Random Forest 推导

Soul Walker

2020 年 7 月 22 日

1 熵的知识补充

决策树这里涉及到的数学问题就是关于特征的选择问题，那我们在构建决策树的时候选择的生成算法是 ID3 还是 C4.5 这两种算法就涉及到了信息增益与信息增益比的概念。

首先，我们先来介绍熵 (entropy)，熵度量了事物的不确定性，越不确定的事物，他的熵就越大

$$H(X) = -\sum_{i=1}^{n} p(x_i) log p(x_i)$$

其中，n 表示了 X 的 n 中不同取值（且离散），log 为以 2 或 e 为底的对数。

同样的，我们将其推广到多个变量的联合熵

$$H(X,Y) = -\sum_{i=1}^{n} p(x_i, y_i) log p(x_i, y_i)$$

其中，n 表示了 X 和 Y 的 n 中不同取值（且离散），log 为以 2 或 e 为底的对数。

那么通过联合熵，我们可以的到条件熵, 条件熵 $H(Y|X)$ 表示在已知随机变量 X 的条件下随机变量 Y 的不确定性。条件熵 $H(Y|X)$ 定义为 X 给定条件下 Y 的条件概率分布的熵对 X 的数学期望

$$\begin{aligned} H(Y|X) &= \sum_{i=1}^{n} p(x_i) H(Y|X = x_i) \\ &= -\sum_{i=1}^{n} p(x_i) \sum_{j=1}^{n} p(y_j|x_i) log p(y_j|x_i) \\ &= -\sum_{i=1}^{n} \sum_{j=1}^{n} p(x_i, y_j) log(y_j|x_i) \end{aligned}$$

条件熵 $H(Y|X)$ 相当于联合熵 $H(Y,X)$ 减去单独的熵 $H(X)$

$$
\begin{aligned}
H(X,Y) &= -\sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i,y_i) logp(x_i,y_i) \\
&= -\sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i,y_i) log(p(y_j|x_i)p(x_i)) \\
&= -\sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i,y_i) logp(y_j|x_i) - \sum_{i=1}^{n}\sum_{j=1}^{n} p(x_i,y_i) logp(x_i) \\
&= H(Y|X) - \sum_{i=1}^{n} logp(x_i) \sum_{j=1}^{n} p(x_i,y_j) \\
&= H(Y|X) - \sum_{i=1}^{n} logp(x_i)p(x_i) \\
&= H(Y|X) - \sum_{i=1}^{n} p(x_i) log(x_i) \\
&= H(Y|X) + H(X)
\end{aligned}
$$

也就是

$$H(Y|X) = H(X,Y) - H(X)$$

2 ID3 和 C4.5 算法

ID3 和 C4 这两个算法的区别在与 ID3 在特征选择的时候使用信息增益准则选择特征，而 C4.5 则选择使用信息增益比准则来选择特征。

输入：训练数据集

$$T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N))\} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$$

其中，$\mathbf{x}_i \in \chi \in \mathbb{R}^N$ 为实例的特征向量，$\mathbf{x}_i = (\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}, ..., \mathbf{x}_i^{(n)})^T$ ，$y_i \in \{1, 2, \cdots, K\}$ 为实例的类别，$i = 1, 2, \cdots, N$，实例特征向量 $\mathbf{x}$，$\mathbf{X}$ 为训练样本集，形状为 (N, p)，$\mathbf{Y}$ 是训练标签集，形状为 $(N, 1)$。这里我们使用 α 来作为拉格朗日乘子。

输出：实例 $\mathbf{x}$ 所属的类别 y。

此处假设训练集中某一特征 A，特征 A 将 D 划分为 n 个子集 $D_1, D_2, \cdots, D_n$，$|C_k|$ 为第 k 类的

样本数，我们来计算信息增益 $g(D,A)$

$$\begin{aligned}
g(D,A) &= H(D) - H(D|A) \\
&= -\sum_{k=1}^{K} p(\mathbf{Y}=C_k) log P(\mathbf{Y}=C_k) - (-\sum_{i=1}^{n} P(\mathbf{x}^{(A)}=a_i) H(D|\mathbf{X}^{(A)}=a_i)) \\
&= -\sum_{k=1}^{K} \frac{|C_k|}{|D|} log \frac{|C_k|}{|D|} - (-\sum_{i=1}^{n} \frac{|D_i|}{|D|} H(D|\mathbf{X}^{(A)}=a_i)) \\
&= -\sum_{k=1}^{K} \frac{|C_k|}{|D|} log \frac{|C_k|}{|D|} - (-\sum_{i=1}^{n} \frac{|D_i|}{|D|} \sum_{k=1}^{K} \frac{|D_{ik}|}{|D_i|} log \frac{|D_{ik}|}{|D_i|}) \\
&= -\sum_{k=1}^{K} \frac{|C_k|}{|D|} log \frac{|C_k|}{|D|} + \sum_{i=1}^{n} \frac{|D_i|}{|D|} \sum_{k=1}^{K} \frac{|D_{ik}|}{|D_i|} log \frac{|D_{ik}|}{|D_i|}
\end{aligned}$$

那么我们对信息增益比的定义

$$g_R(D,A) = \frac{g(D,A)}{H_A(D)}$$

其中

$$H_A(D) = -\sum_{i=1}^{n} \frac{|D_i|}{|D|} log_2 \frac{|D_i|}{|D|}$$

在选择时，这两种算法都会有优化空间，所以我们用的更多的是接下来要讲解的 CART 算法。

3 CART 算法

CART 算法就是递归生成二叉决策树，对回归树使用平方误差最小化准则，对分类树使用基尼指数最小化准则来进行特征选择。在这里我仅做 CART 分类树的讨论。那么也就是通过计算基尼指数选择最优特征并决定最优二值切分点。基尼指数计算公式如下

$$Gini(p) = \sum_{k=1}^{K} p_k(1-p_k) = 1 - \sum_{k=1}^{K} p_k^2$$

对该公式的解读为：在公式中 $p_k(1-p_k)$ 表示我们随机抽取两个样本，得到其中一个属于 k 类，而另外一个不属于 k 类的概率，那么这个概率（乘积）越大，表征数据越分散，总体的不确定性越大。这里我们对于给定样本集合 D

$$Gini(D) = 1 - \sum_{k=1}^{K} (\frac{|C_k|}{|D|})^2$$

其中，C_k 是 D 中属于第 k 类的样本子集，K 是类的个数。那么这里，我们在样本集合 D 根据特征 A 是否可以被划分为 D_1, D_2 两个子集进行讨论

$$\begin{aligned}
Gini(D,A) &= \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \\
&= \sum_{i=1}^{2} \frac{|D_i|}{|D|} Gini(D_i) \\
&= \sum_{i=1}^{2} \frac{|D_i|}{|D|} [1 - \sum_{k=1}^{K} (\frac{|D_{ik}|}{|D_i|})^2]
\end{aligned}$$

其中 $D_1 = \{(x, y) \in D | A(x) = a\}; D_2 = D - D_1$，$D_{ik}$ 表示在数据集 D_i 中分类为 k 的数据子集，此处 D_1 和 D_2 是由特征 $X^{(A)} = a$ 来表征的数据集

那么划分数据集的 $Gini$ 指数可得。

4 Bagging

所谓 Bagging，我们可以这样思考：我们首先有 m 个样本，我们经过 T 次随机采样产生 T 个随机采样集，然后使用每个随机采样集进行独立训练生成 T 个弱学习器，然后 T 个弱学习器使用结合策略生成最终我们需要的强学习器。

这里我们的随机采样一般采用自助采样法 (Bootstrap Sampling)，也就是说对于 m 个样本的原始训练集，我们每次先随机采集一个样本放入采样集，接着把该样本放回，下次采样时该样本仍有可能被采集到，这样采集 m 次，最终可以得到 m 个样本的采样集，由于是随机采样，这样每次的采样集是和原始训练集不同的，和其他采样集也是不同的，这样得到多个不同的弱学习器。

那这里的结合策略在随机森林中选取投票法，即选取相对较多数的类别作为最终预测类别。

具体较为代码体现，所涉及数学如上。