

Data Mining and Machine Learning: Fundamental Concepts and Algorithms

dataminingbook.info

Mohammed J. Zaki¹ Wagner Meira Jr.²

¹Department of Computer Science
Rensselaer Polytechnic Institute, Troy, NY, USA

²Department of Computer Science
Universidade Federal de Minas Gerais, Belo Horizonte, Brazil

Chapter 19: Decision Tree Classifier

Decision Tree Classifier

Let the training dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$ consist of n points in a d -dimensional space, with y_i being the class label for point \mathbf{x}_i .

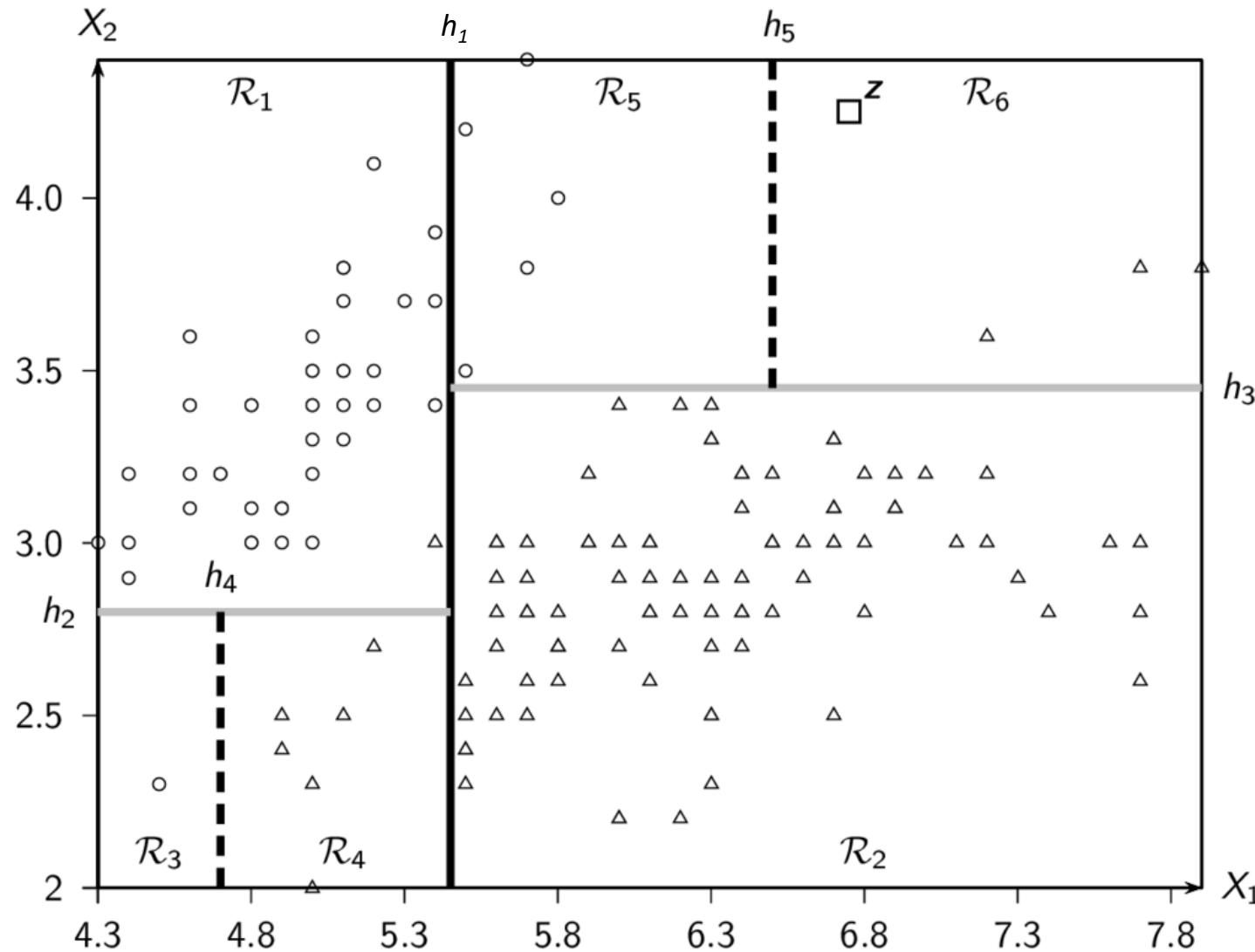
A decision tree classifier is a recursive, partition-based tree model that predicts the class \hat{y}_i for each point \mathbf{x}_i .

Let \mathcal{R} denote the data space that encompasses the set of input points \mathcal{D} . A decision tree uses an axis-parallel hyperplane to split the data space \mathcal{R} into two resulting half-spaces or regions, say \mathcal{R}_1 and \mathcal{R}_2 , which also induces a partition of the input points into \mathcal{D}_1 and \mathcal{D}_2 , respectively.

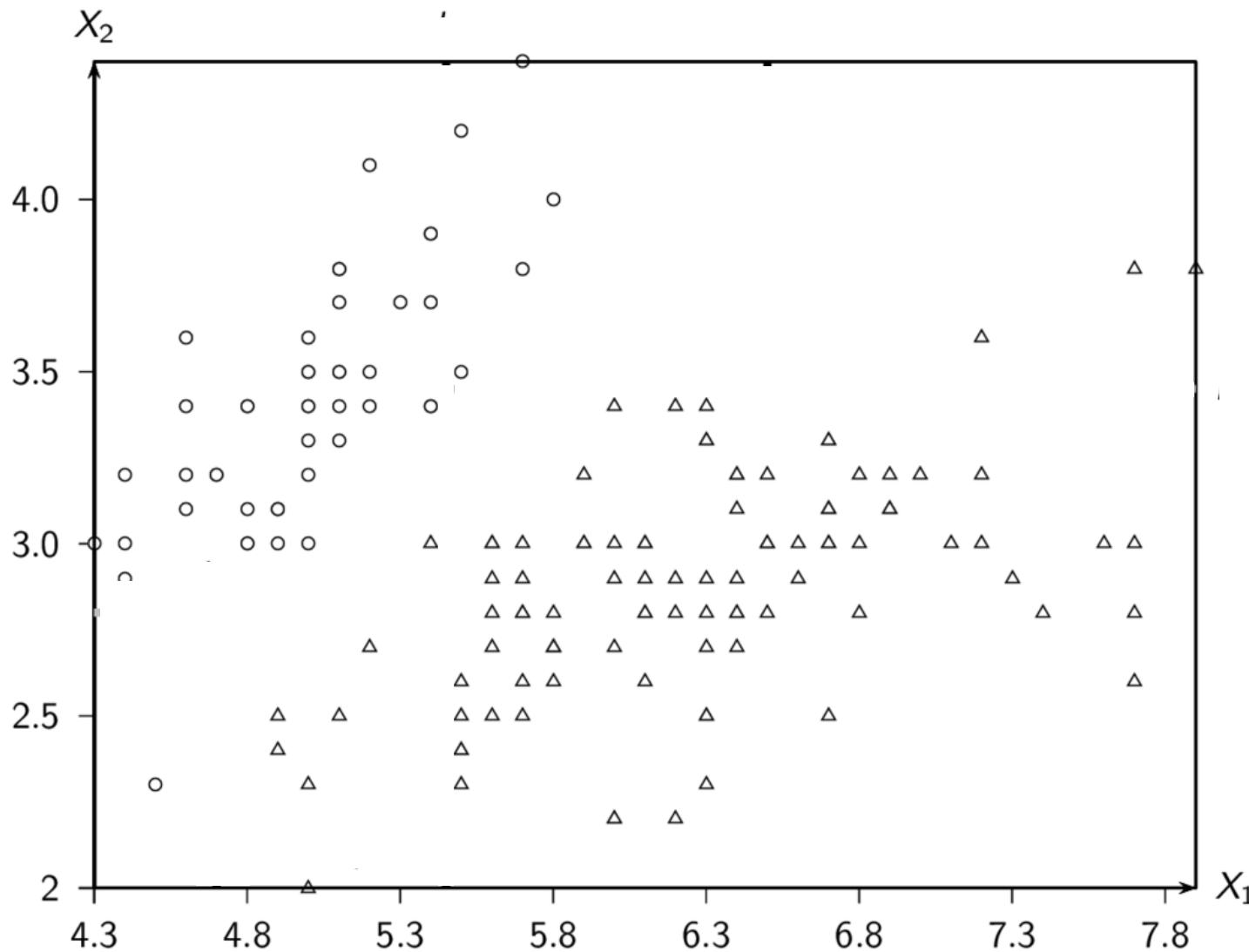
Each of these regions is recursively split via axis-parallel hyperplanes until most of the points belong to the same class.

To classify a new *test* point we have to recursively evaluate which half-space it belongs to until we reach a leaf node in the decision tree, at which point we predict its class as the label of the leaf.

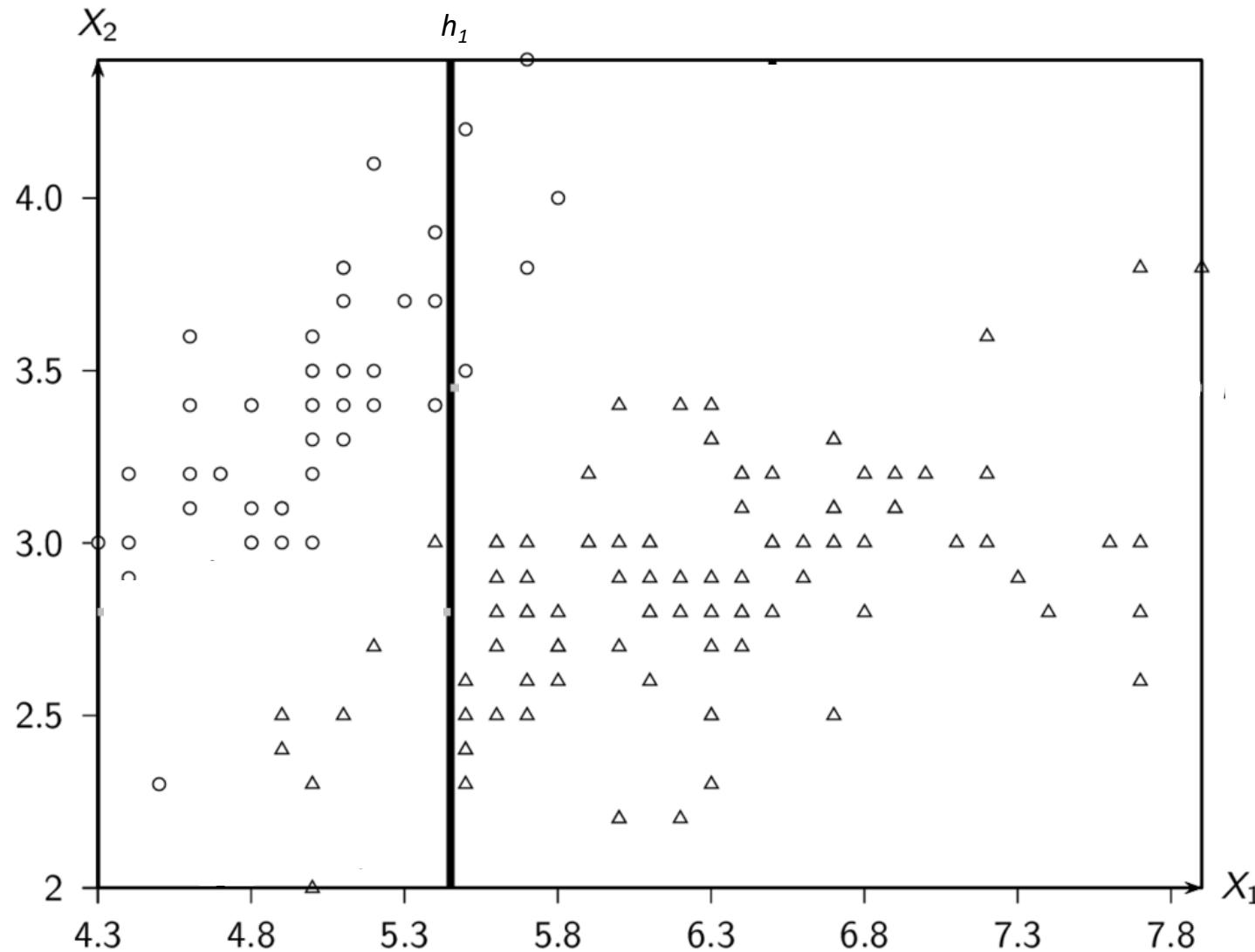
Decision Tree: Recursive Splits



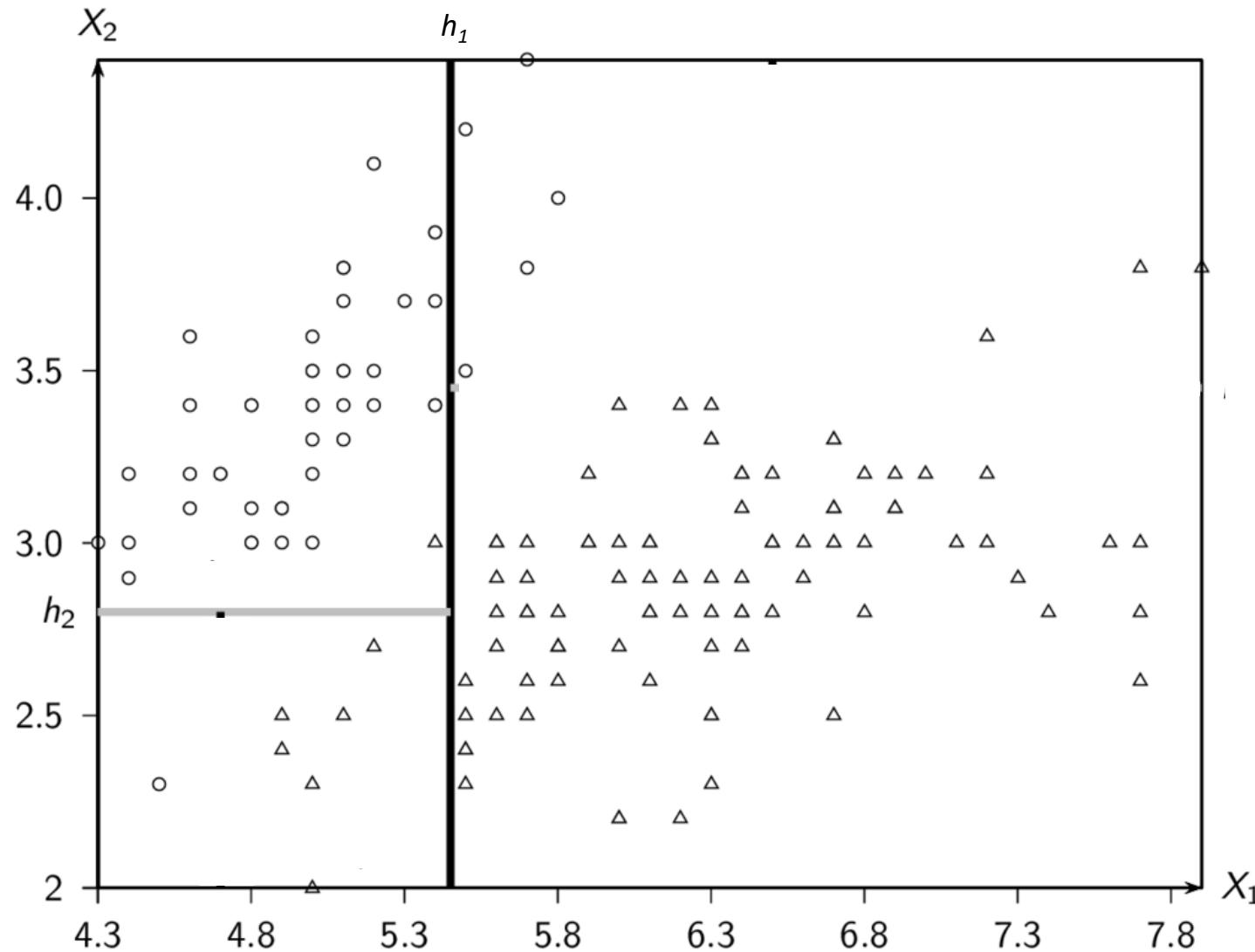
Decision Tree: Recursive Splits



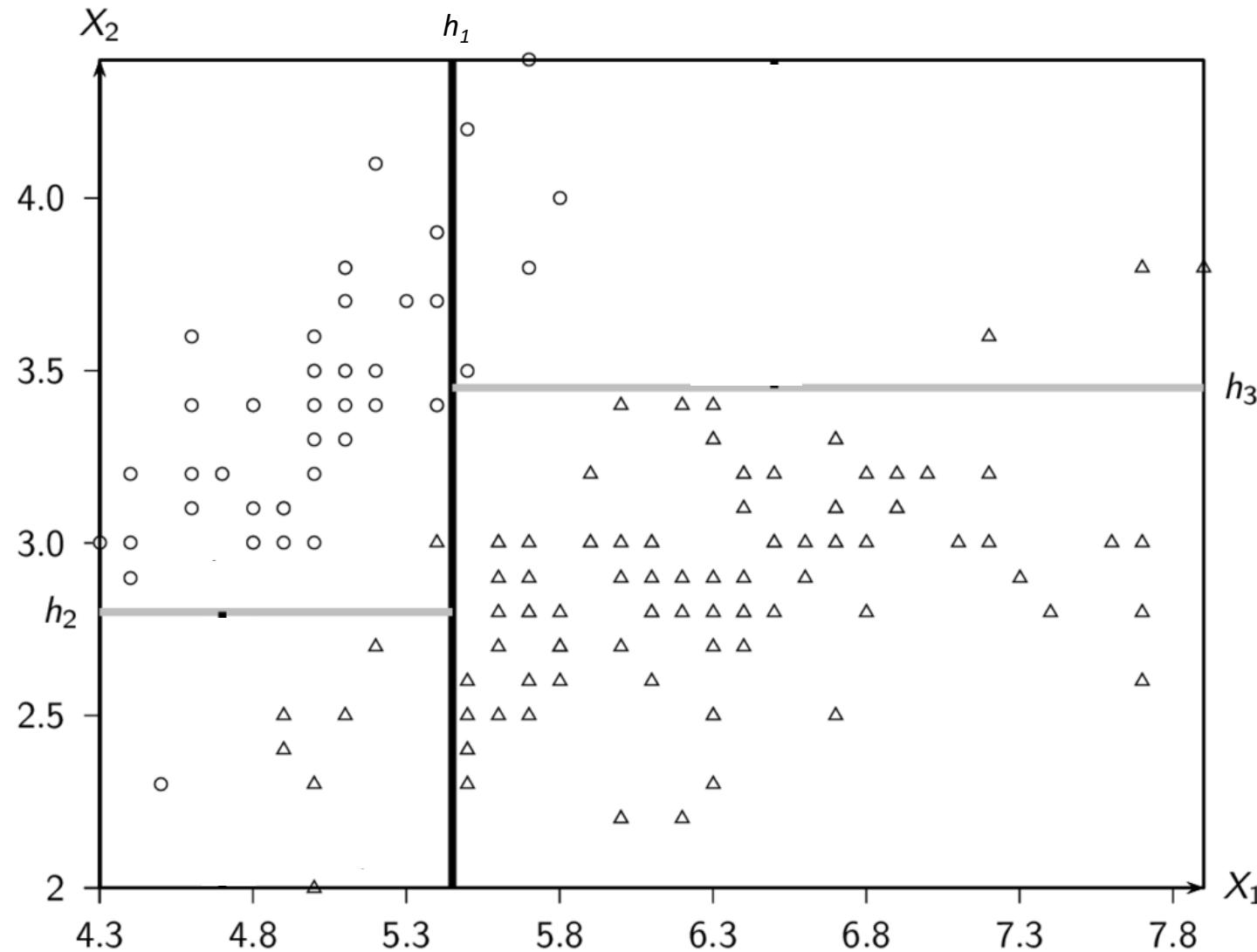
Decision Tree: Recursive Splits



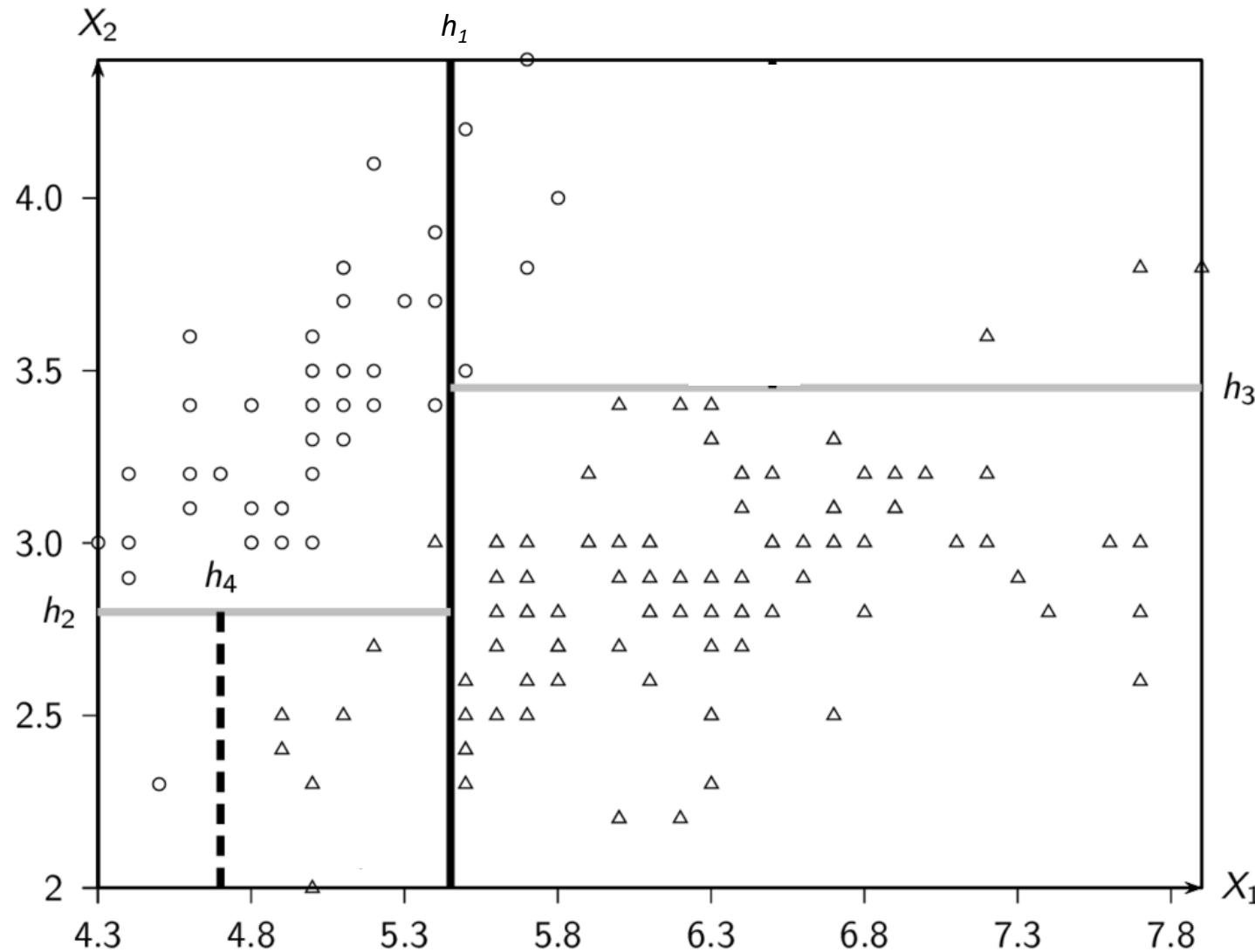
Decision Tree: Recursive Splits



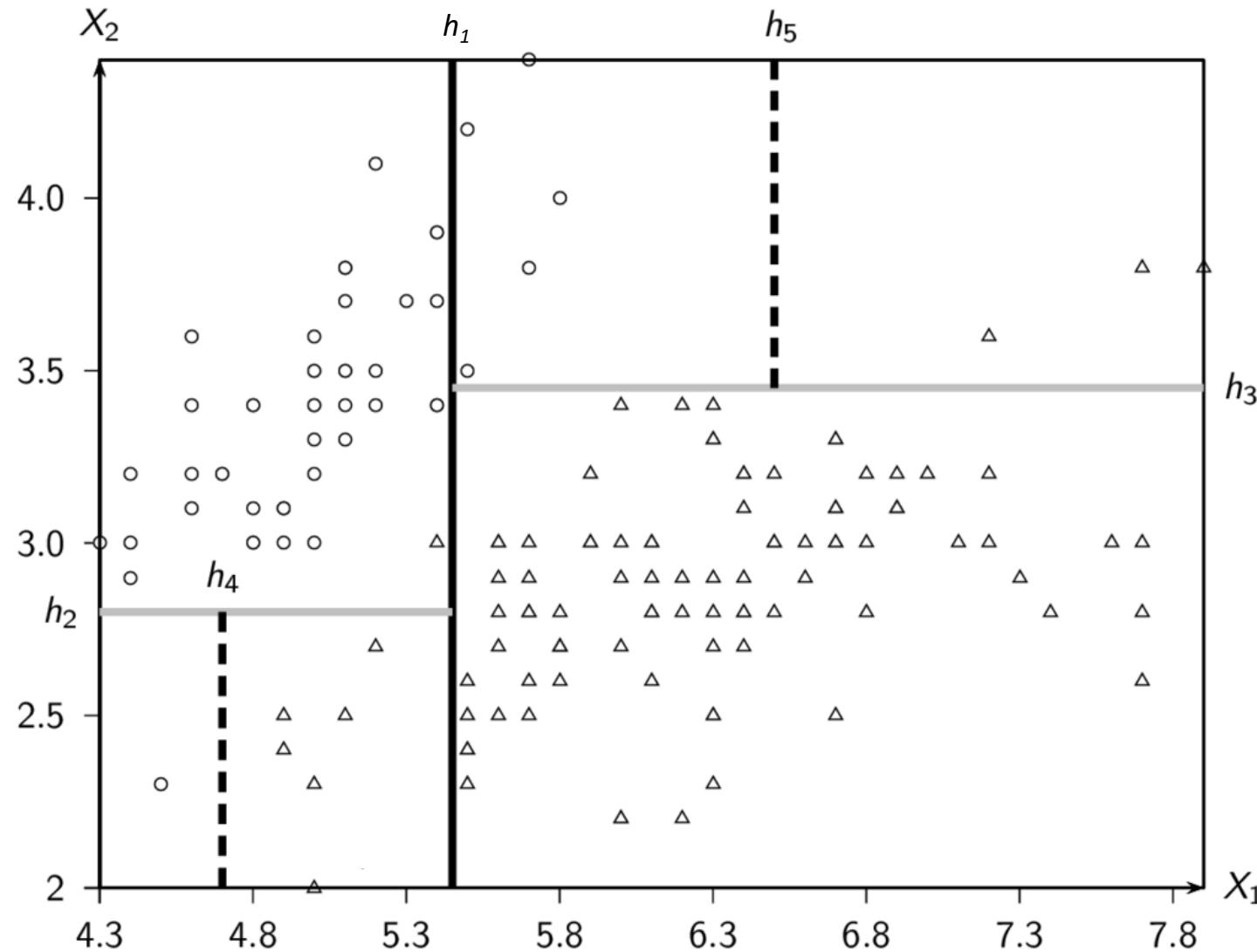
Decision Tree: Recursive Splits



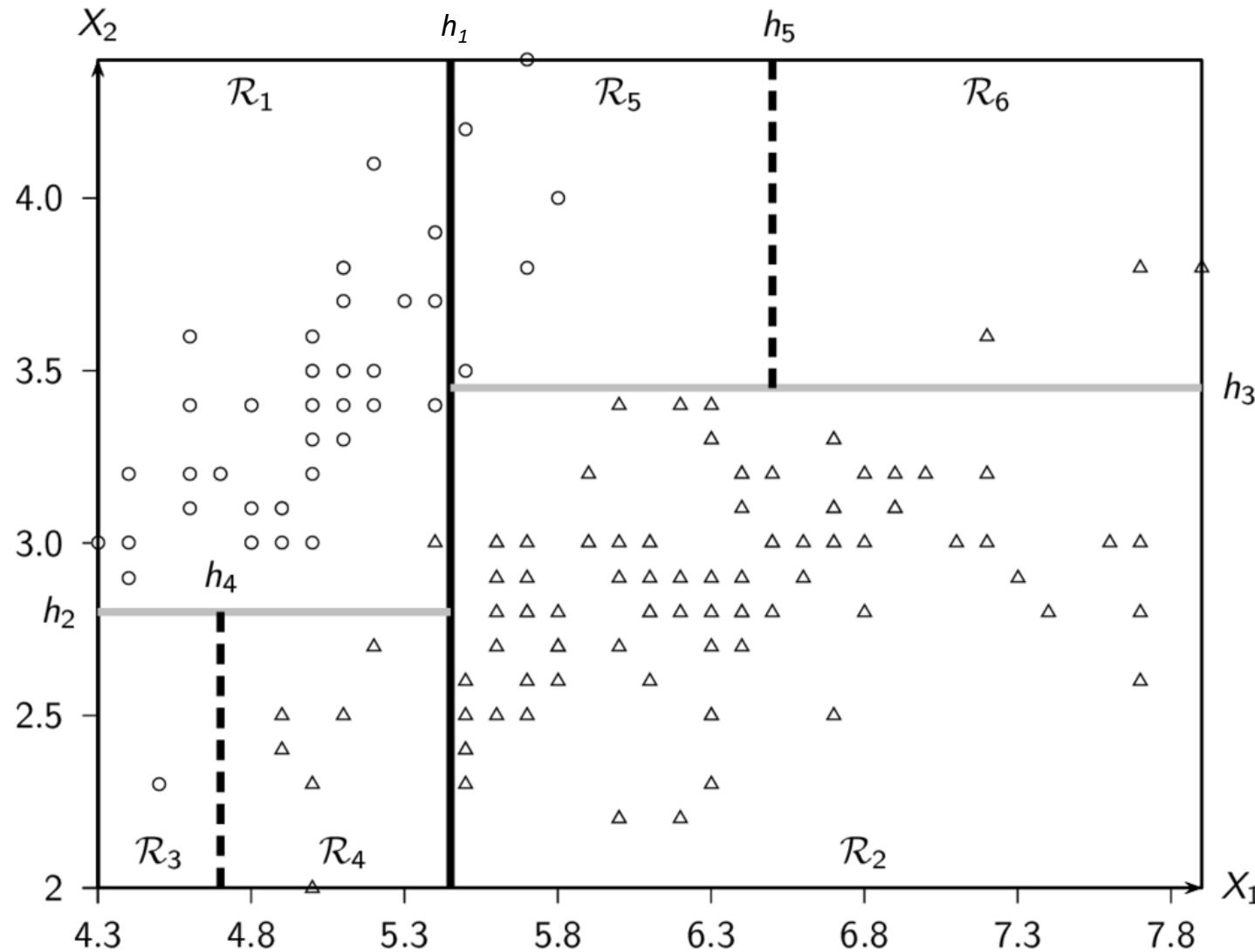
Decision Tree: Recursive Splits



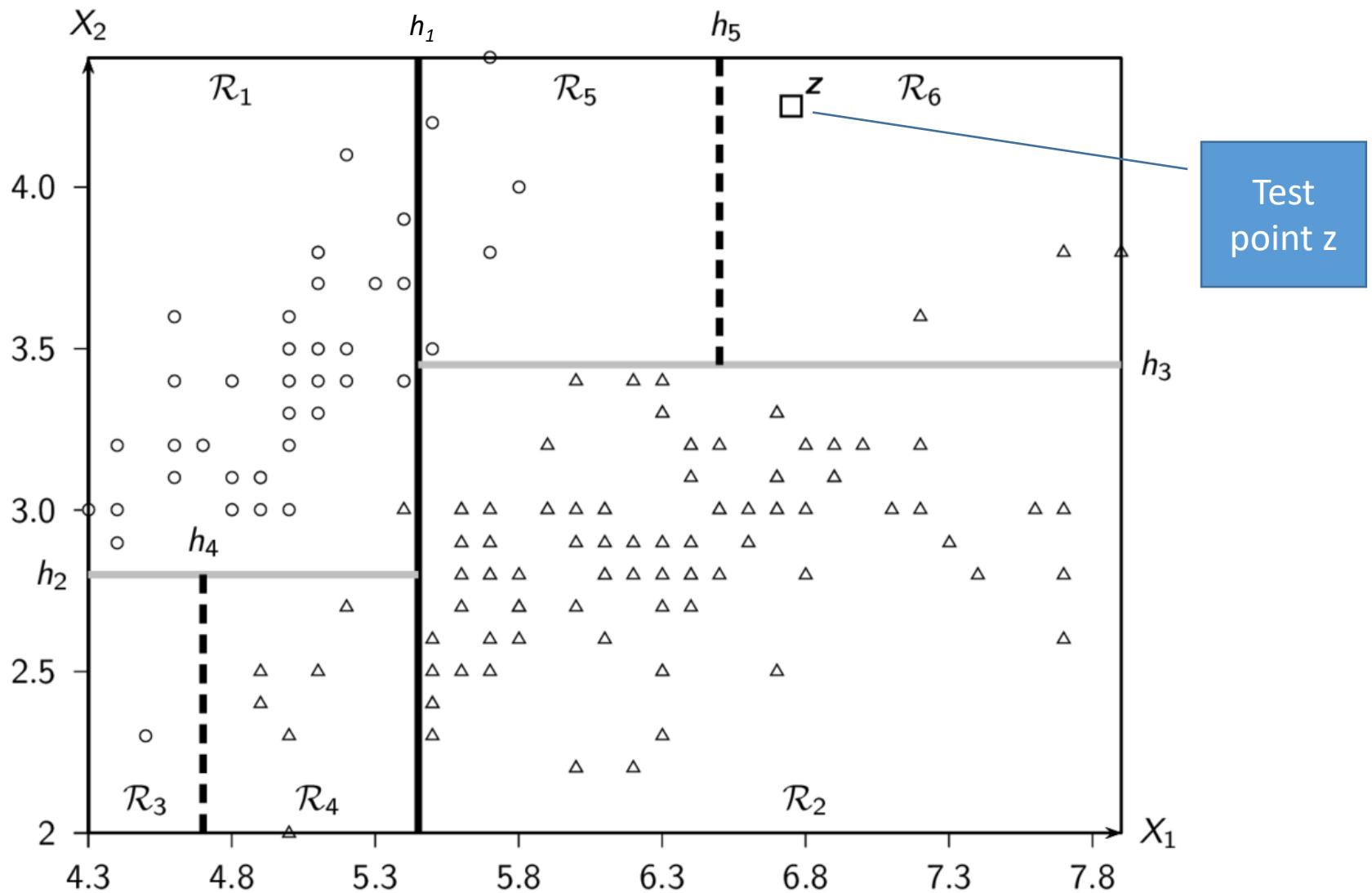
Decision Tree: Recursive Splits



Decision Tree: Recursive Splits

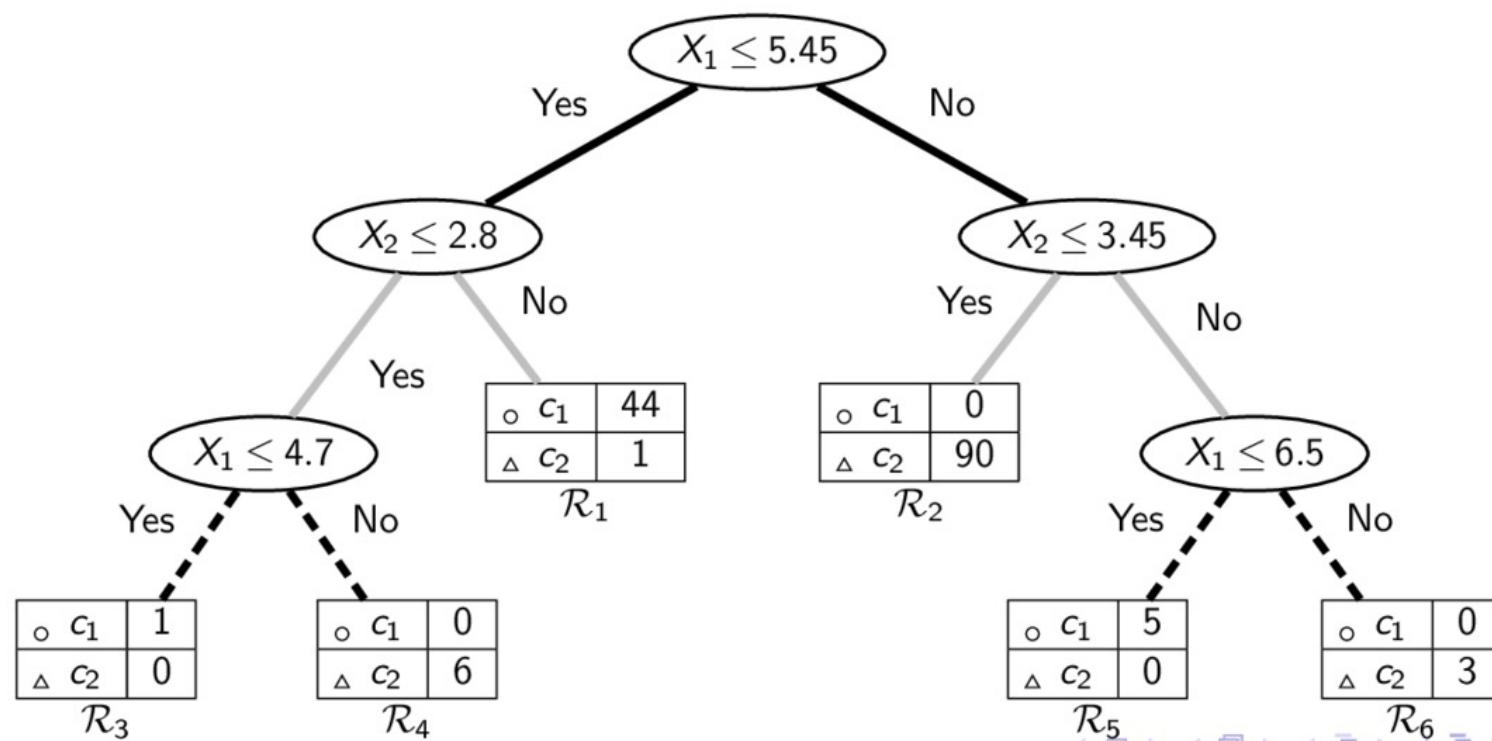


Decision Tree: Recursive Splits



Decision Tree

A decision tree consists of internal nodes that represent the decisions corresponding to the hyperplanes or split points (i.e., which half-space a given point lies in), and leaf nodes that represent regions or partitions of the data space, which are labeled with the majority class. A region is characterized by the subset of data points that lie in that region.



Decision Trees: Axis-Parallel Hyperplanes

A hyperplane $h(\mathbf{x})$ is defined as the set of all points \mathbf{x} that satisfy the following equation

$$h(\mathbf{x}): \mathbf{w}^T \mathbf{x} + b = 0$$

where $\mathbf{w} \in \mathbb{R}^d$ is a *weight vector* that is normal to the hyperplane, and b is the offset of the hyperplane from the origin.

A decision tree considers only *axis-parallel hyperplanes*, that is, the weight vector must be parallel to one of the original dimensions or axes X_j :

$$h(\mathbf{x}): x_j + b = 0$$

where the choice of the offset b yields different hyperplanes along dimension X_j .

Decision Trees: Split Points

A hyperplane specifies a decision or *split point* because it splits the data space \mathcal{R} into two half-spaces. All points \mathbf{x} such that $h(\mathbf{x}) \leq 0$ are on the hyperplane or to one side of the hyperplane, whereas all points such that $h(\mathbf{x}) > 0$ are on the other side.

The split point is written as $h(\mathbf{x}) \leq 0$, i.e.

$$X_j \leq v$$

where $v = -b$ is some value in the domain of attribute X_j .

The decision or split point $X_j \leq v$ thus splits the input data space \mathcal{R} into two regions \mathcal{R}_Y and \mathcal{R}_N , which denote the set of *all possible points* that satisfy the decision and those that do not.

Categorical Attributes: For a categorical attribute X_j , the split points or decisions are of the $X_j \in V$, where $V \subset \text{dom}(X_j)$, and $\text{dom}(X_j)$ denotes the domain for X_j .

Decision Trees: Data Partition and Purity

Each split of \mathcal{R} into \mathcal{R}_Y and \mathcal{R}_N also induces a binary partition of the corresponding input data points \mathbf{D} . A split point of the form $X_j \leq v$ induces the data partition

$$\mathbf{D}_Y = \{\mathbf{x} \mid \mathbf{x} \in \mathbf{D}, x_j \leq v\}$$

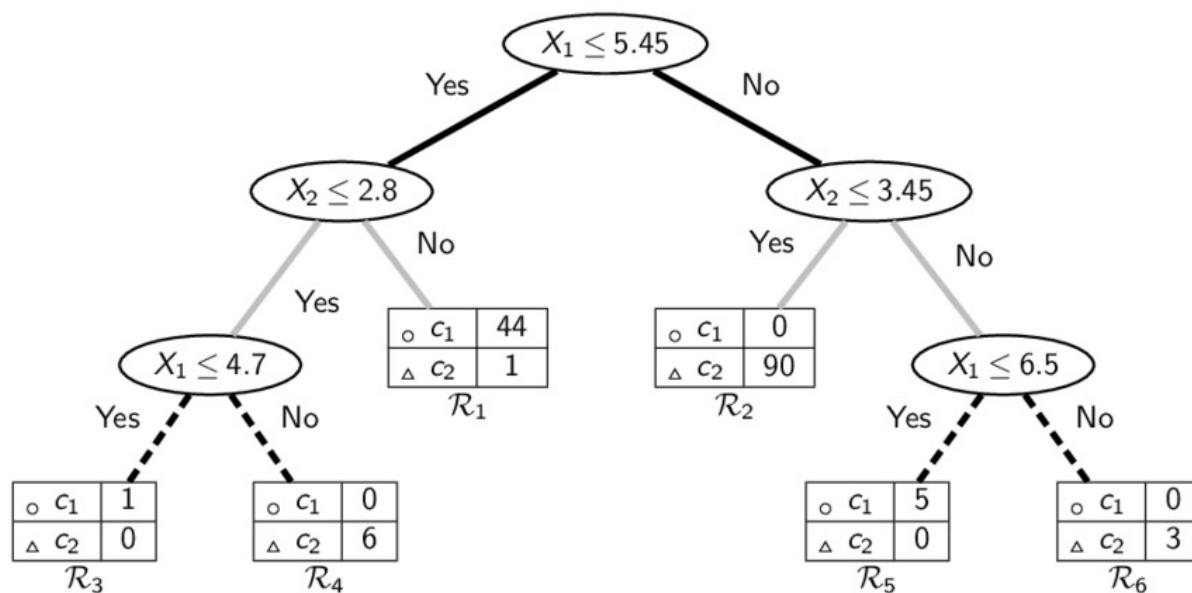
$$\mathbf{D}_N = \{\mathbf{x} \mid \mathbf{x} \in \mathbf{D}, x_j > v\}$$

The purity of a region \mathcal{R}_j is the fraction of points with the majority label in \mathbf{D}_j , that is,

$$purity(\mathbf{D}_j) = \max_i \left\{ \frac{n_{ji}}{n_j} \right\}$$

where $n_j = |\mathbf{D}_j|$ is the total number of data points in the region \mathcal{R}_j , and n_{ji} is the number of points in \mathbf{D}_j with class label c_i .

Decision Trees to Rules



A tree is a set of decision rules; each comprising the decisions on the path to a leaf:

- \mathcal{R}_3 : If $X_1 \leq 5.45$ and $X_2 \leq 2.8$ and $X_1 \leq 4.7$, then class is c_1 , or
- \mathcal{R}_4 : If $X_1 \leq 5.45$ and $X_2 \leq 2.8$ and $X_1 > 4.7$, then class is c_2 , or
- \mathcal{R}_1 : If $X_1 \leq 5.45$ and $X_2 > 2.8$, then class is c_1 , or
- \mathcal{R}_2 : If $X_1 > 5.45$ and $X_2 \leq 3.45$, then class is c_2 , or
- \mathcal{R}_5 : If $X_1 > 5.45$ and $X_2 > 3.45$ and $X_1 \leq 6.5$, then class is c_1 , or
- \mathcal{R}_6 : If $X_1 > 5.45$ and $X_2 > 3.45$ and $X_1 > 6.5$, then class is c_2

Decision Tree Algorithm

The method takes as input a training dataset \mathcal{D} , and two parameters η and π , where η is the leaf size and π the leaf purity threshold.

Different split points are evaluated for each attribute in \mathcal{D} . Numeric decisions are of the form $X_j \leq v$ for some value v in the value range for attribute X_j , and categorical decisions are of the form $X_j \in V$ for some subset of values in the domain of X_j .

The best split point is chosen to partition the data into two subsets, \mathcal{D}_Y and \mathcal{D}_N , where \mathcal{D}_Y corresponds to all points $x \in \mathcal{D}$ that satisfy the split decision, and \mathcal{D}_N corresponds to all points that do not satisfy the split decision. The decision tree method is then called recursively on \mathcal{D}_Y and \mathcal{D}_N .

We stop the process if the leaf size drops below η or if the purity is at least π .

Decision Tree Algorithm

DecisionTree (D, η, π):

```
1  $n \leftarrow |D|$  // partition size
2  $n_i \leftarrow |\{x_j | x_j \in D, y_j = c_i\}|$  // size of class  $c_i$ 
3  $purity(D) \leftarrow \max_i \left\{ \frac{n_i}{n} \right\}$ 
4 if  $n \leq \eta$  or  $purity(D) \geq \pi$  then // stopping condition
5    $c^* \leftarrow \arg \max_{c_i} \left\{ \frac{n_i}{n} \right\}$  // majority class
6   create leaf node, and label it with class  $c^*$ 
7   return
8  $(split point^*, score^*) \leftarrow (\emptyset, 0)$  // initialize best split point
9 foreach (attribute  $X_j$ ) do
10   if ( $X_j$  is numeric) then
11      $(v, score) \leftarrow \text{Evaluate-Numeric-Attribute}(D, X_j)$ 
12     if  $score > score^*$  then  $(split point^*, score^*) \leftarrow (X_j \leq v, score)$ 
13   else if ( $X_j$  is categorical) then
14      $(V, score) \leftarrow \text{Evaluate-Categorical-Attribute}(D, X_j)$ 
15     if  $score > score^*$  then  $(split point^*, score^*) \leftarrow (X_j \in V, score)$ 
16  $D_Y \leftarrow \{x \in D | x \text{ satisfies } split point^*\}$ 
17  $D_N \leftarrow \{x \in D | x \text{ does not satisfy } split point^*\}$ 
18 create internal node  $split point^*$ , with two child nodes,  $D_Y$  and  $D_N$ 
19 DecisionTree( $D_Y, \eta, \pi$ ); DecisionTree( $D_N, \eta, \pi$ )
```

Split Point Evaluation Measures: Entropy

Intuitively, we want to select a split point that gives the best separation or discrimination between the different class labels.

Entropy measures the amount of disorder or uncertainty in a system. A partition has lower entropy (or low disorder) if it is relatively pure, that is, if most of the points have the same label. On the other hand, a partition has higher entropy (or more disorder) if the class labels are mixed, and there is no majority class as such.

The entropy of a set of labeled points \mathbf{D} is defined as follows:

$$H(\mathbf{D}) = - \sum_{i=1}^k P(c_i|\mathbf{D}) \log_2 P(c_i|\mathbf{D})$$

where $P(c_i|\mathbf{D})$ is the probability of class c_i in \mathbf{D} , and k is the number of classes.

If a region is pure, that is, has points from the same class, then the entropy is zero. On the other hand, if the classes are all mixed up, and each appears with equal probability $P(c_i|\mathbf{D}) = \frac{1}{k}$, then the entropy has the highest value, $H(\mathbf{D}) = \log_2 k$.

Split Point Evaluation Measures: Entropy

Define the *split entropy* as the weighted entropy of each of the resulting partitions

$$H(\mathbf{D}_Y, \mathbf{D}_N) = \frac{n_Y}{n} H(\mathbf{D}_Y) + \frac{n_N}{n} H(\mathbf{D}_N)$$

where $n = |\mathbf{D}|$ is the number of points in \mathbf{D} , and $n_Y = |\mathbf{D}_Y|$ and $n_N = |\mathbf{D}_N|$ are the number of points in \mathbf{D}_Y and \mathbf{D}_N .

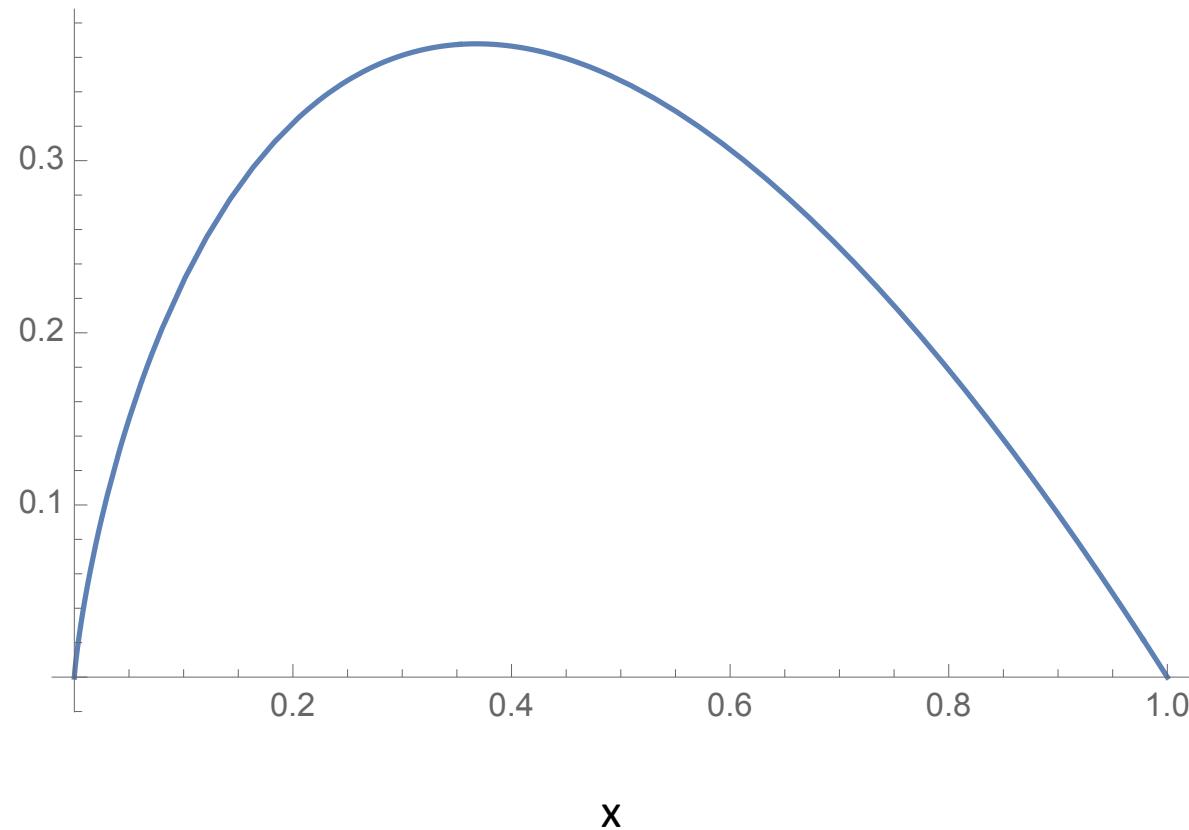
Define the *information gain* for a split point as

$$Gain(\mathbf{D}, \mathbf{D}_Y, \mathbf{D}_N) = H(\mathbf{D}) - H(\mathbf{D}_Y, \mathbf{D}_N)$$

The higher the information gain, the more the reduction in entropy, and the better the split point.

We score each split point and choose the one that gives the highest information gain.

$-x \log(x)$



Relative Entropy in Decision Trees

- Data D , size $|D| = n$
- Classes $c_i : i=1,2,\dots,k$
- Decision partitions D_Y, D_N : $D = D_Y \cup D_N$, sizes $n_Y = |D_Y|$ and $n_N = |D_N|$
- Information Gain:

$$H(D) - (n_Y / n) H(D_Y) - (n_N / n) H(D_N)$$

- Entropy:
 - $H(D) = -\sum_i p_i \log p_i \quad p_i = \Pr[c_i \mid D]$
 - similarly for D_Y, D_N

Relative Entropy in Decision Trees

- Define p_{ix} = fraction of data in D that is in partition D_x and in class i
- $q_i = p_{iY} + p_{iN}$ = fraction from D in class i = $\Pr[c_i | D]$
- $r_x = \sum_i p_{ix}$ = fraction from D in partition D_x = n_x / n
- $\Pr[c_i | D_x] = \Pr[c_i \cap D_x] / \Pr[D_x] = p_{ix} / r_x$
- Information Gain:

$$\begin{aligned} H(D) - (n_Y/n) H(D_Y) - (n_N/n) H(D_N) &= -\sum_i q_i \log q_i + \sum_x r_x \sum_i (p_{ix} / r_x) \log (p_{ix} / r_x) \\ &= -\sum_x \sum_i p_{ix} \log q_i + \sum_x \sum_i p_{ix} \log (p_{ix} / r_x) \\ &= \sum_x \sum_i p_{ix} \log (p_{ix} / q_i r_x) \\ &= \text{Rel. Entropy of } p_{ix} \text{ w.r.t product } q_i r_x \end{aligned}$$

- Information gain = Relative Entropy is nonnegative (by Gibbs' Inequality)
- Worst split:
 - $p_{ix} = q_i r_x$ class conditional probabilities independent on $x = Y$ or N

Gibbs' Inequality

- Theorem

If: p and q are two probability distributions with $q_i = 0 \rightarrow p_i = 0$

Then: $D(p,q) = \sum_i p_i \log (p_i / q_i) \geq 0$ with equality iff $p = q$

- Proof:

$g(x) = \log(x)$ a concave function,

derivative $1/x$ is decreasing

$$\log x - \log 1 \leq g'(1)(x-1) = x-1$$

with equality only if $x = 1$.

$$-\log x = \log(1/x) < 1/x - 1 \text{ so } \log x > 1 - 1/x$$

$$\log p/q > 1 - q/p$$

$$\sum_i p_i \log (p_i / q_i) \geq \sum_i p_i (1 - q_i / p_i) = \sum_i p_i - q_i = 1 - 1 = 0$$

with equality only if $p_i = q_i$ for all i

Proof of Gibbs Inequality

- Want to show $D(p,q) = \sum_i p_i \log (p_i / q_i) \geq 0$
- $g(x) = \log(x)$ a concave function,
 - derivative $1/x$ is decreasing
- $\log x - \log 1 \leq g'(1)(x-1) = x-1$
 - with equality only if $x=1$.
- $-\log x = \log(1/x) < 1/x - 1$ so $\log x > 1 - 1/x$
- $\log p/q > 1 - q/p$
- $\sum_i p_i \log (p_i / q_i) \geq \sum_i p_i (1 - q_i / p_i) = \sum_i p_i - q_i = 1 - 1 = 0$
with equality only if $p_i = q_i$ for all i

Split Point Evaluation Measures

Gini Index and CART Measure

Gini Index: The Gini index is defined as follows:

$$G(\mathbf{D}) = 1 - \sum_{i=1}^k P(c_i|\mathbf{D})^2$$

If the partition is pure, then Gini index is 0. The weighted Gini index is:

$$G(\mathbf{D}_Y, \mathbf{D}_N) = \frac{n_Y}{n} G(\mathbf{D}_Y) + \frac{n_N}{n} G(\mathbf{D}_N)$$

The lower the Gini index value, the better the split point.

CART: The CART measure is

$$CART(\mathbf{D}_Y, \mathbf{D}_N) = 2 \frac{n_Y}{n} \frac{n_N}{n} \sum_{i=1}^k \left| P(c_i|\mathbf{D}_Y) - P(c_i|\mathbf{D}_N) \right|$$

CART maximizes the difference between the class PMF for the two partitions; the higher the CART measure, the better the split point.

Evaluating Split Points: Numeric Attributes

All of the split point evaluation measures depend on the class probability mass function (PMF) for \mathbf{D} , namely, $P(c_i|\mathbf{D})$, and the class PMFs for the resulting partitions \mathbf{D}_Y and \mathbf{D}_N , namely $P(c_i|\mathbf{D}_Y)$ and $P(c_i|\mathbf{D}_N)$.

We have to evaluate split points of the form $X \leq v$. We consider only the midpoints between two successive distinct values for X in the sample \mathbf{D} . Let $\{v_1, \dots, v_m\}$ denote the set of all such midpoints, such that $v_1 < v_2 < \dots < v_m$.

For each split point $X \leq v$, we have to estimate the class PMFs:

$$\hat{P}(c_i|\mathbf{D}_Y) = \hat{P}(c_i|X \leq v)$$

$$\hat{P}(c_i|\mathbf{D}_N) = \hat{P}(c_i|X > v)$$

Using Bayes theorem, we have

$$\hat{P}(c_i|X \leq v) = \frac{\hat{P}(X \leq v|c_i)\hat{P}(c_i)}{\hat{P}(X \leq v)} = \frac{\hat{P}(X \leq v|c_i)\hat{P}(c_i)}{\sum_{j=1}^k \hat{P}(X \leq v|c_j)\hat{P}(c_j)}$$

Thus we have to estimate the prior probability and likelihood for each class in each partition.

Evaluating Split Points: Numeric Attributes

The prior probability for each class in \mathcal{D} can be estimated as

$$\hat{P}(c_i) = \frac{1}{n} \sum_{j=1}^n I(y_j = c_i) = \frac{n_i}{n}$$

where y_j is the class for point x_j , $n = |\mathcal{D}|$ is the total number of points, and n_i is the number of points in \mathcal{D} with class c_i .

Define N_{vi} as the number of points $x_j \leq v$ with class c_i , where x_j is the value of data point x_j for the attribute X , given as

$$N_{vi} = \sum_{j=1}^n I(x_j \leq v \text{ and } y_j = c_i)$$

Evaluating Split Points: Numeric Attributes

We can estimate $\hat{P}(X \leq v|c_i)$ and $\hat{P}(X > v|c_i)$ as follows:

$$\hat{P}(X \leq v|c_i) = \frac{N_{vi}}{n_i}$$

$$\hat{P}(X > v|c_i) = 1 - \hat{P}(X \leq v|c_i) = \frac{n_i - N_{vi}}{n_i}$$

Finally, we have

$$\hat{P}(c_i|\mathcal{D}_Y) = \hat{P}(c_i|X \leq v) = \frac{N_{vi}}{\sum_{j=1}^k N_{vj}}$$

$$\hat{P}(c_i|\mathcal{D}_N) = \hat{P}(c_i|X > v) = \frac{n_i - N_{vi}}{\sum_{j=1}^k (n_j - N_{vj})}$$

The total cost of evaluating a numeric attribute is $O(n \log n + nk)$, where k is the number of classes, and n is the number of points.

Algorithm Evaluate-Numeric-Attribute

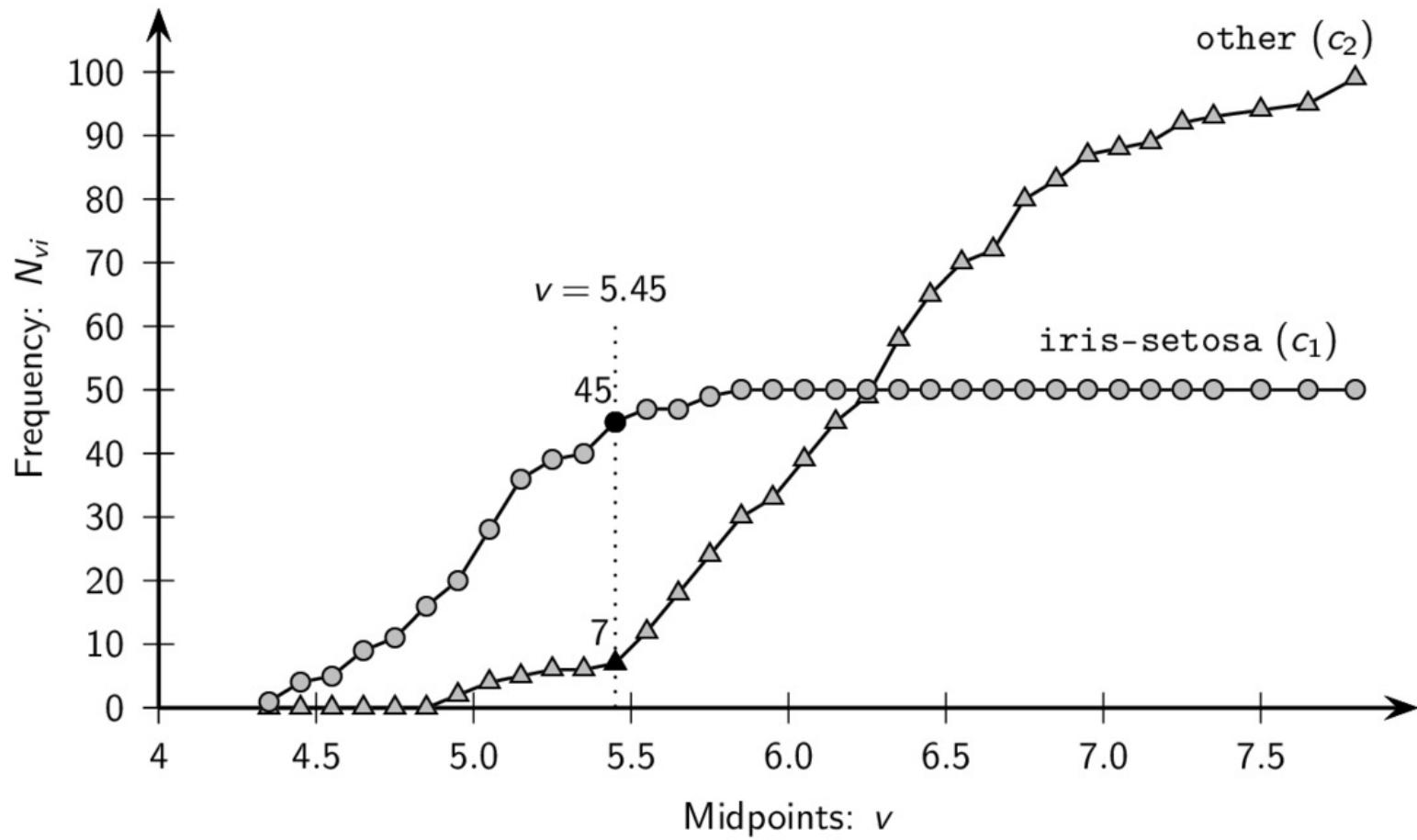
Evaluate-Numeric-Attribute (D, X):

```
1 sort  $D$  on attribute  $X$ , so that  $x_j \leq x_{j+1}, \forall j = 1, \dots, n - 1$ 
2  $\mathcal{M} \leftarrow \emptyset$  // set of midpoints
3 for  $i = 1, \dots, k$  do  $n_i \leftarrow 0$ 
4 for  $j = 1, \dots, n - 1$  do
5   if  $y_j = c_i$  then  $n_i \leftarrow n_i + 1$ 
6   if  $x_{j+1} \neq x_j$  then
7      $v \leftarrow (x_{j+1} + x_j)/2$ ;  $\mathcal{M} \leftarrow \mathcal{M} \cup \{v\}$  // midpoints
8     for  $i = 1, \dots, k$  do
9        $N_{vi} \leftarrow n_i$  // Number of points such that  $x_j \leq v$  and  $y_j = c_i$ 
10  if  $y_n = c_i$  then  $n_i \leftarrow n_i + 1$ 
11   $v^* \leftarrow \emptyset$ ;  $score^* \leftarrow 0$  // initialize best split point
12  forall  $v \in \mathcal{M}$  do
13    for  $i = 1, \dots, k$  do
14       $\hat{P}(c_i | D_Y) \leftarrow \frac{N_{vi}}{\sum_{j=1}^k N_{vj}}$ 
15       $\hat{P}(c_i | D_N) \leftarrow \frac{n_i - N_{vi}}{\sum_{j=1}^k n_j - N_{vj}}$ 
16     $score(X \leq v) \leftarrow Gain(D, D_Y, D_N)$ 
17    if  $score(X \leq v) > score^*$  then
18       $v^* \leftarrow v$ ;  $score^* \leftarrow score(X \leq v)$ 
19 return  $(v^*, score^*)$ 
```

Iris Data: Class-specific Frequencies N_{vi}

Classes c_1 and c_2 for attribute X_1 - sepal length

We first compute the frequencies N_{vi} . Consider the split point $X_1 \leq 5.45$:



Iris Data: Finding the best split point

We do have two classes c_1 and c_2 :

$$\begin{aligned}\hat{P}(c_1) &= 50/150 = 1/3 \\ \hat{P}(c_2) &= 100/150 = 2/3\end{aligned}$$

The entropy of the dataset \mathcal{D} is therefore

$$H(\mathcal{D}) = - \left(\frac{1}{3} \log_2 \frac{1}{3} + \frac{2}{3} \log_2 \frac{2}{3} \right) = 0.918$$

After computing the frequencies N_{vi} , we check each split point, e.g., $X_1 = 5.45$:

$$N_{v1} = 45$$

$$N_{v2} = 7$$

$$\hat{P}(c_1 | \mathcal{D}_Y) = \frac{N_{v1}}{N_{v1} + N_{v2}} = \frac{45}{45 + 7} = 0.865$$

$$\hat{P}(c_2 | \mathcal{D}_Y) = \frac{N_{v2}}{N_{v1} + N_{v2}} = \frac{7}{45 + 7} = 0.135$$

Iris Data: Finding the best split point

$$\hat{P}(c_1 | \mathcal{D}_N) = \frac{n_1 - N_{v1}}{(n_1 - N_{v1}) + (n_2 - N_{v2})} = \frac{50 - 45}{(50 - 45) + (100 - 7)} = 0.051$$

$$\hat{P}(c_2 | \mathcal{D}_N) = \frac{n_2 - N_{v2}}{(n_1 - N_{v1}) + (n_2 - N_{v2})} = \frac{(100 - 7)}{(50 - 45) + (100 - 7)} = 0.949$$

Computing the entropy of the partitions \mathcal{D}_Y and \mathcal{D}_N :

$$H(\mathcal{D}_Y) = -(0.865 \log_2 0.865 + 0.135 \log_2 0.135) = 0.571$$

$$H(\mathcal{D}_N) = -(0.051 \log_2 0.051 + 0.949 \log_2 0.949) = 0.291$$

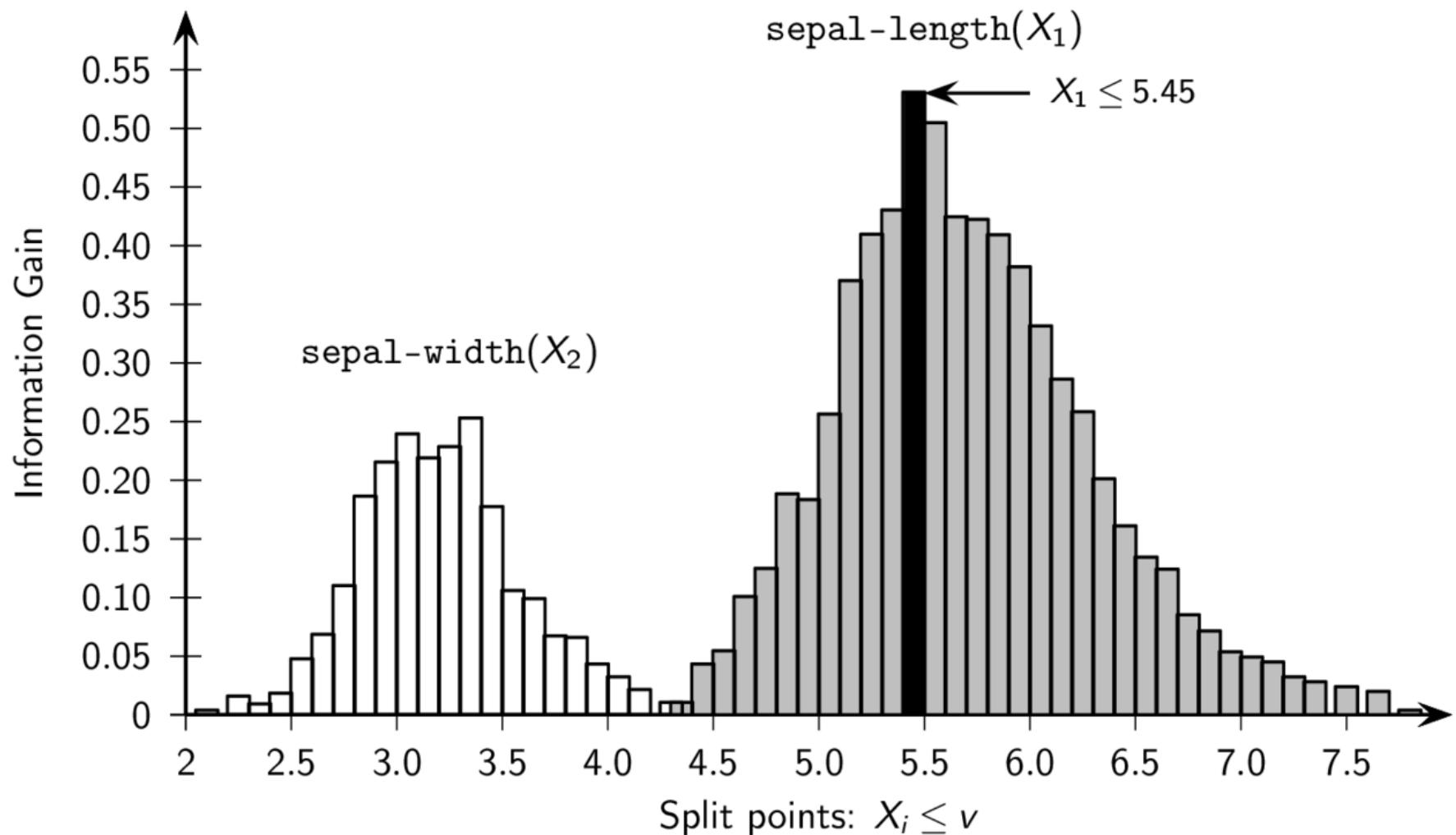
The entropy of $X \leq 5.45$ is:

$$H(\mathcal{D}_Y, \mathcal{D}_N) = \frac{52}{150} H(\mathcal{D}_Y) + \frac{98}{150} H(\mathcal{D}_N) = 0.388$$

The information gain for $X = 5.45$ is therefore

$$Gain = H(\mathcal{D}) - H(\mathcal{D}_Y, \mathcal{D}_N) = 0.918 - 0.388 = 0.53$$

Iris Data: Information Gain for Different Splits



Categorical Attributes

For categorical X the split points are of the form $X \in V$, where $V \subset \text{dom}(X)$ and $V \neq \emptyset$. All distinct partitions of the set of values of X are considered.

If $m = |\text{dom}(X)|$, then there are $O(2^{m-1})$ distinct partitions, which can be too many. One simplification is to restrict V to be of size one, so that there are only m split points of the form $X_j \in \{v\}$, where $v \in \text{dom}(X_j)$.

Define n_{vi} as the number of points $x_j \in \mathcal{D}$, with value $x_j = v$ for attribute X and having class $y_j = c_i$:

$$n_{vi} = \sum_{j=1}^n I(x_j = v \text{ and } y_j = c_i)$$

The class conditional empirical PMF for X is then given as

$$\hat{P}(X = v | c_i) = \frac{\hat{P}(X = v \text{ and } c_i)}{\hat{P}(c_i)} = \frac{n_{vi}}{n_i}$$

We then have

$$\hat{P}(c_i | \mathcal{D}_Y) = \frac{\sum_{v \in V} n_{vi}}{\sum_{j=1}^k \sum_{v \in V} n_{vj}} \quad \hat{P}(c_i | \mathcal{D}_N) = \frac{\sum_{v \notin V} n_{vi}}{\sum_{j=1}^k \sum_{v \notin V} n_{vj}}$$

Algorithm Evaluate-Categorical-Attribute

Evaluate-Categorical-Attribute (D, X, I):

```
1 for  $i = 1, \dots, k$  do
2    $n_i \leftarrow 0$ 
3   forall  $v \in \text{dom}(X)$  do  $n_{vi} \leftarrow 0$ 
4   for  $j = 1, \dots, n$  do
5     if  $x_j = v$  and  $y_j = c_i$  then  $n_{vi} \leftarrow n_{vi} + 1$  // frequency statistics
6   //
// evaluate split points of the form  $X \in V$ 
7  $V^* \leftarrow \emptyset$ ;  $score^* \leftarrow 0$  // initialize best split point
8 forall  $V \subset \text{dom}(X)$ , such that  $1 \leq |V| \leq I$  do
9   for  $i = 1, \dots, k$  do
10     $\hat{P}(c_i | D_Y) \leftarrow \frac{\sum_{v \in V} n_{vi}}{\sum_{j=1}^k \sum_{v \in V} n_{vj}}$ 
11     $\hat{P}(c_i | D_N) \leftarrow \frac{\sum_{v \notin V} n_{vi}}{\sum_{j=1}^k \sum_{v \notin V} n_{vj}}$ 
12     $score(X \in V) \leftarrow Gain(D, D_Y, D_N)$ 
13    if  $score(X \in V) > score^*$  then
14       $V^* \leftarrow V$ ;  $score^* \leftarrow score(X \in V)$ 
15 return  $(V^*, score^*)$ 
```

Discretized sepal length: Class Frequencies

Consider the 2-dimensional Iris dataset comprising the sepal length and sepal width attributes.

sepal length has been discretized as shown in Table, where we can see the class frequencies n_{vi} :

Bins	v : values	Class frequencies (n_{vi})	
		c_1 :iris-setosa	c_2 :other
[4.3,5.2]	Very Short (a_1)	39	6
(5.2,6.1]	Short (a_2)	11	39
(6.1,7.0]	Long (a_3)	0	43
(7.0,7.9]	Very Long (a_4)	0	12

Discretized sepal length: Best Split Point

Consider the split point $X_1 \in \{a_1, a_3\}$:

$$\hat{P}(c_1 | \mathcal{D}_Y) = \frac{n_{a_1 1} + n_{a_3 1}}{(n_{a_1 1} + n_{a_3 1}) + (n_{a_1 2} + n_{a_3 2})} = \frac{39 + 0}{(39 + 0) + (6 + 43)} = 0.443$$

$$\hat{P}(c_2 | \mathcal{D}_Y) = 1 - \hat{P}(c_1 | \mathcal{D}_Y) = 0.557$$

with the entropy given as

$$H(\mathcal{D}_Y) = -(0.443 \log_2 0.443 + 0.557 \log_2 0.557) = 0.991$$

Compute the class PMF for \mathcal{D}_N :

$$\hat{P}(c_1 | \mathcal{D}_N) = \frac{n_{a_2 1} + n_{a_4 1}}{(n_{a_2 1} + n_{a_4 1}) + (n_{a_2 2} + n_{a_4 2})} = \frac{11 + 0}{(11 + 0) + (39 + 12)} = 0.177$$

$$\hat{P}(c_2 | \mathcal{D}_N) = 1 - \hat{P}(c_1 | \mathcal{D}_N) = 0.823$$

with the entropy given as

$$H(\mathcal{D}_N) = -(0.177 \log_2 0.177 + 0.823 \log_2 0.823) = 0.673$$

Discretized sepal length: Best Split Point

$V \in \{a_1, a_3\}$ splits the input data \mathcal{D} into partitions of size

$$|\mathcal{D}_Y| = 39 + 6 + 43 = 88$$

$$|\mathcal{D}_N| = 150 - 88 = 62.$$

The entropy of the split is therefore given as

$$H(\mathcal{D}_Y, \mathcal{D}_N) = \frac{88}{150} H(\mathcal{D}_Y) + \frac{62}{150} H(\mathcal{D}_N) = 0.86$$

Since the entropy of \mathcal{D} is $H(\mathcal{D}) = 0.918$. The gain is given as:

$$Gain = H(\mathcal{D}) - H(\mathcal{D}_Y, \mathcal{D}_N) = 0.918 - 0.86 = 0.058$$

Categorical Split Points for Discretized sepal length

V	Split entropy	Info. gain
$\{a_1\}$	0.509	0.410
$\{a_2\}$	0.897	0.217
$\{a_3\}$	0.711	0.207
$\{a_4\}$	0.869	0.049
$\{a_1, a_2\}$	0.632	0.286
$\{a_1, a_3\}$	0.860	0.058
$\{a_1, a_4\}$	0.667	0.251
$\{a_2, a_3\}$	0.667	0.251
$\{a_2, a_4\}$	0.860	0.058
$\{a_3, a_4\}$	0.632	0.286

Best split: $X \in \{a_1\}$.

Data Mining and Machine Learning: Fundamental Concepts and Algorithms

dataminingbook.info

Mohammed J. Zaki¹ Wagner Meira Jr.²

¹Department of Computer Science
Rensselaer Polytechnic Institute, Troy, NY, USA

²Department of Computer Science
Universidade Federal de Minas Gerais, Belo Horizonte, Brazil

Chapter 19: Decision Tree Classifier