Data Mining 2025
Solutions Classification Trees

Exercise 1: Computing Splits

(a) The number of splits for a categorical variable with L distinct values is $2^{L-1} - 1$. So: $2^4 - 1 = 15$.

(b) Sort the values of x_1 on probability of class B (or A) and consider all splits between adjacent values in the sorted list. We find $P(B|x_1 = a) = \frac{1}{2}$, $P(B|x_1 = b) = \frac{2}{3}$, $P(B|x_1 = c) = 1$, $P(B|x_1 = d) = 0$, $P(B|x_1 = e) = \frac{1}{3}$. Hence, the order is

$$c \quad b \quad a \quad e \quad d$$

So the splits are: {c},{c,b},{c,b,a},{c,b,a,e}. Note: in our notation, the split {a,b} is the split that sends all cases with $x_1 \in \{a, b\}$ to one child node, and all cases with $x_1 \in \{c, d, e\}$ (the complement of $\{a, b\}$) to the other child node. Note also that it doesn't matter whether we sort on $P(B|x_1)$ or on $P(A|x_1)$, nor whether we sort ascending or descending. In all cases we get the same collection of splits.

(c) The sorted distinct values of x_2 are : 28,31,35,40,45,52,60. So the number of splits is 6 (all splits halfway between 2 adjacent distinct values in the sorted list)

(d) The best split can only occur on the border of a segment. To determine the segments we merge all values of the split attribute that are adjacent in the sorted list and have the same class distribution (i.e., the same relative frequencies for all classes). This gives the three segments: (28,31,35), (40,45), and (52,60). So we have to evaluate just 2 splits: $x_2 \leq 37.5$ and $x_2 \leq 48.5$.

(e) Both splits are equally good, and have an impurity reduction of

$$\frac{1}{4} - \left(\frac{3}{10} \times 0 \times 1 + \frac{7}{10} \times \frac{5}{7} \times \frac{2}{7} \right) = \frac{3}{28}$$

Exercise 2: More On Computing Splits

(a) The segments are (6,8), (12,14), and (18,20). So the candidate splits are $x \leq 10$ and $x \leq 16$.

(b) If we perform the split $x \leq 10$, the left child has class counts $(A:2, B:0, C:0)$ and the right child $(A:4, B:2, C:2)$. If we perform the split $x \leq 16$, the left child has class counts $(A:6, B:2, C:0)$ and the right child $(A:0, B:0, C:2)$. Obviously the second split is better. The impurity of the parent node is:

$$i(t) = 1 - \sum_j p(j|t)^2 = 1 - \left(\frac{6}{10}\right)^2 - \left(\frac{2}{10}\right)^2 - \left(\frac{2}{10}\right)^2 = 1 - \frac{44}{100} = \frac{56}{100} = \frac{14}{25}$$

The impurity of the left child is:

$$i(\ell) = 1 - \sum_j p(j|\ell)^2 = 1 - \left(\frac{6}{8}\right)^2 - \left(\frac{2}{8}\right)^2 - \left(\frac{0}{8}\right)^2 = 1 - \frac{40}{64} = \frac{24}{64} = \frac{3}{8}$$

The impurity of the right child is:

$$i(r) = 1 - \sum_j p(j|r)^2 = 1 - \left(\frac{0}{2}\right)^2 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right)^2 = 0$$

The impurity reduction is:

$$\Delta i = i(t) - \pi(\ell)i(\ell) - \pi(r)i(r) = \frac{14}{25} - \frac{4}{5} \times \frac{3}{8} - \frac{1}{5} \times 0 = \frac{14}{25} - \frac{12}{40} = \frac{112}{200} - \frac{60}{200} = \frac{26}{100}$$

(c) There is only one split that satisfies the `min_samples_leaf` constraint, namely $x \leq 13$. It is not on the border of a segment. If we have a `min_samples_leaf` constraint, the best split (that satisfies the constraint) no longer has to be on the border of a segment.

Exercise 3: Cost-Complexity Pruning

(a) When we prune in t_3, the node t_3 becomes a leaf node.

$$C_\alpha(T_{\max} - T_{t_3}) = R(t_4) + \alpha + R(t_5) + \alpha + R(t_3) + \alpha$$

(b) The leaf nodes of T_{t_3} are replaced by t_3, so $R(t_6) + R(t_8) + R(t_9) + 3\alpha$ is replaced by $R(t_3) + \alpha$.

(c) The costs are equal when

$$\begin{aligned} R(t_6) + R(t_8) + R(t_9) + 3\alpha &= R(t_3) + \alpha \\ 0 + 3\alpha &= \frac{1}{10} + \alpha \\ 2\alpha &= \frac{1}{10} \\ \alpha &= \frac{1}{20} \end{aligned}$$

When the costs are equal, the smaller tree is preferred.

(d) T_1 is equal to $T_{\max}$. In general, if we continue splitting until all leaf nodes are pure, then T_1 is equal to $T_{\max}$.

(e) Use the formula

$$g_k(t) = \frac{R(t) - R(T_{k,t})}{|\tilde{T}_{k,t}| - 1}$$

The subscript k indicates the iteration number of the pruning algorithm. In each iteration the nodes with minimum g values are pruned (indicated with a star in the table).

	t_1	t_2	t_3	t_7
$g_1(\cdot)$	$\frac{1}{8}$	$\frac{1}{20}^*$	$\frac{1}{20}^*$	$\frac{1}{10}$
$g_2(\cdot)$	$\frac{7}{20}^*$	–	–	–

Summarizing:

(1) T_1 is the smallest minimizing subtree for $\alpha \in [0, \frac{1}{20})$.

(2) T_2 is obtained by pruning T_1 in t_2 and t_3, and it is the best tree for $\alpha \in [\frac{1}{20}, \frac{7}{20})$.

(3) T_3 is obtained by pruning T_2 in t_1, and it is the best tree for $\alpha \in [\frac{7}{20}, \infty)$.

Exercise 4: Cost-Complexity Pruning

Since we continued splitting until all leaf nodes were pure, we have: $T_1 = T_{\max}$. The subscript of g indicates the iteration of the pruning algorithm:

	t_1	t_2	t_3	t_4	t_5
g_1	$\frac{2}{15}$	$\frac{2}{15}$	$\frac{2}{15}$	$\frac{2}{15}$	$\frac{1}{15}^*$
g_2	$\frac{3}{20}$	$\boxed{\frac{2}{15}^*}$	$\boxed{\frac{2}{15}^*}$	$\frac{6}{30}$	–
g_3	$\frac{1}{6}^*$	–	–	$\boxed{\frac{6}{30}}$	–

We prune in the nodes with the starred values. The boxed values did not need to recomputed; they were copied from the previous iteration. Summarizing: T_2 is obtained from T_1 by pruning in node t_5. T_3 is obtained from T_2 by pruning in t_3 and t_2, and $T_4 = \{t_1\}$. The α- intervals are: $T_1 : [0, \frac{1}{15})$, $T_2 : [\frac{1}{15}, \frac{2}{15})$, $T_3 : [\frac{2}{15}, \frac{1}{6})$, and $T_4 : [\frac{1}{6}, \infty)$.

Exercise 5: An Alternative Pruning Procedure

(a)
1. T_1: prune $T_{\max}$ in t_2.
2. T_2: prune $T_{\max}$ in t_3.
3. T_3: prune $T_{\max}$ in t_2 and t_3.
4. T_4: prune $T_{\max}$ in t_1.

(b) No, T_2 is not a subtree of T_1. In general, with this pruning method the pruning sequence may not be nested: as we go through the sequence, nodes may reappear that were previously cut off. Computation of the sequence can be performed by dynamic programming, but it is more complex than the cost-complexity pruning algorithm. Therefore Breiman et al. discarded this pruning method.

(c) Yes, it is a subsequence. Suppose there is another tree T with m leaf nodes such that

$$R(T) \leq R(T(\alpha)). \tag{1}$$

Since T and $T(\alpha)$ have the same number of leaf nodes, this implies that

$$C_\alpha(T) \leq C_\alpha(T(\alpha)). \tag{2}$$

But this contradicts the fact that $T(\alpha)$ is the smallest minimizing subtree.

Exercise 6: Gini index

We have defined the gini index for binary classification as

$$i(t) = p(0|t)p(1|t) = p(0|t)(1 - p(0|t)), \tag{3}$$

where the class values are coded as 0 and 1, and $p(j|t)$ denotes the relative frequency of class j in node t. The generalization to an arbitrary number of classes is given by:

$$i(t) = \sum_{j=1}^{C} p(j|t)(1 - p(j|t)), \tag{4}$$

where C denotes the number of classes. If we apply equation (4) to the binary case, we should get the same results as when we apply equation (3). Is this indeed the case?

Call impurity according to equation (3) i_1, and according to equation (4) i_2. For the binary case, we have

$$i_2(t) = p(0|t)(1 - p(0|t)) + p(1|t)(1 - p(1|t)) = 2\, p(0|t)(1 - p(0|t)) = 2\, i_1(t)$$

So impurity according to equation (4) is twice as large as according to equation (3). Therefore, also $\Delta i_2 = 2\, \Delta i_1$. This makes no difference in determining the optimal split because it is only the order of the values that matters. The same split will win.

Show that equation (4) can alternatively be written as

$$i(t) = 1 - \sum_{j=1}^{C} p(j|t)^2.$$

Starting with (4):

$$\begin{aligned} i(t) &= \sum_{j=1}^{C} p(j|t)(1 - p(j|t)) \\ &= \sum_{j=1}^{C} p(j|t) - p(j|t)^2 \\ &= \sum_{j=1}^{C} p(j|t) - \sum_{j=1}^{C} p(j|t)^2 \\ &= 1 - \sum_{j=1}^{C} p(j|t)^2. \end{aligned}$$

Exercise 7: More about the Gini index

(a) The expected value of Bernoulli random variable X is:

$$\mathbb{E}[X] = \sum_x x \times P(X = x) = 0 \times P(X = 0) + 1 \times P(X = 1) = 0 \times (1-p) + 1 \times p = p.$$

(b) Its variance is:

$$\begin{aligned} \mathbb{V}[X] = \mathbb{E}[(X - \mathbb{E}[X])^2] &= \sum_x (x - p)^2 \times P(X = x) \\ &= (0 - p)^2(1 - p) + (1 - p)^2 p \\ &= p^2(1 - p) + p(1 - p)^2 = (1 - p)\left(p^2 + p(1 - p)\right) \\ &= p(1 - p) \end{aligned}$$

Notice that we used the rule that $\mathbb{E}[f(X)] = \sum_x f(x) \times P(X = x)$ with $f(X) = (X - p)^2$.

(c) The derivative is:

$$\phi'(p) = 1 - 2p$$

Equating to zero gives:

$$1 - 2p = 0 \Rightarrow p = \tfrac{1}{2}$$

This is a maximum, since the second derivative is negative in this point (in fact, it is negative everywhere, see (d))

(d) Since $\phi'(p) = 1 - 2p$, we have $\phi''(p) = -2$. Since the second derivative is negative everywhere, the function is strictly concave.

Exercise 8: The distribution of a sample proportion

(a) The expected value is:

$$\begin{aligned}
\mathbb{E}[Y] &= \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] \\
&= \frac{1}{n}\mathbb{E}\left[\sum_{i=1}^{n} X_i\right] && (\mathbb{E}[cX] = c\mathbb{E}[X]) \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[X_i] && (\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]) \\
&= \frac{1}{n}np = p. && (\mathbb{E}[X_i] = p)
\end{aligned}$$

(b) The variance is:

$$\begin{aligned}
\mathbb{V}[Y] &= \mathbb{V}\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] \\
&= \frac{1}{n^2}\mathbb{V}\left[\sum_{i=1}^{n} X_i\right] && (\mathbb{V}[cX] = c^2\mathbb{V}[X]) \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\mathbb{V}[X_i] && (\mathbb{V}[X+Y] = \mathbb{V}[X] + \mathbb{V}[Y] \text{ if } X \text{ and } Y \text{ independent}) \\
&= \frac{1}{n^2}np(1-p) && (\mathbb{V}[X_i] = p(1-p)) \\
&= \frac{p(1-p)}{n}.
\end{aligned}$$

Exercise 9: Splitting can not increase impurity

Let's first introduce some notation. Let $N(t)$ denote the number of observations in node t, and let $N(t, 0)$ denote the number of observations in node t with class label 0. Then we can write the given equality as:

$$\frac{N(t,0)}{N(t)} = \frac{N(\ell)}{N(t)}\frac{N(\ell,0)}{N(\ell)} + \frac{N(r)}{N(t)}\frac{N(r,0)}{N(r)}$$

After cancelling terms, we are left with the equality

$$N(t,0) = N(\ell,0) + N(r,0),$$

which we decide to accept without further proof.