

Random Forest

MUSA AL-HAWAMDAH / 128129001011

15-10-2012

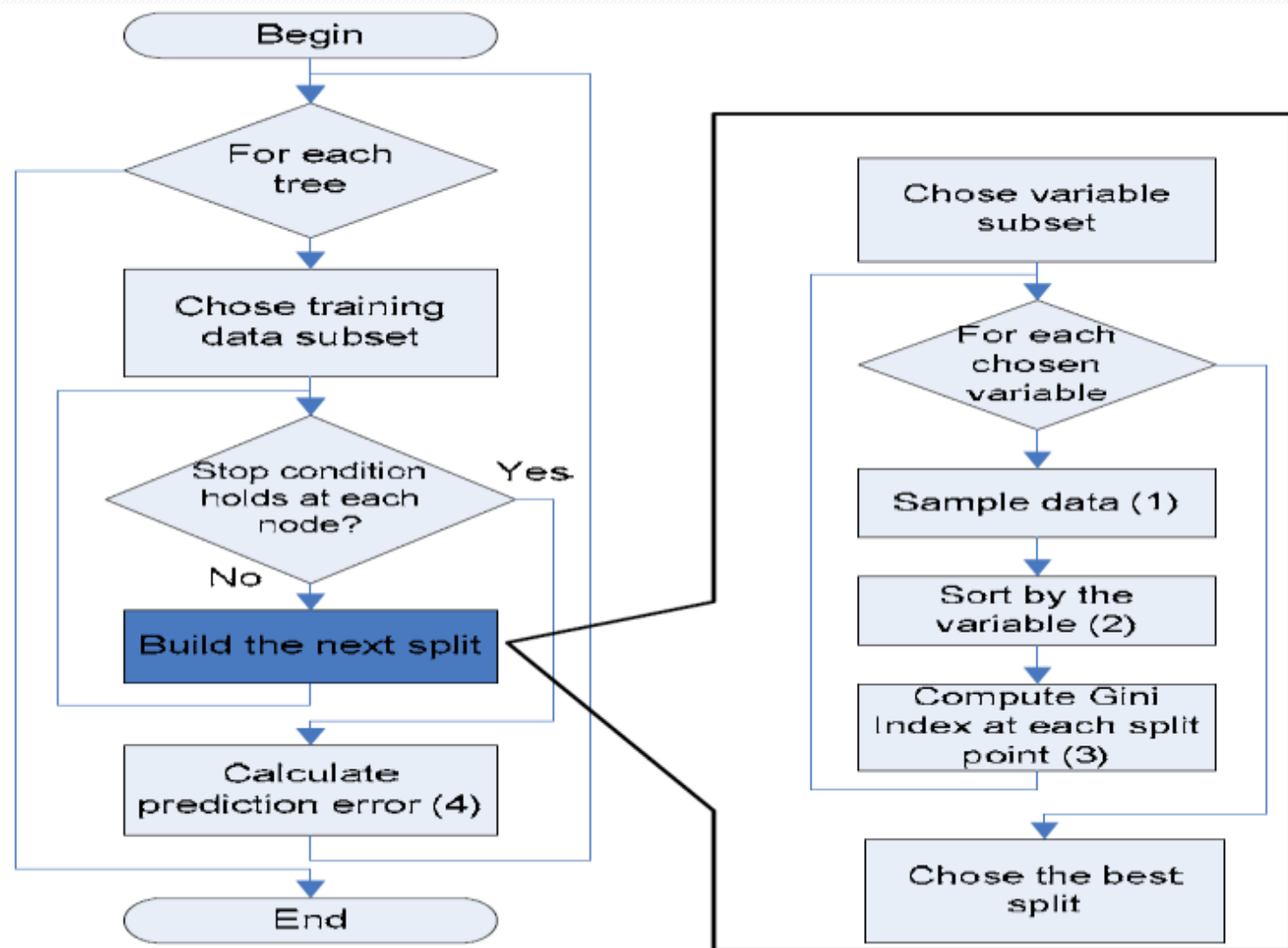
What is random forests

- **An ensemble classifier using many decision tree models.**
- **Can be used for classification or Regression.**
- **Accuracy and variable importance information is provided with the results.**

The Algorithm

- Let the number of training cases be N , and the number of variables in the classifier be M .
- The number m of input variables are used to determine the decision at a node of the tree; m should be much less than M .
- Choose a training set for this tree by choosing N times with replacement from all N available training cases. Use the rest of the cases to estimate the error of the tree, by predicting their classes.
- For each node of the tree, randomly choose m variables on which to base the decision at that node. Calculate the best split based on these m variables in the training set.
- Each tree is fully grown and not pruned.

Random forest algorithm(flow chart):



Advantages

- It produces a highly accurate classifier and learning is fast
- It runs efficiently on large data bases.
- It can handle thousands of input variables without variable deletion.
- It computes proximities between pairs of cases that can be used in clustering, locating outliers or (by scaling) give interesting views of the data.
- It offers an experimental method for detecting variable interactions.

- A Random Forest is a classifier consisting of a collection of tree-structured classifiers $\{h(x, \Theta_k), k = 1, \dots\}$ where the Θ_k are independently, identically distributed random trees and each tree casts a unit vote for the final classification of input x . Like **CART**, Random Forest uses the **gini index** for determining the final class in each tree.
- The final class of each tree is aggregated and voted by weighted values to construct the final classifier.

Gini Index

- Random Forest uses the gini index taken from the CART learning system to construct decision trees. **The gini index** of node impurity is the measure most commonly chosen for classification-type problems. If a dataset T contains examples from n classes,

gini index, **Gini(T)** is defined as:

$$Gini(T) = 1 - \sum_{j=1}^n (p_j)^2$$

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


- If a dataset T is split into two subsets T_1 and T_2 with sizes N_1 and N_2 respectively, the **gini index** of the split data contains examples from n classes, the gini index (T) is defined as:

$$Gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$

****The attribute value that provides the smallest SPLIT Gini (T) is chosen to split the node.**

Operation of Random Forest

- The working of random forest algorithm is as follows.
 1. A random seed is chosen which pulls out at random a collection of samples from the training dataset while maintaining the class distribution.
 2. With this selected data set, a random set of attributes from the original data set is chosen based on user defined values. All the input variables are not considered because of enormous computation and high chances of over fitting.

- 
3. In a dataset where M is the total number of input attributes in the dataset, only R attributes are chosen at random for each tree where $R < M$.
 4. The attributes from this set creates the best possible split using the gini index to develop a decision tree model. The process repeats for each of the branches until the termination condition stating that leaves are the nodes that are too small to split.

Example:

- The example below shows the construction of a single tree using the abridged dataset .
- Only two of the original four attributes are chosen for this tree construction.

RECORD	ATTRIBUTES		CLASS
	HOME_TYPE	SALARY	
1	31	3	1
2	30	1	0
3	6	2	0
4	15	4	1
5	10	4	0

- Assume that the first attribute to be split is HOME_TYPE attribute.
- The possible splits for HOME_TYPE attribute in the left node range from $6 \leq x < 31$, where x is the split value.
- All the other values at each split form the right child node. The possible splits for the HOME_TYPE attributes in the dataset are HOME_TYPE ≤ 6 , HOME_TYPE ≤ 10 , HOME_TYPE ≤ 15 , HOME_TYPE ≤ 30 , and HOME_TYPE ≤ 31 .
- Taking the first split, the gini index is calculated as follows:

- Partitions after the Binary Split on HOME_TYPE ≤ 6 by the Random Forest

Attribute	Number of records		
	Zero(0)	One (1)	N = 5
HOME_TYPE ≤ 6	1	0	n1 = 1
HOME_TYPE > 6	2	2	n2 = 4

- Then $Gini(D_1)$, $Gini(D_2)$, and $Gini_{SPLIT}$ are calculated as follows:

Then $Gini(D_1)$, $Gini(D_2)$, and $Gini_{SPLIT}$ are calculated as follows:

$$Gini(HOME_TYPE \leq 6) = 1 - (1^2 + 0^2) = 0$$

$$Gini(HOME_TYPE > 6) = 1 - \left(\left(\frac{2}{4} \right)^2 + \left(\frac{2}{4} \right)^2 \right) = 0.5$$

$$Gini_{SPLIT} = \left(\frac{1}{5} \right) \times 0 + \left(\frac{4}{5} \right) \times 0.5 = 0.4$$

- In the next step, the data set at HOME_TYPE ≤ 10 is split and tabulated in Table.
- Partitions after the Binary Split on HOME_TYPE ≤ 10 by the Random Forest:

Attribute	Number of records		
	Zero(0)	One (1)	N = 5
HOME_TYPE ≤ 10	2	0	n1 = 2
HOME_TYPE > 10	1	2	n2 = 3

- Then Gini (D1) , Gini (D2) , and Gini_{SPLIT} are calculated as follows:

$$Gini(HOME_TYPE \leq 10) = 1 - (1^2 + 0^2) = 0$$

$$Gini(HOME_TYPE > 10) = 1 - \left(\left(\frac{1}{3} \right)^2 + \left(\frac{2}{3} \right)^2 \right) = 0.4452$$

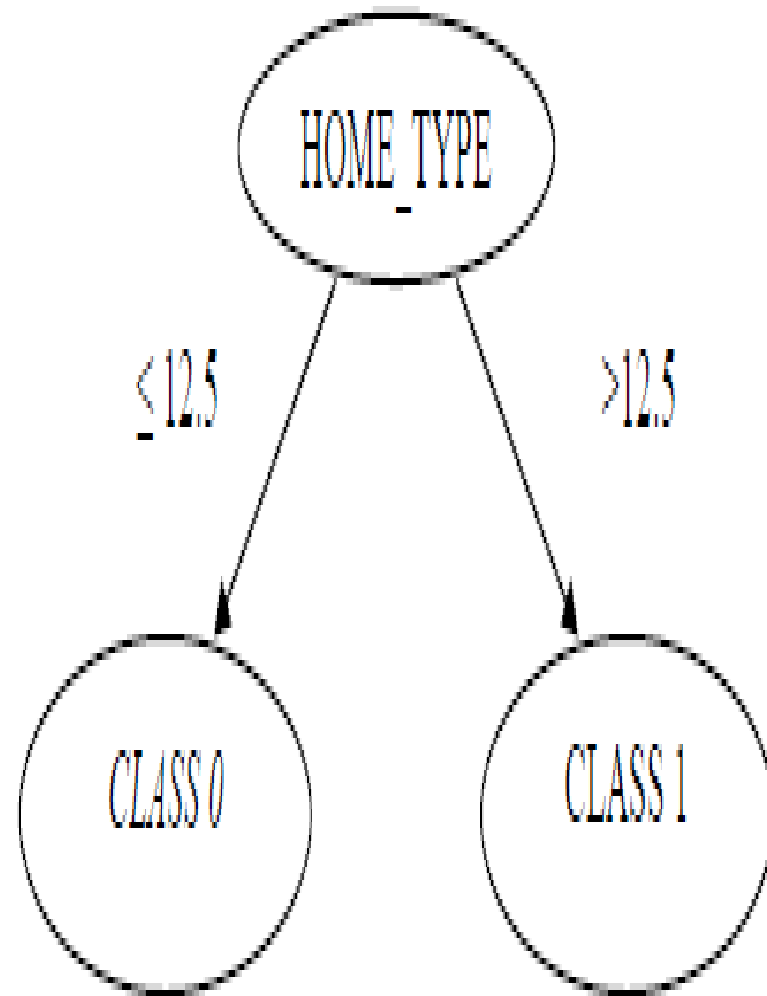
$$Gini_{SPLIT} = \left(\frac{2}{5} \right) \times 0 + \left(\frac{3}{5} \right) \times 0.4452 = 0.2671$$


- tabulates the gini index value for the HOME_TYPE attribute at all possible splits.

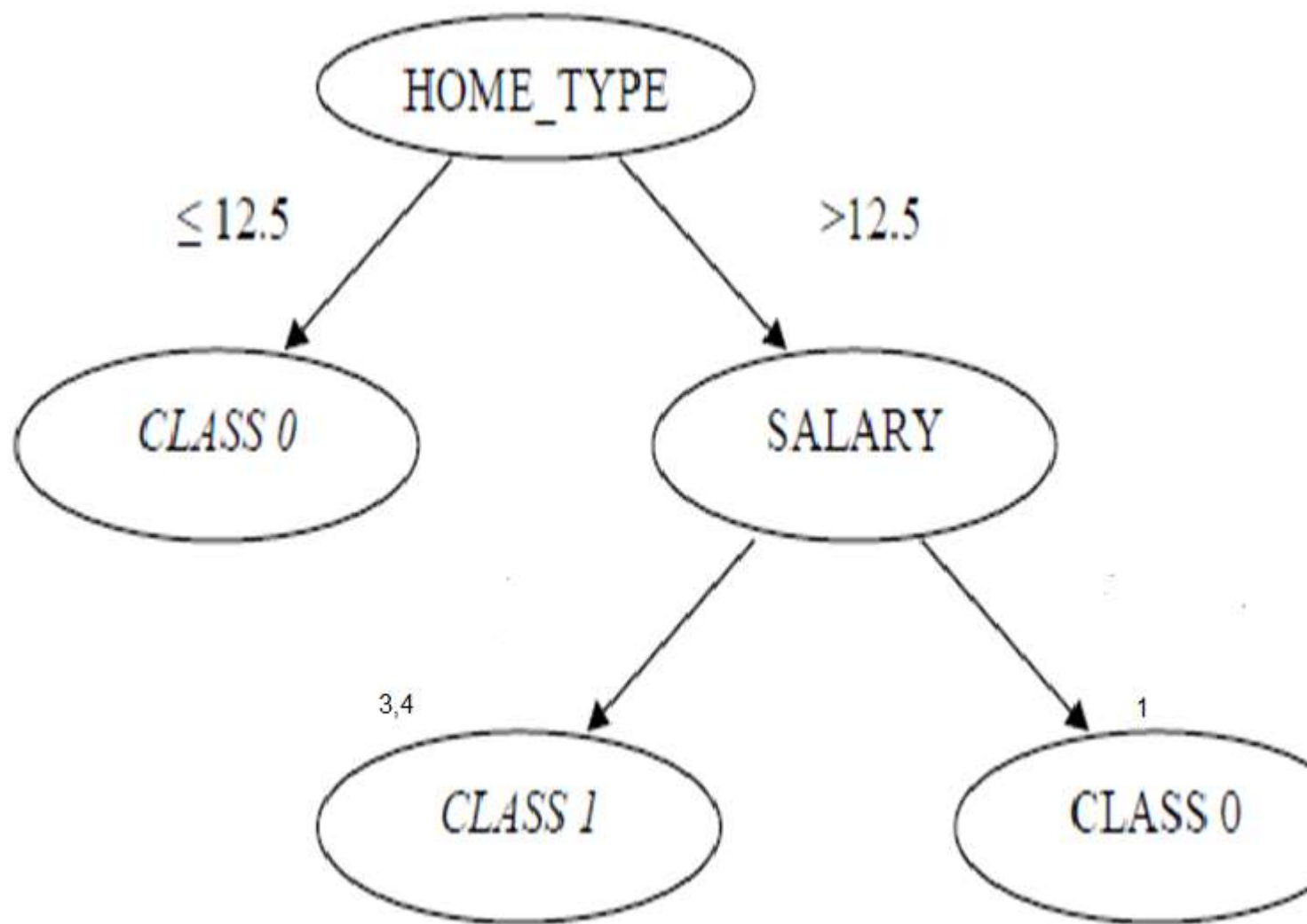
Gini SPILT	Value
Gini _{SPILT} (HOME_TYPE ≤ 6)	0.4000
Gini_{SPILT}(HOME_TYPE ≤ 10)	0.2671
Gini _{SPILT} (HOME_TYPE ≤ 15)	0.4671
Gini _{SPILT} (HOME_TYPE ≤ 30)	0.3000
Gini _{SPILT} (HOME_TYPE ≤ 31)	0.4800

- the split HOME_TYPE ≤ 10 has the lowest value

- In Random Forest, the split at which the gini index is lowest is chosen at the split value.
- However, since the values of the HOME_TYPE attribute are continuous in nature, the midpoint of every pair of consecutive values is chosen as the best split point.
- The best split in our example, therefore, is at $\text{HOME_TYPE} = (10+15)/2=12.5$ instead of at $\text{HOME_TYPE} \leq 10$. The decision tree after the first split is shown in:



- 
- This procedure is repeated for the remaining attributes in the dataset.
 - In this example, the gini index values of the second attribute SALARY are calculated.
 - The lowest value of the gini index is chosen as the best split for the attribute.
 - The final decision trees shown in:



- The decision rules for the decision tree illustrated are:
- If HOME_TYPE ≤ 12.5 , then
Class value is 0.
- If HOME_TYPE is > 12.5 and SALARY is $3/4$, then
Class value is 1.
- If HOME_TYPE is > 12.5 and SALARY is 1, then Class
value is 0.

- This is a single tree construction using the CART algorithm. Random forest follows this same methodology and constructs multiple trees for the forest using different sets of attributes. Random forest has uses a part of the training set to calculate the model error rate by an inbuilt error estimate, the out-of-bag (OOB) error estimate.



THANK YOU