

年 月 日

对于叶结点  $j$  的损失:

$$\tilde{L}_j^{(t)} = \left( \sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left( \sum_{i \in I_j} h_i + \lambda \right) w_j^2 + \gamma$$

对于当前的树结构求  $w_j$  使  $\tilde{L}_j^{(t)}$  最小, 显然这是一个一元二次方程求最小值问题。  $a = -\frac{b}{2a}$

可以得到叶结点权重  $w_j$  的最优值:

$$w_j^* = \frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

上面是对单个叶结点计算出了最优权重, 对于新建的这棵树 (树结构  $q$ ) 在此权重下对应的最小损失函数为每个叶节点上样本最小损失之和:

$$\tilde{L}^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^T \frac{\left( \sum_{i \in I_j} g_i \right)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T$$

2. (1) 决策树构建的终止条件是当前节点个数小于分裂所需最小样本个数或者深度大于 max\_depth 或者叶节点只有一类样本无需再分，满足其一

(2) 树的结构参数有

reg\_lambda = 1 # 叶节点权重 L2 正则系数

min\_samples\_split = 1 # 分裂所需的最小样本个数

max\_depth = 3 # 树的深度

(3) XGboost 算法与决策树算法在此处合在一起，通过 n\_estimators = 2 (可调) 来控制树的个数

3 因这次实验的数据量较大，不知道是不是因为我对数据做了正则化处理的原因，算法运行得很慢，所以可以看到我的算法中只测试了两个属性或三个属性参与运算的情况，树的个数通过 n\_estimators 设定在 2，我有通过改变树的个数，和选取更多不同的属性来测试，但效果一般，所以这里没有训练出较满意的模型。