1. 决策树

什么是**决策树**?

决策树是由结点和有向边组成的**树**,其中结点包括内部结点和叶结点.内部结点表示一个特征或属性,叶结点表示一个类.

决策树(Decision Tree)算法是一种常用的机器学习算法,在分类问题中,它通过样本中某一维属性的值将样本划分到不同的类别中.

决策树算法是基于**树结构**进行决策的.

* 1. 数据集的最佳划分标准

在决策树中,通常有这些标准: 信息增益 (Information Gain),增益率(Gain Ratio),基尼系数(Gini Index).

度量样本集合纯度最常用的一种指标是信息熵. 对于包含个训练样本的数据集D:{(,),…,(,)},第类样本所占的比例为, 则数据集的**信息熵**(Entropy)定义为:

其中,K表示数据集D中类别数.

`

注:

1. 信息熵,又称**香农熵**,为纪念信息论的创始人克劳德.香农**.**
2. **信息**定义为事件发生前后我们所掌握的香农熵的变化**.** 如果一个事件发生为不同状态的概率为 (k=1,…,K),那么在它确定发生以前,我们认为事件含有一定量的熵,其值定义为:

而一旦它发生,就没有任何不确定性,熵下降为0. 所以,上式中的就被定义为事件的信息量.[1]

**例2**. 求投掷一枚硬币的信息熵.

解: y{0,1},所以, .

硬币正面向上和反面向上的概率**相等**且和为1,有

, .

信息熵等于

=1

**例3**. 求投掷一枚骰子的信息熵.

解: {1,2,3,4,5,6},所以, .

骰子各面向上的概率都相等且和为1,有

信息熵等于

总结:

* + - 1. 在**概率**均等情况下,存在可能性越多,则信息熵越大.
      2. 一般地,概率分布越平均,信息熵越大;当所有概率均等时,信息熵达到最大.

若可划分为两个独立的子数据集,则此时整个数据集的信息熵为:

.

其中,

为数据集中样本的个数,

E(D1) = - p1\*log(p1) = -1 \* log(1) 为数据集中的个数,为数据集中样本的个数.

= 0

E(D2) = - p1\*log(p1) - p2\*log(p2) = - (2/3)\*log(2/3) - (1/3)\*log(1/3)

|D1|/|D| = 3/6

|D2|/|D| = 3/6

例4. 投掷一枚骰子. 将按”朝上的面的点数是否为奇数”划分为两个独立的**子数据集**. 求整个数据集的信息熵. ;

解:

2.58 -1.58 = 1.00

总结: 划分后,数据集D的信息熵比划分前少了!

练习:

; .求信息熵.

2.58 -1.67 = 0.91

将给定数据集D按照特征的值划分后,定义**信息增益** (Information Gain)为

,

其中,表示属于第类的样本的个数.

可将信息增益率作为划分数据集的一种方法.通常,在选择数据划分的标准时,我们要选使信息增益最大的划分.

对于给定数据集D,定义**信息增益率** (Information Gain Ratio)为

其中,称为特征A的**固有值**(Intrinsic Value):

也可将信息增益率作为划分数据集的一种方法.

与信息熵类似,对于有个分类的数据集 样本属于第个类的概率为,定义**基尼指数**为

.

若可划分为两个独立的子数据集,则此时整个数据集的基尼指数为:

.

其中,为数据集中样本的个数, 为数据集中的个数,为数据集中样本的个数. 我们也可将基尼指数作为划分数据集的一种方法.

例5. 已知数据集D如下. 求基尼指数.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **有鳃否?** | **有鳍否?** | **是鱼否?** |
| 鲨 | 1 | 1 | 1 |
| 鲫鱼 | 1 | 1 | 1 |
| 河蚌 | 1 | 0 | 0 |
| 鲸 | 0 | 1 | 0 |
| 海豚 | 0 | 1 | 0 |

解: D:{(,),…,(,)},其中

;

…

利用”是否有鳃”这一特征将数据集D划分为独立的两个数据集后,

基尼指数为

Gini(D)= (3/5)\*Gini(D1) + (2/5)\*Gini(D2)

注: 在**CART算法**中,就是利用**基尼指数来**划分数据集的.

* 1. 停止划分的标准

划分终止条件有:(split)

* + - 1. 结点中样本数小于给定阈值
      2. 样本集的Gini指数小于给定阈值

已没有更多特征.

决策树学习常用的算法有：ID3，C4.5，**CART**，包含：特征选择（决定用哪个特征划分特征空间），决策树的生成和剪枝（将已生成的决策树进行简化）过程.

* 1. 附. 决策树算法

本质上是从训练数据集中归纳出一组分类规则。可能有多个，可能没有。我们需要的是一个与训练数据矛盾较小的决策树，同时具有很好的泛化能力。从另一个角度看，决策树学习是由训练数据集估计条件概率模型。基于特征空间划分的类的条件概率模型有无穷多个。我们选择的条件概率模型应该不仅对训练数据有很好的拟合，而且对未知数据有很好的预测。

**决策树学习的损失函数：通常是正则化的极大似然函数**

**决策树学习的策略：是以损失函数为目标函数的最小化.**

因为从所有可能的决策树中选取最优决策树是NP完全问题，所以现实中决策树学习算法通常采用启发式方法，近似求解这一最优化问题，得到的决策树是次最优

**决策树学习的算法通常是一个递归地选择最优特征，并根据该特征对训练数据进行分割，使得对各个子数据集有一个最好的分类的过程。**

**剪枝：决策树可能对训练数据有很好的分类能力，但可能发生过拟合现象.。所以需要对已生成的树自下而上进行剪枝，将树变得更简单，从而使它具有更好的泛化能力。具体地，就是去掉过于细分的叶结点，使其回退到父结点，甚至更高的结点，然后将父结点或更高的结点改为新的叶结点.**

**特征选择：如果特征数量很多，在决策树学习开始时对特征进行选择，只留下对训练数据有足够分类能力的特征。**

由于决策树表示一个条件概率分布，所以深浅不同的决策树对应着不同复杂度的概率模型。决策树的生成对应模型的局部选择，决策树的剪枝对应于模型的全局选择。决策树的生成只考虑局部最优，决策树的剪枝则考虑全局最优。