Decision Tree Learning using Gini (jee-nee) impurity

Lets start with data set given below, here target is to predict If the user will buy a computer or not(Yes or No) based various condition such as age, income and credit rating. There are 14 instances in this dataset.

Sr. no.	Age	Income	Student	Credit Rating	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	High	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

Fig. 1 Dataset

Gini = 1 — Σ $(Pi)^2$ for i=1 to number of classes

Gini Index of Age

Taking the first feature

Age	Yes	No	Number of Instances
Youth	2	3	5
Middle-aged	4	0	4
Senior	3	2	5

Fig. 2

Gini(Age=Youth) = 1 — $(2/5)^2$ — $(3/5)^2$ = 1–0.16–0.36 = 0.48

Gini(Age=Middle-aged) = 1 — $(4/4)^2$ — $(0/4)^2$ = 0

Gini(Age=Senior) = 1 — $(3/5)^2$ — $(2/5)^2$ = 1–0.36–0.16 = 0.48

Now, we calculate weighted sum of Gini indexes for Age feature:

Gini(Age) = (5/14) x 0.48 + (4/14) x 0 + (5/14) x 0.48 = 0.171 + 0 + 0.171 = 0.342

Gini Index of Income

Income	Yes	No	Number of Instances
Low	3	1	4
Medium	3	2	5
High	3	2	5

Fig. 3

Gini(Income=Low)= 1 — $(3/5)^2$ — $(1/5)^2$ = 1–0.36–0.04= 0.6

Gini(Income=Medium) = 1 — $(3/5)^2$ — $(2/5)^2$ = 1–0.36–0.16 = 0.48

Gini(Income=High) = 1 — $(3/5)^2$ — $(2/5)^2$ = 1–0.36–0.16 = 0.48

Now, we calculate weighted sum of Gini indexes Income feature:

Gini(Income) = (4/14) x 0.6 + (5/14) x 0.48 + (5/14) x 0.48 = 0.171 + 0.171 + 0.171 = 0.513

Gini Index of Student

Student	Yes	No	Number of Instances
Yes	6	1	7
No	3	4	7

Fig. 4

Gini(Student=Yes) = $1 — (6/7)^2 — (1/7)^2 = 1–0.734–0.020 = 0.246$

Gini(Student=No) = $1 — (3/7)^2 — (4/7)^2 = 1–0.183–0.326 = 0.489$

Now, we calculate weighted sum of Gini indexes Student feature:

Gini(Student)= (7/14) x 0.246 + (7/14) x 0.489 = 0.123 + 0.244= 0.367

Credit Rating

Credit Rating	Yes	No	Number of Instances
Fair	6	2	8
Excellent	3	3	6

Fig. 5

Gini(Credit Rating= Fair) = 1 — $(6/8)^2$ — $(2/8)^2$ = 1–0.562–0.0625 = 0.375

Gini(Credit Rating= Excellent) = 1 — $(3/6)^2$ — $(3/6)^2$ = 1–0.25–0.25 = 0.5

Now, we calculate weighted sum of Gini indexes Credit Rating feature:

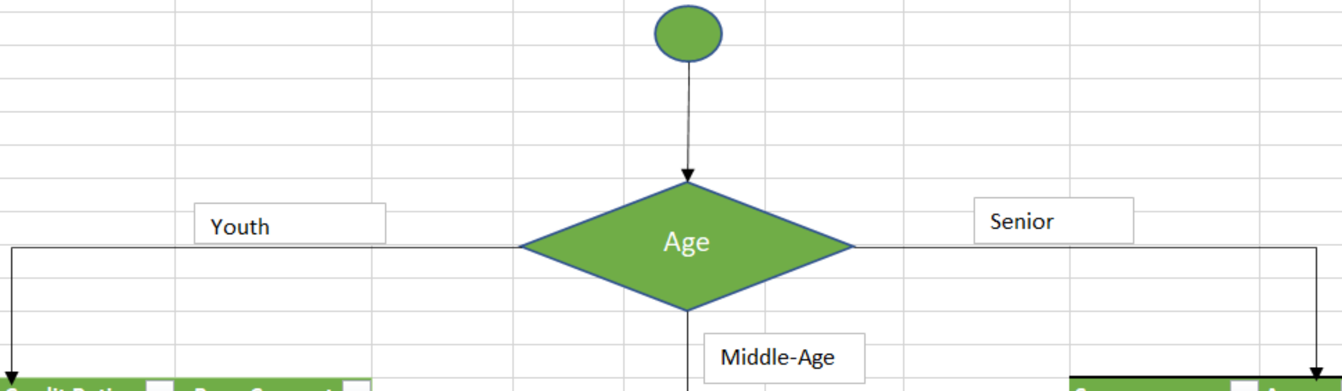
Gini(Credit Rating) = (8/14) x 0.375 + (6/14) x 0.5 = 0.214 + 0.214 = 0.428

So far we've calculated Gini index values for each feature. Further we will choose Age feature as it has the lowest cost.

Feature	Gini Index
Age	0.342
Income	0.513
Student	0.367
Credit Rating	0.428

Fig. 6

As we have got the first node which will be Age:

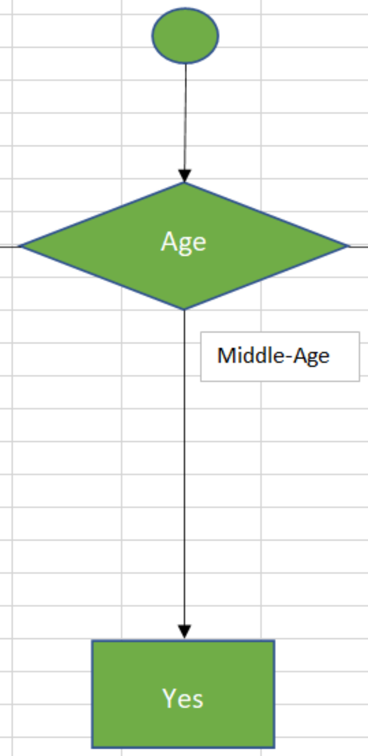


Sr. no	Age	Incom	Studer	Credit Rating	Buys Compute
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes

Sr. no.	Age	Income	Student	Credit Rat	Buys Compu
3	Middle-aged	High	No	Fair	Yes
7	Middle-aged	Low	Yes	Excellent	Yes
12	Middle-aged	Medium	No	Excellent	Yes
13	Middle-aged	High	Yes	Fair	Yes

Sr. no.	Age	Income	Student	Credit R	Buys Compu
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
10	Senior	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

As we can see here the Middle age has all decisions as Yes it will be stopped for Middle-aged in the age feature.



Sr. no	Age	Incom	Studer	Credit Rating	Buys Compute
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes

Sr. no.	Age	Income	Student	Credit R	Buys Compu
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
10	Senior	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

Now taking subset Youth we will calculated Gini index for Income, Student, Credit Rating with respect to Youth.

Sr. no.	Age	Income	Student	Credit Rating	Buys Computer
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes

Fig. 9

Gini of Income for Youth-Age

Income	Yes	No	Number of Instances
Low	1	0	1
Medium	1	1	2
High	0	2	2

Fig. 10

Gini(Age=Youth and Income=Low)= $1 — (1/1)^2 — (0/1)^2 = 0$

Gini(Age=Youth and Income=Medium)= $1 — (1/2)^2 — (1/2)^2 = 0.5$

Gini(Age=Youth and Income=High = $1 — (0/2)^2 — (2/2)^2 = 0$

Now, we calculate weighted sum of Gini indexes for Youth with Income feature:

Gini(Age=Youth and Income) = (2/5) x 0+ (2/5) x 0.5 + (2/5) x 0 = 0.2

Gini of Student for Youth-Age

Student	Yes	No	Number of Instances
Yes	2	0	2
No	0	3	3

Fig. 11

Gini(Age=Youth and Student=Yes) = $1 — (2/2)^2 — (0/2)^2 = 0$

Gini(Age=Youth and Student=No) = $1 — (0/3)^2 — (3/3)^2 = 0$

Now, we calculate weighted sum of Gini indexes for Youth with student feature:

Gini(Age=Youth and Student) = (2/5)x0 + (3/5)x0 = 0

Gini for Credit Rating and Youth-Age

Credit Rating	Yes	No	Number of Instances
Fair	1	2	3
Excellent	1	1	2

Fig. 12

Gini(Age=Youth and Credit Rating=Fair) = 1 — $(1/3)^2$ — $(2/3)^2$ = 0.266

Gini(Age=Youth and Credit Rating=Excellent) = 1 — $(1/2)^2$ — $(1/2)^2$ = 0.2

Now, we calculate weighted sum of Gini indexes for Youth with credit rating feature:

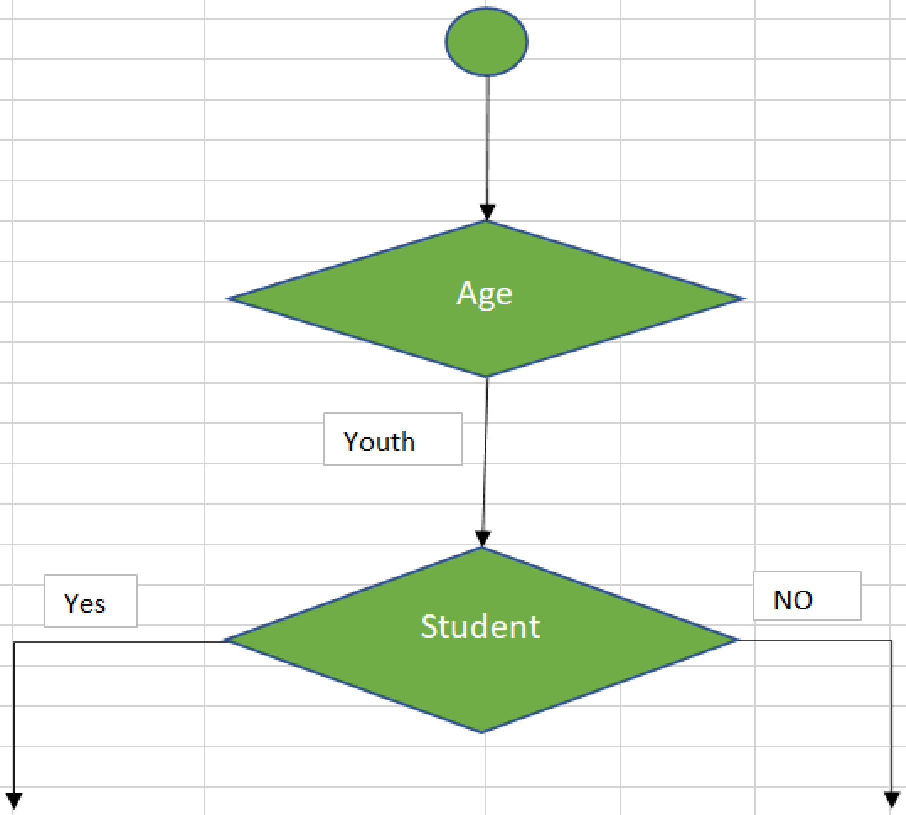
Gini(Age=Youth and Credit Rating) = (3/5)x0.266 + (2/5)x0.2 = 0.466

Decision For Age=Youth

Feature	Gini Index
Income	0.2
Student	0
Credit Rating	0.466

Fig. 13

Here the next feature will be taken as student as it has lowest cost.

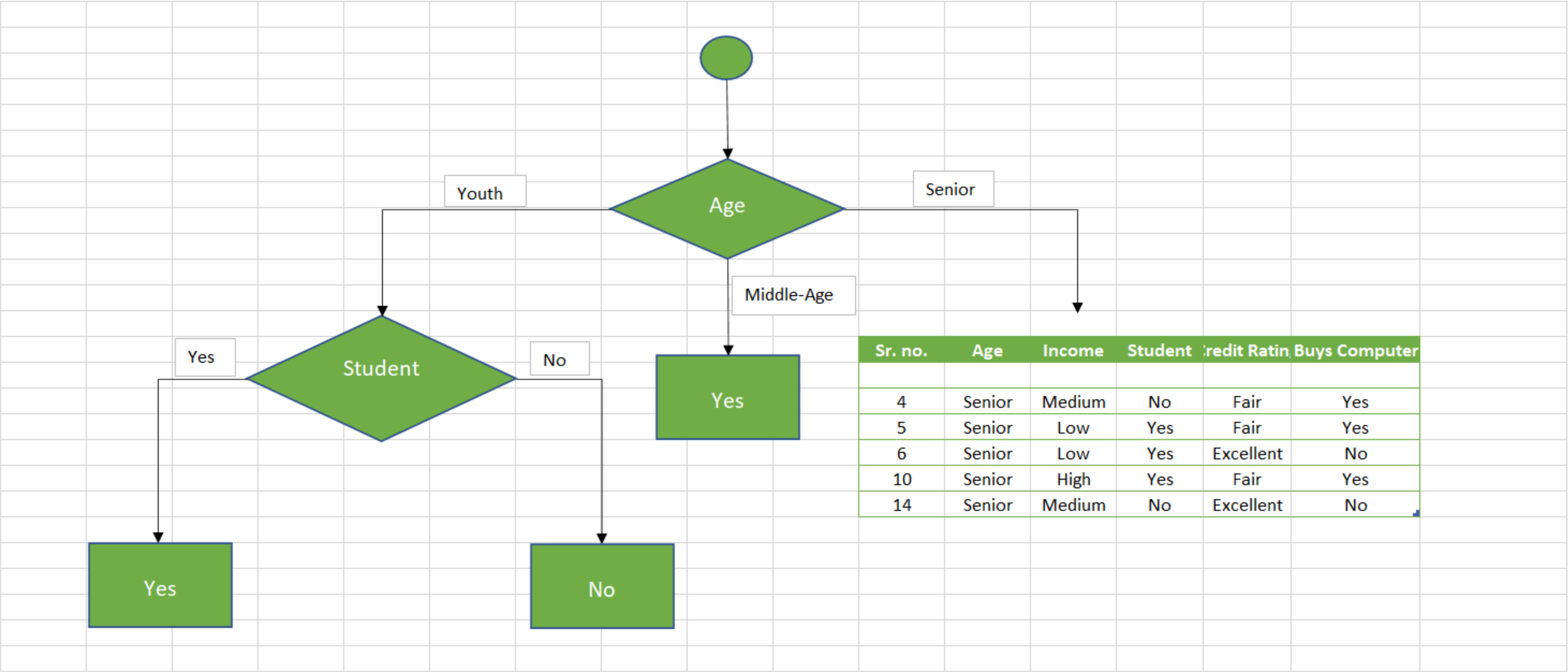


Sr. no.	Age	Income	Student	Credit Rating	Buys Computer
9	Youth	Low	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes

Age	Income	Student	Credit Rating	Buys Computer
Youth	High	No	Fair	No
Youth	High	No	Excellent	No
Youth	Medium	No	Fair	No

As seen, decision is always no for Student, Income and for Age Youth. On the other hand, decision will always be yes for Student, Income and for Age Youth. We can now say this particular branch is complete.

Moving ahead we will now check for Age = Senior with other features such as income, Student and Credit Rating.



Sr. no.	Age	Income	Student	:redit Ratin	Buys Computer
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
10	Senior	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

Gini of Income with Age- Senior

Income	Yes	No	Number of Instances
Low	1	1	2
Medium	1	1	2
High	1	0	1

Fig. 16

Gini(Age=Senior and Income=Low)=1 — $(1/2)^2$ — $(1/2)^2$ = 0.5

Gini(Age=Senior and Income=Medium)=1 — $(1/2)^2$ — $(1/2)^2$ = 0.5

Gini(Age=Senior and Income=High)=1 — $(1/1)^2$ — $(0/2)^2$ = 0

Now, we calculate weighted sum of Gini indexes for Senior with Income feature:

Gini(Age=Senior and Income) = (1/5) x 0.5+ (1/5) x 0.5 + = 0.2

Gini of Student with Age-Senior

Student	Yes	No	Number of Instances
Yes	2	1	3
No	1	1	2

Fig. 17

Gini(Age=Senior and Student=Yes) = $1 — (2/3)^2 — (1/3)^2 = 0.444$

Gini(Age=Senior and Student=No)= $1 — (1/2)^2 — (1/2)^2 = 0.5$

Now, we calculate weighted sum of Gini indexes for Senior with student feature:

Gini(Age=Senior and Student) = (2/5)x0.5 + (3/5)x0.444 = 0.466

Gini of Credit Rating with Age-Senior

Credit Rating	Yes	No	Number of Instances
Fair	3	0	3
Excellent	0	2	2

Fig. 18

Gini(Age=Senior and Credit Rating=Fair) = 1 — $(3/3)^2$ — $(0/3)^2$ = 0

Gini(Age=Senior and Credit Rating=Excellent) = 1 — $(0/2)^2$ — $(2/2)^2$ = 0

Now, we calculate weighted sum of Gini indexes for Senior with Credit rating feature:

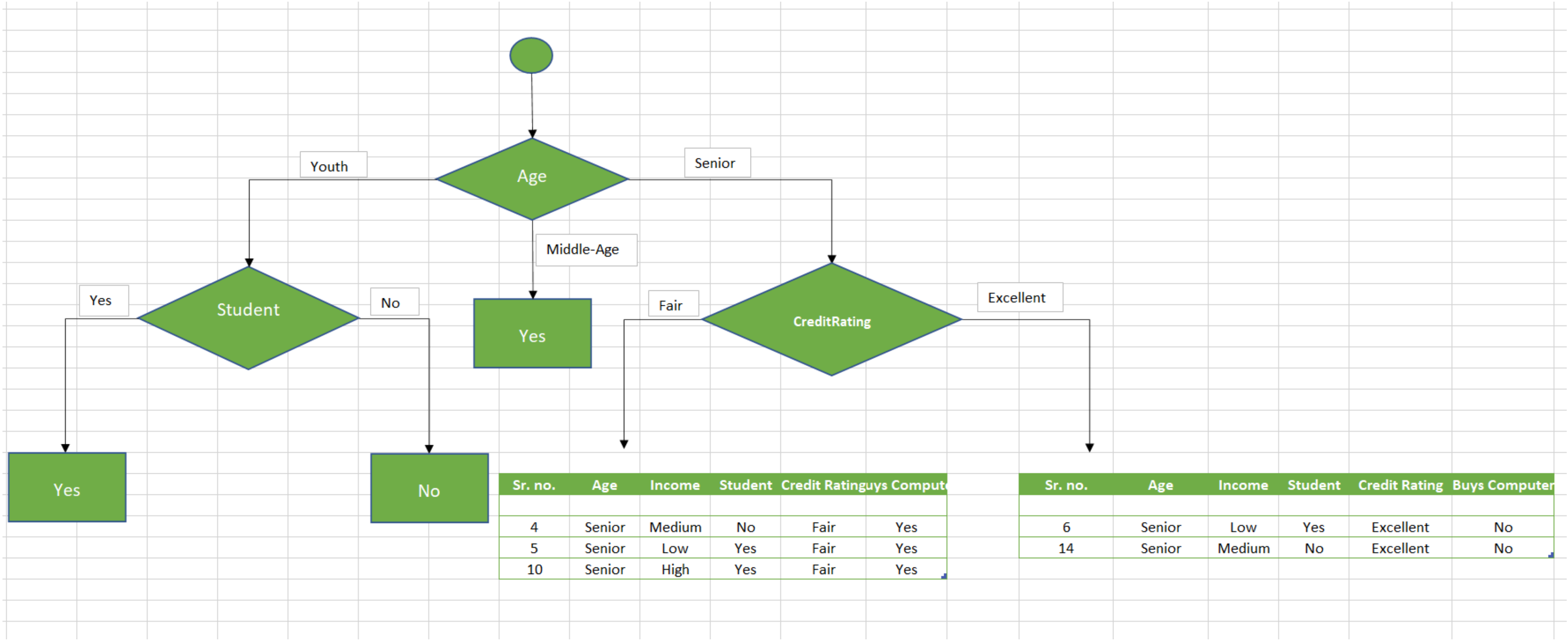
Gini(Age=Senior and Credit Rating) = (3/5)x0 + (2/5)x0 = 0

Decision for Age Senior

Feature	Gini Index
Income	0.2
Student	0.466
Credit Rating	0

Fig. 19

Now we take Credit Rating as next feature and check for same



As seen, decision is always yes when Credit Rating is Fair. On the other hand, decision is always no if Credit Rating is Excellent. This branch ends here.

The final output of the tree is as follows:

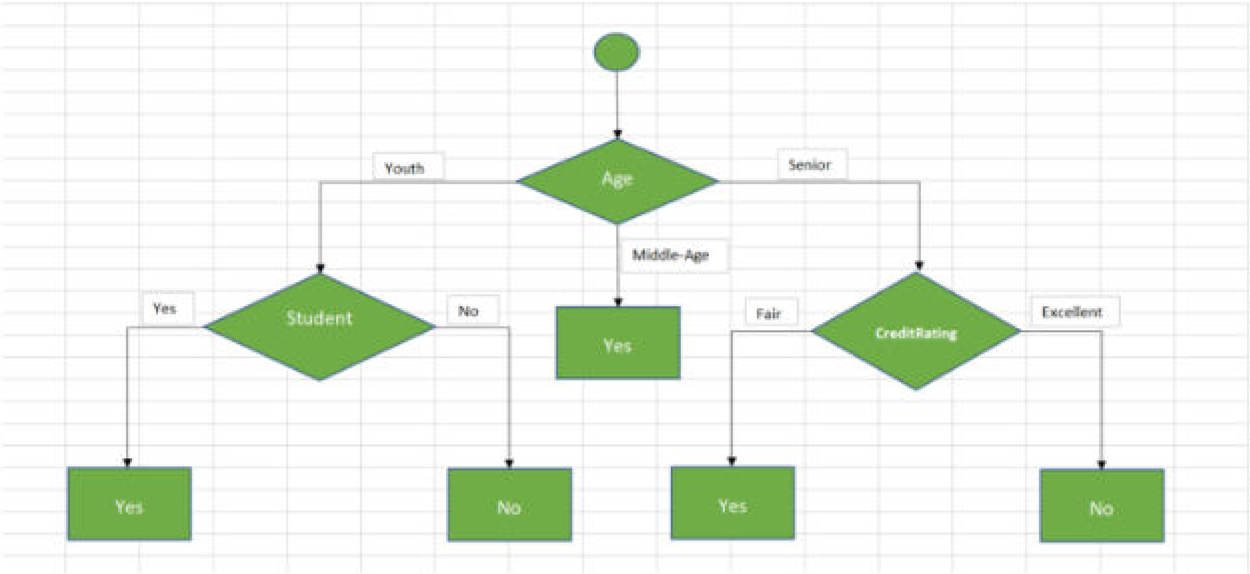


Fig. 20 Final Output of Decision Tree by CART algorithm

As the objective of a decision tree is to make the optimal choice at the end of each node, an algorithm that can do that is required. Hence we can conclude by saying CART is one of the algorithm which helps in doing so.

		Yes	No	Total
Feature 2: **Outlook**	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	3	2	5
	Total	10	4	

Gini (PlayTennis, Outlook=Sunny)
= 1-(⅖)^2 - (⅗)^2 =0.48

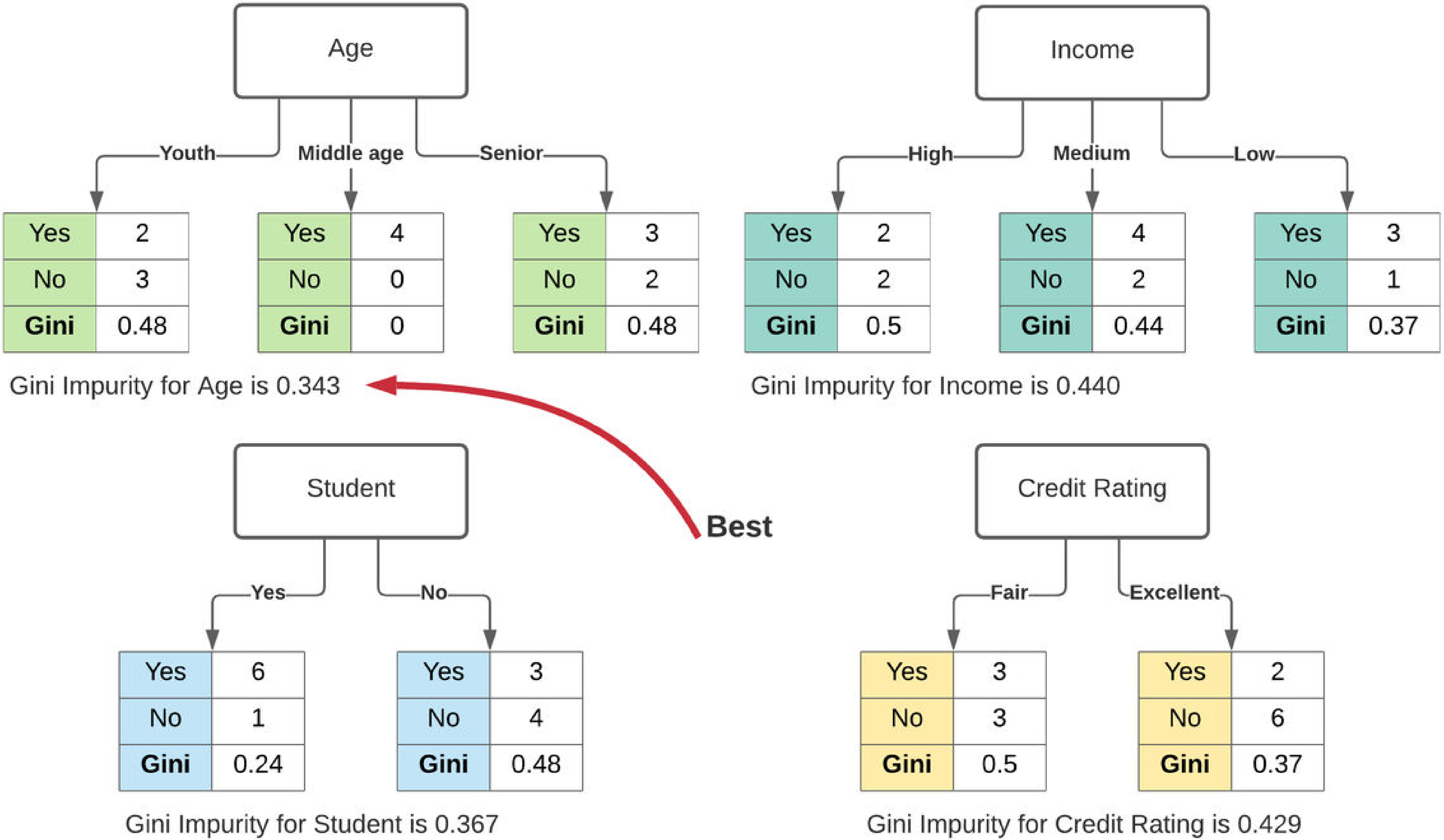
Gini (PlayTennis, Outlook=Overcast)
= 1-(4/4)^2 - (0/4)^2=0

Gini (PlayTennis, Outlook=Rainy)
= 1-(⅗)^2 - (⅖)^2 =0.48

The Gigi Index of Outlook (children node)
= 5/14 x 0.48 + 4/14 x 0 + 5/14 x 0.48 = 0.3429

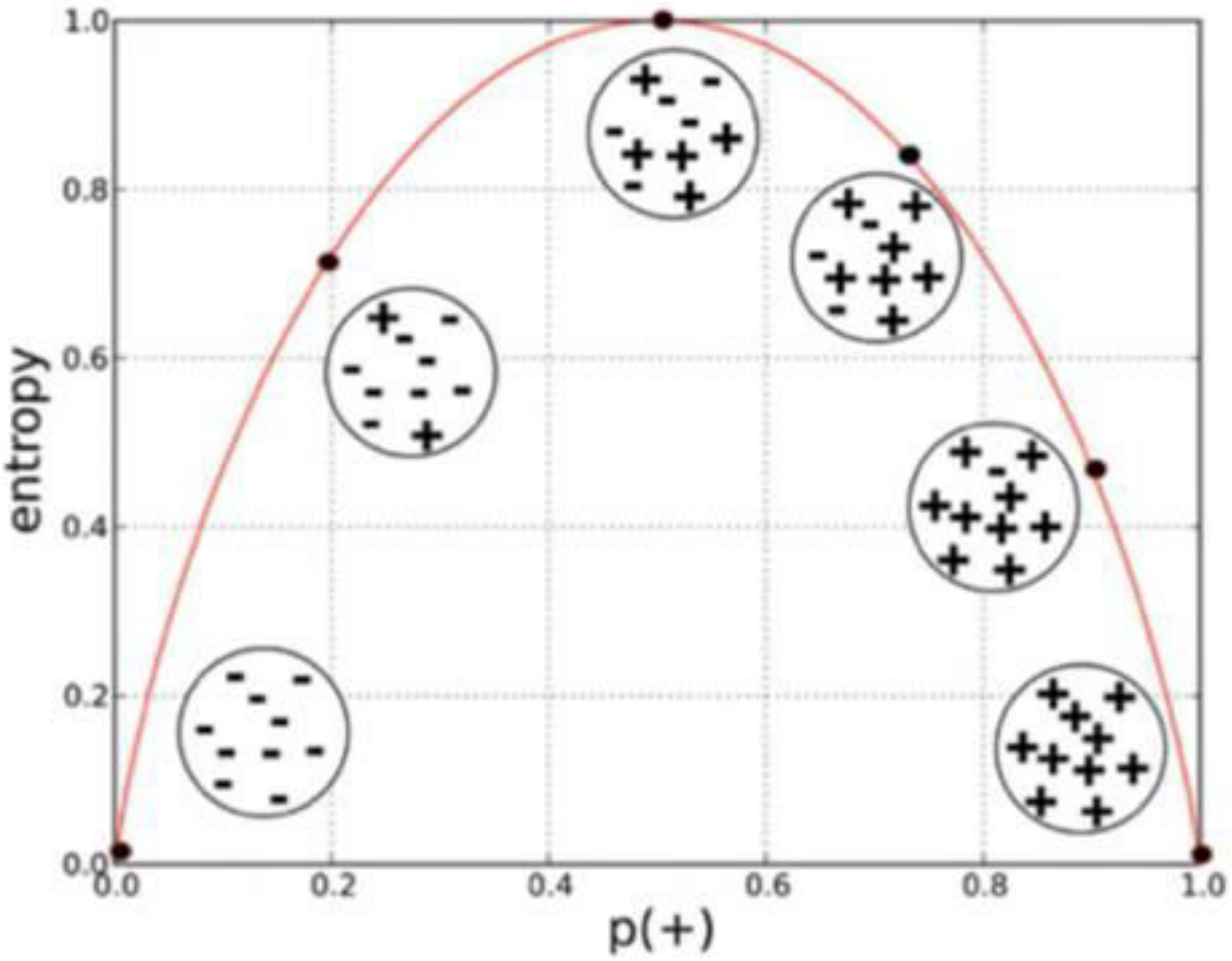
Gini Gain = Gini (parent node) - Gini (children node)

= [1- (10/14)^2 -(4/14)^2] - 0.3429
= 0.4082 - 0.3429
= 0.065

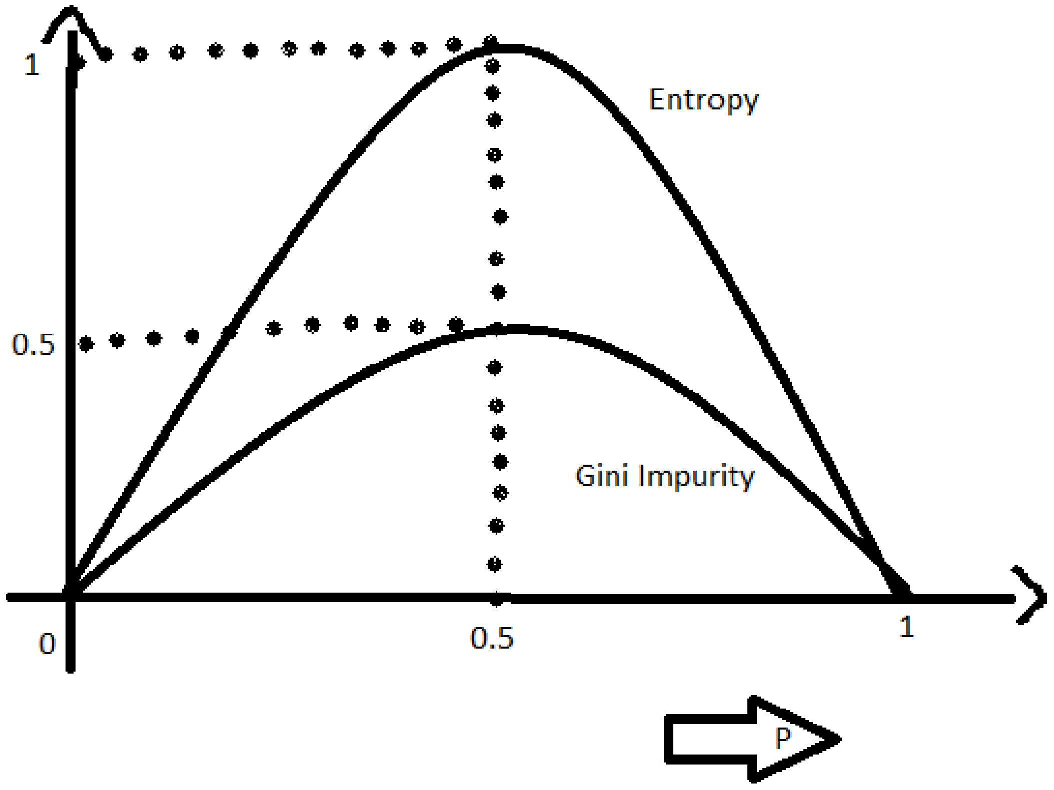


Age
Youth
Middle age
Senior
Yes 2
No 3
Gini 0.48
Yes 4
No 0
Gini 0
Yes 3
No 2
Gini 0.48
Gini Impurity for Age is 0.343
Income
High
Medium
Low
Yes 2
No 2
Gini 0.5
Yes 4
No 2
Gini 0.44
Yes 3
No 1
Gini 0.37
Gini Impurity for Income is 0.440
Best
Student
Yes
No
Yes 6
No 1
Gini 0.24
Yes 3
No 4
Gini 0.48
Gini Impurity for Student is 0.367
Credit Rating
Fair
Excellent
Yes 3
No 3
Gini 0.5
Yes 2
No 6
Gini 0.37
Gini Impurity for Credit Rating is 0.429

Entropy vs Probability



Comparison of Entropy and Gini



$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

$$Gini(E) = 1 - \sum_{j=1}^{c} p_j^2$$

Handling continuous attributes

Instance	a_1	a_2	a_3	Target Class
1	T	T	1.0	+
2	T	T	6.0	+
3	T	F	5.0	−
4	F	F	4.0	+
5	F	T	7.0	−
6	F	T	3.0	−
7	F	F	8.0	−
8	T	F	7.0	+
9	F	T	5.0	−

Step1:
Sort Dataset

a3	Target Class
1.0	+
3.0	-
4.0	+
5.0	-
5.0	-
6.0	+
7.0	-
7.0	+
8.0	-

Step2:
Find the best split point

a3	Target Class	Split Point	Entropy	Gain
1.0	+			
3.0	-			
4.0	+			
5.0	-			
5.0	-			
6.0	+			
7.0	-			
7.0	+			
8.0	-			

Calculating Entropy at 2.0

a3	Target Class	Split Point	Entropy	Gain
1.0	+	2.0	0.8484	0.1427
3.0	-	3.5		
4.0	+	4.5		
5.0	-			
5.0	-	5.5		
6.0	+	6.5		
7.0	-			
7.0	+	7.5		
8.0	-			

- $E = -\frac{4}{9}\log_2\frac{4}{9} - \frac{5}{9}\log_2\frac{5}{9} = 0.9911$
- *Split Point* $= 2.0$
- $E(a_3) = \frac{1}{9}[-\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}] + \frac{8}{9}[-\frac{3}{8}\log_2\frac{3}{8} - \frac{5}{8}\log_2\frac{5}{8}] = 0.8484$
- $Gain(a_3) = 0.9911 - 0.8484 = 0.1427$

Best split is at 2.0

a3	Target Class	Split Point	Entropy	Gain
1.0	+	2.0	0.8484	0.1427
3.0	-	3.5	0.9885	0.0026
4.0	+	4.5	0.9183	0.0728
5.0	-			
5.0	-	5.5	0. 9839	0.0072
6.0	+	6.5	0. 9728	0.0183
7.0	-			
7.0	+	7.5	0. 8889	0.1022
8.0	-			

https://tejaswinishinde1110.medium.com/decision-tree-cart-algorithm-9998290bba17