Decision Making through Random Forest

Random Forest

Random forest is a decision tree based non-linear machine learning model for classification, regression and feature selection.

Random Forest

- The word “Random” is for random selection of data instances, which is known as **bootstrapping** method in statistics and in ML as well.
- The word “Forest” is for using several decision trees in developing decision models through **bagging** method.

Random Forest

Steps in Random Forest Classification Method:

- 1. Bootstrapping for random data subset generation
- 2. Decision tree construction for each of the data subset
  - i) Determination of GINI impurity of each of the features.
  - Ii) Determination of GINI impurity of prospective splitting sub-tree
  - Iii) Construction of Decision tree based on the splitting GINI impurity (i.e. if sum of the GINI impurity of splitted sub-tree is lower than the GINI impurity of parent node then split the parent node)
- 3. Bagging for ensemble classification
- 4. Majority voting for classification decision making.

Implement Random forest on the given dataset

Day	Outlook	Temparature	Humidity	Wind	Play Tennis
Day1	Sunny	Hot	High	Weak	No
Day2	Sunny	Hot	High	Strong	No
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	No
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	No
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	No

Bootstrapped Dataset 1

Day	Outlook	Temparature	Humidity	Wind	Play Tennis
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	No
Day2	Sunny	Hot	High	Strong	No

Create decision trees using random subset of variables or columns [Here, we considered only 2 columns randomly]

Day	Temparature	Humidity	Play Tennis
Day10	Mild	Normal	Yes
Day11	Mild	Normal	Yes
Day12	Mild	High	Yes
Day13	Hot	Normal	Yes
Day14	Mild	High	No
Day2	Hot	High	No

Calculations

Temperature

Mild [Yes: 3, No: 1]

Hot [Yes: 1, No: 1]

GINI(Temperature=Mild)

=1-(3/4)^2-(1/4)^2= 1-0.5625-0.0625 = 0.375

GINI(Temperature = Hot)

= 1-(1/2)^2-(1/2)^2 = 0.5

Now, Gini impurity of parent node = weighted average of Gini impurities of leaf nodes.

GINI(Temperature**)** = (4/6)*0.375 + (2/6)*0.5 = 0.417

Humidity

High [Yes: 1, No: 2]

Normal [Yes: 3, No: 0]

GINI(Humidity = High)

= 1 -(1/3)^2-(2/3)^2= 1 - 0.1111 - 0.4444 = 0.444

GINI(Humidity = Normal)

= 1-(3/3)^2-(0/3)^2 = 1-1-0 = 0

GINI(Humidity**)** = (3/6)* 0.444 + (3/6)*0 = 0.22223

Calculations

Now, we should consider for next level nodes for better separation

Day	Outlook	Tempara ture	Humidity	Wind	Play Tennis
Day12	Overcast	Mild	High	Strong	Yes
Day14	Rain	Mild	High	Strong	No
Day2	Sunny	Hot	High	Strong	No

Day	Outlook	Temparat ure	Play Tennis
Day12	Overcast	Mild	Yes
Day14	Rain	Mild	No
Day2	Sunny	Hot	No

Calculations

Temperature

Mild [Yes: 1, No: 1]

Hot [Yes: 0, No: 1]

GINI(Temperature=Mild)=

1-(1/2)^2-(1/2)^2= 0.5

GINI(Temperature = Hot) =

1-(0/1)^2-(1/1)^2 = 1-0-1=0

Now,

Gini impurity of parent node = weighted average of Gini impurities of leaf nodes

GINI(Temperature**)** = (2/3)*0.5 + (1/3)*0 = 0.333

Outlook

Sunny [Yes: 0, No: 1]

Overcast [Yes: 1, No: 0]

Rain [Yes: 0, No: 1]

GINI(Outlook=sunny) = 0

GINI(Outlook= Overcast) = 0

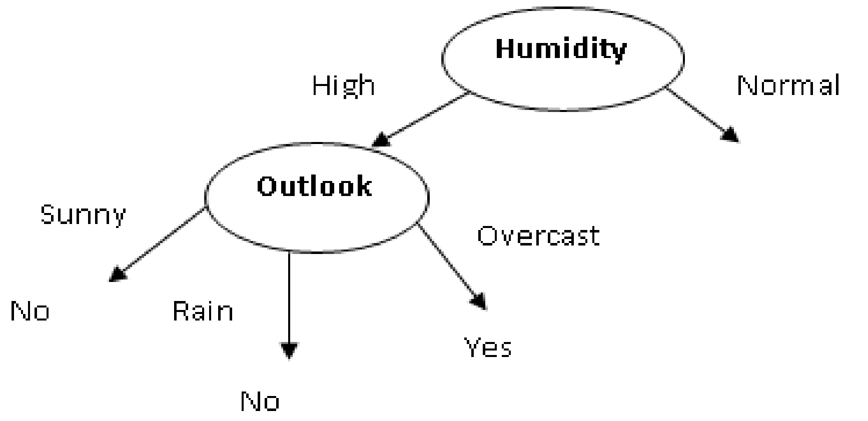
GINI(Outlook= Rain) = 0

Now,

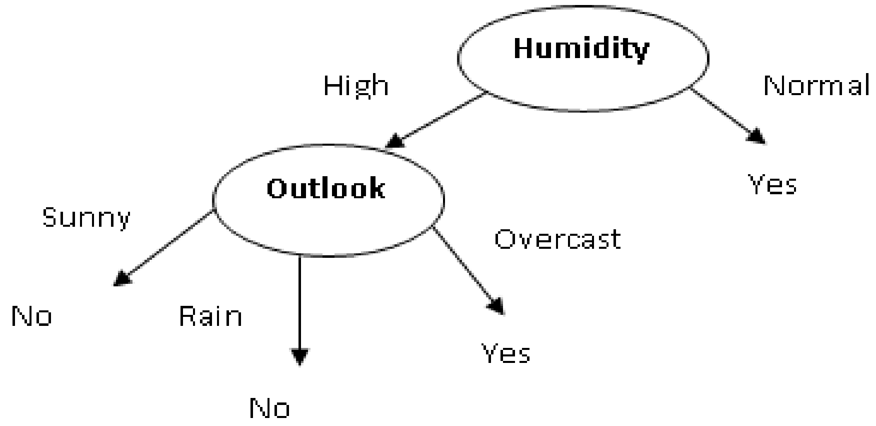
Gini impurity of parent node = weighted average of Gini impurities of leaf nodes

GINI(Outlook) = (1/3)*0 + (1/3)*0 + (1/3)*0 = 0

Calculations



Day	Outlook	Temparature	Humidity	Wind	Play Tennis
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes



Bootstrapped dataset creation-2

Day	Outlook	Temparature	Humidity	Wind	Play Tennis
Day1	Sunny	Hot	High	Weak	No
Day2	Sunny	Hot	High	Strong	No
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day2	Sunny	Hot	High	Strong	No

2. Create decision trees using random subset of variables or columns [Here, we considered only 2 columns randomly] from Bootstrapped dataset

Day	Outlook	Temparature	Play Tennis
Day1	Sunny	Hot	No
Day2	Sunny	Hot	No
Day3	Overcast	Hot	Yes
Day4	Rain	Mild	Yes
Day5	Rain	Cool	Yes
Day2	Sunny	Hot	No

3. Calculations

Outlook

Sunny [Yes: 0, No: 3]

Overcast [Yes: 1, No: 0]

Rain [Yes: 2, No: 0]

GINI(Outlook=sunny) = 1 - (0/3)^2-(3/3)^2 = 1 - 0 - 1 = 0

GINI(Outlook= Overcast) = 1 - (1/1)^2-(0/1)^2 = 1 - 1 - 0 = 0

GINI(Outlook= Rain) = 1 - (2/2)^2-(0/2)^2 = 1 - 1 - 0 = 0

Now,

GINI impurity of parent node = weighted average of Gini impurities of leaf nodes

GINI(Outlook) = (3/6)*0 + (1/6)*0 + (2/6)*0 = 0

3. Calculations (cont…)

Temperature

Hot [Yes: 1, No: 3]

Mild [Yes: 1, No: 0]

Cool [Yes: 1, No: 0]

GINI(Temperature=Hot)= 1-(1/4)^2-(3/4)^2= 1-0.0625-0.5625 = 0.375

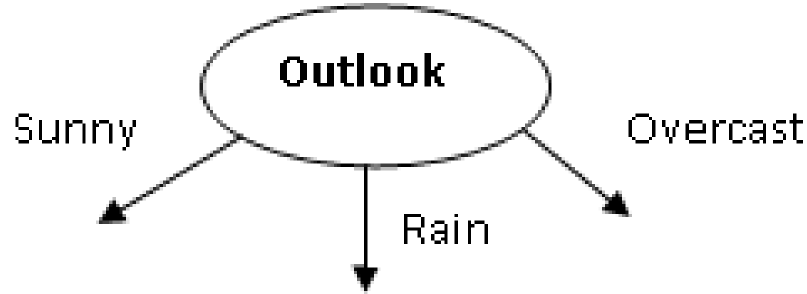
GINI(Temperature=Mild) = 1 - (1/1)^2-(0/1)^2 = 1 - 1 - 0 = 0

GINI(Temperature=Cool) = 1 - (1/1)^2-(0/1)^2 = 1 - 1 - 0 = 0

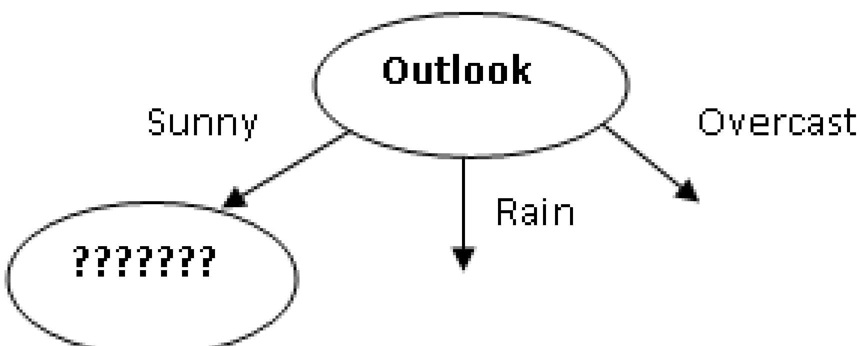
GINI(Temperature) = (4/6)* 0.375 + (1/6)*0 + (1/6)*0 = 0.25

The lowest impurity means, the feature with lowest impurity separates the classes well.

As GINI(Outlook) < GINI(Temperature), so Outlook will be in the root of our decision tree.



Now, we should consider for next level nodes for better separation.



Bootstrapped Dataset 3

Day	Outlook	Temparature	Humidity	Wind	Play Tennis
Day6	Rain	Cool	Normal	Strong	No
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	No
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day13	Overcast	Hot	Normal	Weak	Yes

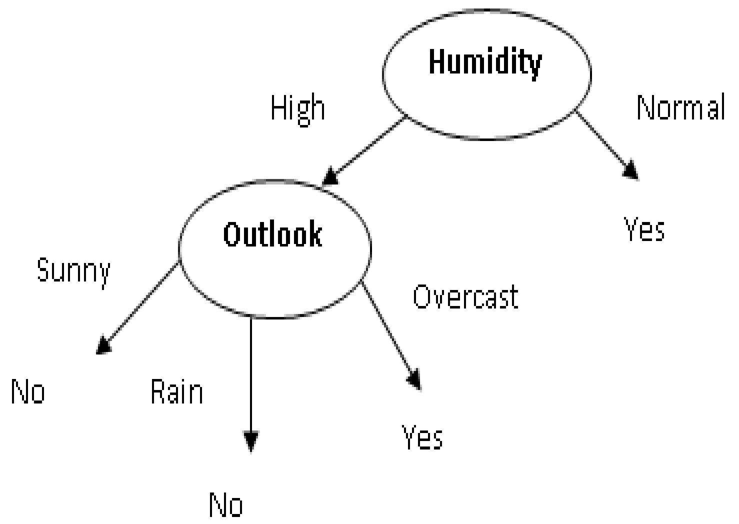
Create decision trees using random subset of variables or columns [Here, we considered only 2 columns randomly]

Day	Humidity	Wind	Play Tennis
Day6	Normal	Strong	No
Day7	Normal	Strong	Yes
Day8	High	Weak	No
Day9	Normal	Weak	Yes
Day10	Normal	Weak	Yes
Day13	Normal	Weak	Yes

NOW, A Query:

Day	Outlook	Temparature	Humidity	Wind	Play Tennis
Day13	Overcast	Hot	Normal	Weak	Yes

Bagging = Yes: 1



Bagging = Yes: 2

If Tree 3 result is NO.
Then Bagging: Yes: 2, No: 1
So, Final result of the query is YES

Calculations

Humidity

High [Yes: 0, No: 1]

Normal [Yes: 4, No: 1]

GINI(Humidity = High) = 1 - (0/1)^2-(1/1)^2= 1 - 1 = 0

GINI(Humidity = Normal) = 1 - (4/5)^2-(1/5)^2 = 1 - 0.64 - 0.04 = 0.32

GINI(Humidity**)** = (5/6)* 0.32 + (1/6)*0 = 0.27

Wind

Strong [Yes: 1, No: 1]

Weak [Yes: 3, No: 1]

GINI(Wind = Strong)=1 - (1/2)^2-(1/2)^2= 1 -0.25- 0.25 = 0.5

GINI(Wind = Weak) = 1 - (3/4)^2 - (1/4)^2 = 0.375

GINI(Wind**)** = (2/6)*0.5+ (4/6)*0.375 = 0.42

As GINI(Wind) > GINI(Humidity), so Humidity will be the the root of our decision tree.