决策树分裂过程详解

1 决策树分裂过程数学推导

1.1 核心公式

设节点D中共有N个样本，类别数为K，第k类的比例为p_k。

- **基尼指数（Gini）**

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

- **熵（Entropy）**

$$H(D) = -\sum_{k=1}^{K} p_k \log_2 p_k$$

- **分裂后的加权不纯度**

$$Impurity_{after} = \frac{N_L}{N} \cdot impurity(D_L) + \frac{N_R}{N} \cdot impurity(D_R)$$

- **不纯度下降**

$$\Delta = impurity(D) - Impurity_{after}$$

1.2 算法伪代码

Algorithm 1 决策树节点分裂算法

best_gain $\leftarrow -\infty$
best_feature $\leftarrow$ None
best_threshold $\leftarrow$ None
for each feature A **do**
  values $\leftarrow$ sorted(unique values of A in node)
  for $j \leftarrow 1$ to len(values)-1 **do**
    $t \leftarrow (values[j] + values[j+1])/2$
    split D into D_{left} $(A \leq t)$ and D_{right} $(A > t)$
    compute $impurity(D_{left})$, $impurity(D_{right})$
    weighted_after $\leftarrow \frac{|D_{left}|}{|D|} \cdot imp(D_{left}) + \frac{|D_{right}|}{|D|} \cdot imp(D_{right})$
    gain $\leftarrow imp(D) - weighted\_after$
    if gain $>$ best_gain **then**
      best_gain $\leftarrow$ gain
      best_feature, best_threshold $\leftarrow A, t$
    end if
  end for
end for
choose (best_feature, best_threshold) to split this node

1.3 数值计算示例

以Iris数据集的分裂点petal length ≤ 2.45为例：

- 父节点Gini：

$$Gini_{parent} = 1 - \left(\frac{1}{3}^2 + \frac{1}{3}^2 + \frac{1}{3}^2\right) = \frac{2}{3}$$

- 左节点（50个setosa）：

$$Gini_L = 1 - 1^2 = 0$$

- 右节点（50 versicolor + 50 virginica）：

$$Gini_R = 1 - \left(\frac{1}{2}^2 + \frac{1}{2}^2\right) = \frac{1}{2}$$

- 加权不纯度：

$$Weighted = \frac{1}{3} \cdot 0 + \frac{2}{3} \cdot \frac{1}{2} = \frac{1}{3}$$

- 基尼下降：

$$\Delta = \frac{2}{3} - \frac{1}{3} = \frac{1}{3} \approx 0.3333$$

2 代码解析

```
data = load_iris()
df = pd.DataFrame(data.data, columns=data.feature_names)
df[’Species’] = data.target

target = np.unique(data.target)             # [0,1,2]
target_names = np.unique(data.target_names) # [’setosa’,’versicolor’,’virginica’]
targets = dict(zip(target, target_names))   # {0:’setosa’,1:’versicolor’,2:’virginica
df[’Species’] = df[’Species’].replace(targets)

X = df.drop(columns="Species")
y = df["Species"]
feature_names = X.columns
labels = y.unique()

X_train, test_x, y_train, test_lab = train_test_split(
X, y, test_size=0.4, random_state=42)
```

- `df.shape`: (150, 5)（4个特征+1个标签）
- 训练集/测试集划分：
  - 训练集: 90样本 (60%)
  - 测试集: 60样本 (40%)
- `random_state=42`保证可复现性

3 参数说明

决策树关键参数：

- `criterion`: 'gini'或'entropy'
- `max_depth`: 控制树深度
- `min_samples_split`: 节点最小分裂样本数
- `min_samples_leaf`: 叶节点最小样本数