K值为最终选择结果；**

若同时有多个K值有最小$\mu_{Err(K)}$，则取较小$\sigma_{Err(K)}$对应的K值。

3. 在训练集 D 内找到预处理的样本 x 的前 K 个近邻。
   就是找到距离当前样本最近的 K 个近邻。
4. 结合指定的分类规则，对 x 的类别进行预测。
   结合最近邻的 K 个样本的类别，来决策当前 x 的类别。
   决策方式：①多数表决，即近邻的 K 个样本中哪一类的样本最多就预测为哪一类。
   ②基于距离的加权投票。在最近邻的 K 个样本中，距离当前样本越近的样本权重越大。

二、决策树模型

决策树模型分为分类树（用于预测类别）和回归树（用于实值函数的回归）。

度量不纯度的方法：（这个只能记住公式）

1. 熵不纯度

$$I_{Entropy}(D) = -\sum_{i=1}^{K} P_i \log_2 P_i$$

约定： $0 \log 0 = 0$

各类别等概率出现： $I_{Entropy}(D) = \sum_{i=1}^{K} \frac{1}{K} \log_2 K = \log_2 K$

只出现一个类别： $I_{Entropy}(D) = 0$

2. Gini 不纯度

$$I_{Gini}(D) = \sum_{j=1}^{K} \sum_{\substack{i=1 \\ i \neq j}}^{K} P_i P_j = 1 - \sum_{j=1}^{K} P_j^2$$

各类别等概率出现： $I_{Entropy}(D) = 1 - \sum_{i=1}^{K} \frac{1}{K^2} = \frac{K-1}{K}$

只出现一个类别： $I_{Entropy}(D) = 0$

3. 误差不纯度

$$I_{Error}(D) = 1 - \max_{j \in \{1,\ldots,k\}} P_j$$

各类别等概率出现： $I_{Entropy}(D) = 1 - \frac{1}{K} = \frac{K-1}{K}$

只出现一个类别： $I_{Entropy}(D) = 0$

基于不纯度的节点特征选择（具体得记公式，有点复杂跳过了）

1. 绝对信息增益
2. 信息增益比
3. 基尼指数

三种决策树:

①ID3 决策树：基于最大绝对信息增益的特征选择原则。（只能用于分类）

算法基本点：

- 若当前结点只含同一类样本，则为**纯结点**，则停止分裂；
- 若当前特征列表中**再无可用特征**，则根据**多数表决**确定该结点的类标号，停止分裂；
- 其它：选择最佳分裂的**特征**(**最大信息增益足够大**)

根据所选特征取值(**特征取值数目决定了该结点分裂为后继子结点的数目**)，逐一进行分裂；递归构造决策树。

只适用于离散型或者非数值型特征描述的样本集，不处理缺失信息、不涉及剪枝操作。

②C4.5 决策树：基于最大增益率的特征选择原则。(只能用于分类)

相比 ID3 决策树：可以对连续数值特征进行处理、可以对缺失值进行处理、首先让树充分生长，然后利用分枝的统计显著性来实现剪枝（后剪枝）。

对于连续数值型的特征采用：二分法。基于最大绝对信息增益找最佳切分点，基于信息增益比选择最佳分裂特征。

③CART 决策树：基于划分后基尼指数最小的特征选择原则。(既能用于分类也可用于回归)

CART 树一定是二叉树!

CART 分类树算法：

CART树--递归二叉分类树的生成算法

基本思想：

一个分类树对应输入空间(或特征空间)的一个划分，以及在该划分单元上的类别输出值。

根据训练样本集D，从根结点开始，将输入空间进行划分，递归构建二叉分类树。

借助**基尼指数**进行特征选择，同时决定该特征的**最优二值切分点**

CART 回归树算法：

CART树--最小二乘回归树的生成算法

基本思想：

一个回归树对应输入空间(或特征空间)的一个划分，以及在该划分单元上的输出值。

在训练样本集D所在的输入空间，递归地将每个区域划分为两个子区域，并根据落入每个子区域的训练样本输出值，决定该子区域的输出，构建二叉树。

过拟合：具有较低训练误差的模型，其泛化误差可能高于具有较高训练误差的模型，这种情况称为模型过拟合(过学习)。

欠拟合：决策树规模很小时，训练和检验误差都很大， 这种情况为模型的欠拟合(欠学习)，原因是模型尚未学习到数据的真实结构。

剪枝：

1. 预剪枝：在完全拟合整个训练集之前就停止决策树的生长。
2. 后剪枝：初始阶段--决策树按照最大规模生长。剪枝阶段--修剪完全增长的决策树。