Impurity Measures

M.M. Pedram
pedram@tmu.ac.ir
Kharazmi University
(Fall 2009)

1

Entropy

- Consider the distribution consisting of just two events. Let p be the probability of the first symbol (event). Then, the entropy function is

$$H_q(P) = \sum_{i=1}^{q} p_i \log_r\left(\frac{1}{p_i}\right)$$

$$H_2(P) = p \log_2(1/p) + (1-p)\log_2[1/(1-p)]$$

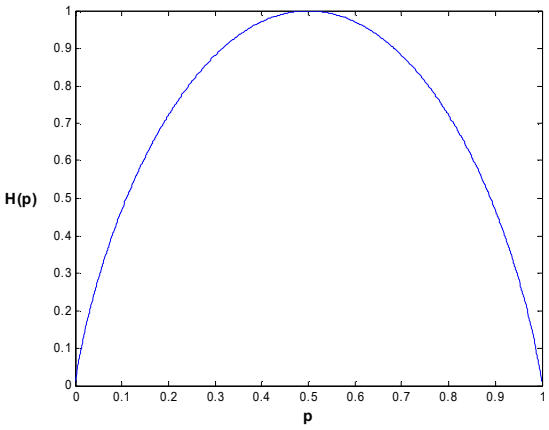
$$\frac{d}{dp}\{p \log_2(1/p) + (1-p) \log_2[1/(1-p)]\} = \log_2(1/p) - \log_2[1/(1-p)]$$

- The maximum of $H_2(P)$ occurs when $p = 1/2$.

2

Entropy

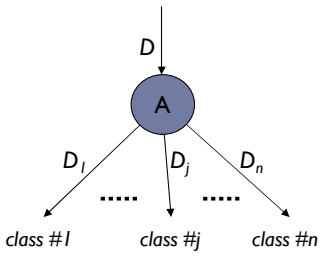
- If the probabilities of all events are equal, then *Entropy* is maximal under all distributions



3

Impurity measure

- *Impurity measure* show how well are the classes in the training dataset separated.
- In general the impurity measure should satisfy:
  - the impurity measure is largest when data are split evenly to classes,
  - the impurity measure should be 0 (smallest) when all data belong to one class.



4

Impurity measure

- p_j : the *relative frequency of class j* in D
  - $|D|$: Total number of data entries
  - $|D_j|$: Number of data entries classified as j
  - p_j : ratio of instances classified as j

$$p_j = \frac{|D_j|}{|D|}$$

- When data are split evenly to classes,

$$p_j = \frac{|D_j|}{|D|} = \frac{1}{no.\ of\ classes}$$

5

Impurity measure

- A measure of impurity is defined as a function of p_j s' and satisfies the conditions stated earlier:

$$IM = \varphi(p_1, p_2, ..., p_j, ...)$$

while $\sum_{j=1}^{n} p_j = 1$

6

Various Impurity Measures

- *Entropy based measure* (Quinlan, C4.5)

$$IM(D) = Entropy(D) = \sum_{j=1}^{n}\left(p_j \cdot \log_2 \frac{1}{p_j} \right)$$
$$= -\sum_{j=1}^{n}\left(p_j \cdot \log_2 p_j \right)$$

- *Gini (diversity) measure* (Breiman, CART)

$$IM(D) = gini(D) = \sum_{j \neq i} p_j \cdot p_i$$
$$= 1 - \sum_{j=1}^{n} p_j^2$$

7

Various Impurity Measures

- Note:

$$\sum_{\substack{i,j=1 \\ j \neq i}}^{n} p_j \cdot p_i = \left(\sum_{j=1}^{n} p_j \right)^2 - \sum_{j=1}^{n} p_j^2 = 1 - \sum_{j=1}^{n} p_j^2$$

- Note: $j \neq i$ was assumed to avoid a trivial *IM*.

- In the two class problem, i.e. binary classification:

$$IM(D) = gini(D) = 1 - \sum_{j=1}^{2} p_j^2 = 2p_1 \cdot p_2$$

8

Gini index (CART, IBM IntelligentMiner)

- If a dataset D contains examples from n classes, gini index, $gini(D)$ is defined as

$$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$$

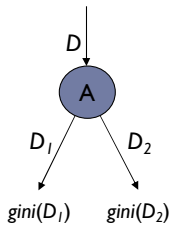
where p_j is the relative frequency of class j in D.

9

Gini index (CART, IBM IntelligentMiner)

- If a dataset D is split on A into two subsets D_1 and D_2, the *gini* index $gini(D)$ is defined as

$$gini_A(D) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$



- Reduction in Impurity:

$$\Delta IM(A,D) = \Delta gini(A) = gini(D) - gini_A(D)$$

  - The attribute provides the smallest $gini_{split}(D)$ (or the largest reduction in impurity) is selected as the splitting attribute(*need to enumerate all the possible splitting points for each attribute*). This attribute and either its splitting subset (for a discrete-valued splitting attribute) or split-point (for a continuous valued splitting attribute) together form the splitting criterion.

10

Gini index interpretations

- Suppose an item is selected randomly and assigned to class *i* with probability p_i, *then Gini* index shows the estimated probability of misclassification in the assignment:

$$gini(D) = \sum_{j \neq i} p_j \cdot p_i$$

11

Gini index example

Class-labeled training tuples from the *AllElectronics* customer database.

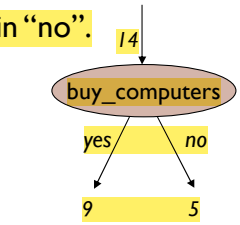
RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

12

Gini index example

- D has 9 tuples in buys_computer = "yes" and 5 in "no".

$$gini(D) = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2 = 0.459$$



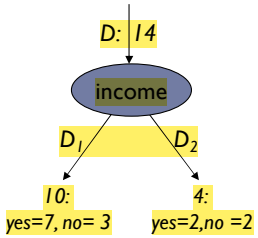
- Suppose A= income:
  - the attribute income partitions D into 10 tuples in partition or class D_1: {low, medium} and the remaining 4 tuples in D_2.

$$gini_{income \in \{low, medium\}}(D) = \left(\frac{10}{14}\right) \cdot gini(D_1) + \left(\frac{4}{14}\right) \cdot gini(D_2)$$

$$gini(D_1) = 1 - \left(\frac{7}{10}\right)^2 - \left(\frac{3}{10}\right)^2, \quad gini(D_2) = 1 - \left(\frac{2}{4}\right)^2 - \left(\frac{2}{4}\right)^2$$

$$gini_{income \in \{low, medium\}}(D) = 0.442 = gini_{income \in \{high\}}$$

$$\Delta gini(A) = gini(D) - gini_A(D) = 0.459 - 0.442 = 0.017$$



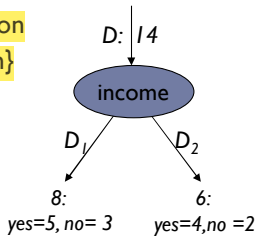
13

Gini index example

- the attribute income partitions D into 8 tuples in partition D_1: {low, high} and the remaining 6 tuples in D_2: {medium}

$$gini_{income \in \{low, high\}}(D) = 0.458 = gini_{income \in \{medium\}}$$

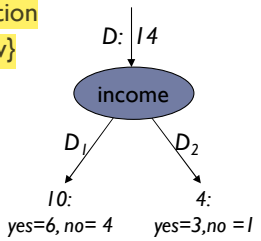
$$\Delta gini(A) = gini(D) - gini_A(D) = 0.459 - 0.458 = 0.001$$



- the attribute income partitions D into 10 tuples in partition D_1: {medium, high} and the remaining 4 tuples in D_2: {low}

$$gini_{income \in \{medium, high\}}(D) = 0.450 = gini_{income \in \{low\}}$$

$$\Delta gini(A) = gini(D) - gini_A(D) = 0.459 - 0.450 = 0.009$$



the best binary split for attribute income is: {medium, high} (or {low})

14

Gini index example

Homework: determine the tree for the example.

Gini index (criterion) property

- Gini index considered as a function $\varphi(p_1, p_2, \ldots, p_j, \ldots)$ is a quadratic function of p_k s' with nonnegative coefficient. Hence, it is a concave function, i.e., for any two points $(p_1, p_2, \ldots, p_j, \ldots)$ and $(p_1', p_2', \ldots, p_j', \ldots)$, and for any nonnegative λ and μ that belong to [0,1] and satisfy $\lambda + \mu = 1$, we have:

$$\varphi(\lambda p_1 + \mu p_1', \lambda p_2 + \mu p_2', \ldots, \lambda p_j + \mu p_j', \ldots) \geq$$
$$\lambda\varphi(p_1, p_2, \ldots, p_j, \ldots) + \mu\varphi(p_1', p_2', \ldots, p_j', \ldots)$$

Actually, Gini index is strictly concave, so that $\Delta gini = 0$ only and if only $p_1 = p_2 = \ldots = p_j = \ldots$.

Comparing Attribute Selection Measures

- The three measures, in general, return good results but
  1. Information gain:
     - biased towards multi-valued attributes.
  2. Gain ratio:
     - adjusts for the above bias,
     - tends to prefer unbalanced splits in which one partition is much smaller than the others.
  3. Gini index:
     - biased to multi-valued attributes,
     - has difficulty when the number of classes is large,
     - tends to favor tests that result in equal-sized partitions and purity in both partitions.

Twoing Criterion

- The twoing criterion is not a true impurity measure!
- This criterion is useful for *multiclass binary tree* creation.

- The overall goal is to find the split that best splits groups of the C categories, i.e., a candidate *super-category* or *super-class* C_1 consisting of all patterns in some subset of the categories, and candidate super-category C_2 as all remaining patterns.
  - Denote the class of categories by C :

    $$C = \{\text{class \#1}, \text{class \#2}, \dots, \text{class \#j}, \dots\}$$

    at each node split the classes into two super-categories(super-classes) :

    $$C_1 = \{\text{class \#k}_1, \text{class \#k}_2, \dots, \text{class \#k}_{n1}\}, \qquad C_2 = C - C_1$$

    For every candidate split s, we compute a change in n impurity $\Delta i(s, C_i)$ as though it corresponded to a standard two-class problem, i.e., we find the split $s^*(C_i)$ that maximizes the change in impurity.

Twoing Criterion

- Both entropy impurity and *gini* impurity can be used with twoing criterion.
- In practice, the followings are more important than the impurity function itself in determining final classifier accuracy:
  - *stopping criterion*: when to stop splitting nodes,
  - *pruning method*: how to merge leaf nodes,

19

Multi-way split

- The matter of allowing the branching ratio at each node to be set during training, *gain ratio impurity*, discussed in C4.5, is used.

20

Note

- The optimization of *info_gain* and $\Delta gini$ is local, i.e., done at a single node. As with most of such greedy methods, there is no guarantee that successive locally optimal decisions lead to the global optimum. There is no guarantee that after training we have the smallest tree.