ID3 Decision Tree Class Activity

CS 450

The Dataset

Consider the following dataset from a bank loan scenario. The target value, or label, or class, that we are trying to predict is the "Should Loan" column.

As a team, follow the process of the ID3 algorithm to induce a decision tree from this training data.

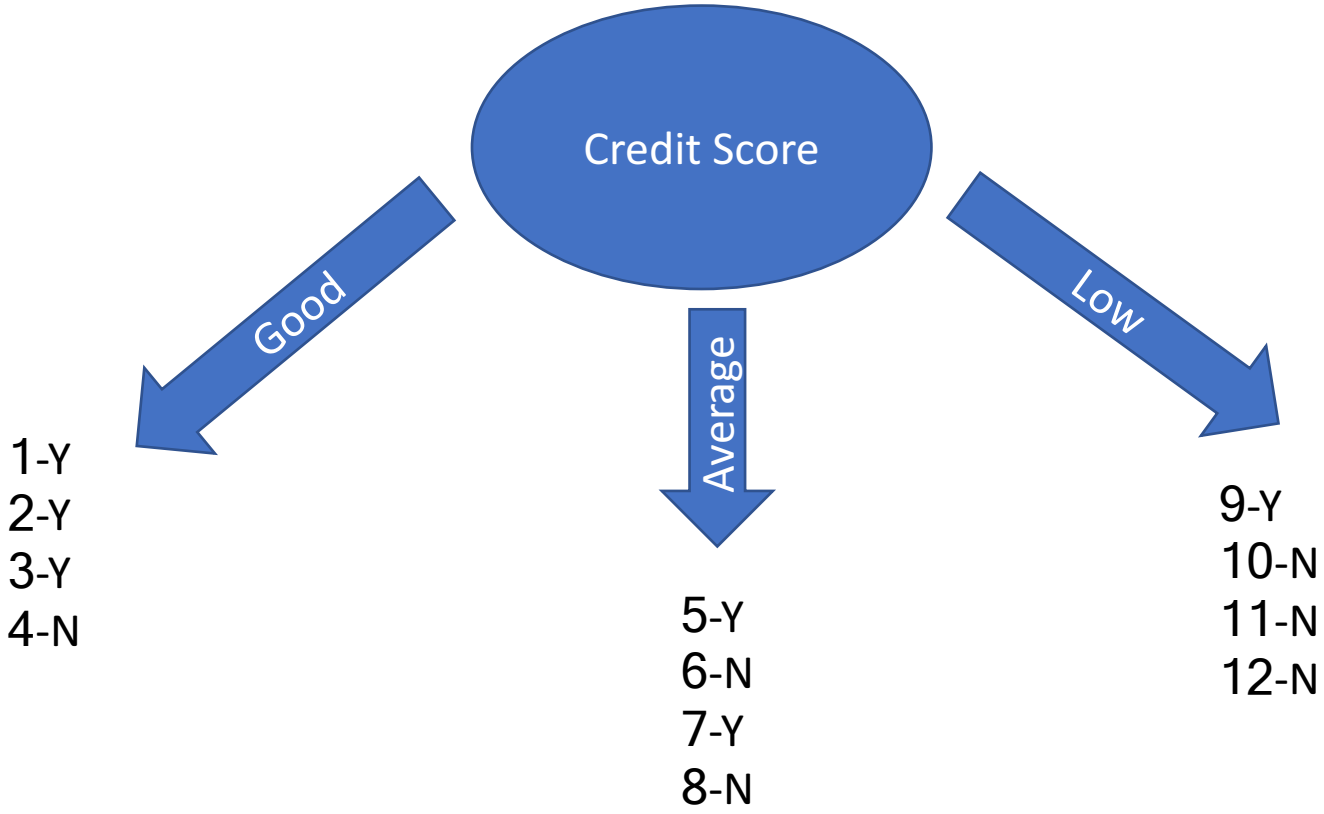
Row #	Credit Score	Income	Collateral	Should Loan
1	Good	High	Good	Yes
2	Good	High	Poor	Yes
3	Good	Low	Good	Yes
4	Good	Low	Poor	No
5	Average	High	Good	Yes
6	Average	Low	Poor	No
7	Average	High	Poor	Yes
8	Average	Low	Good	No
9	Low	High	Good	Yes
10	Low	High	Poor	No
11	Low	Low	Good	No
12	Low	Low	Poor	No

Choosing a Root Node

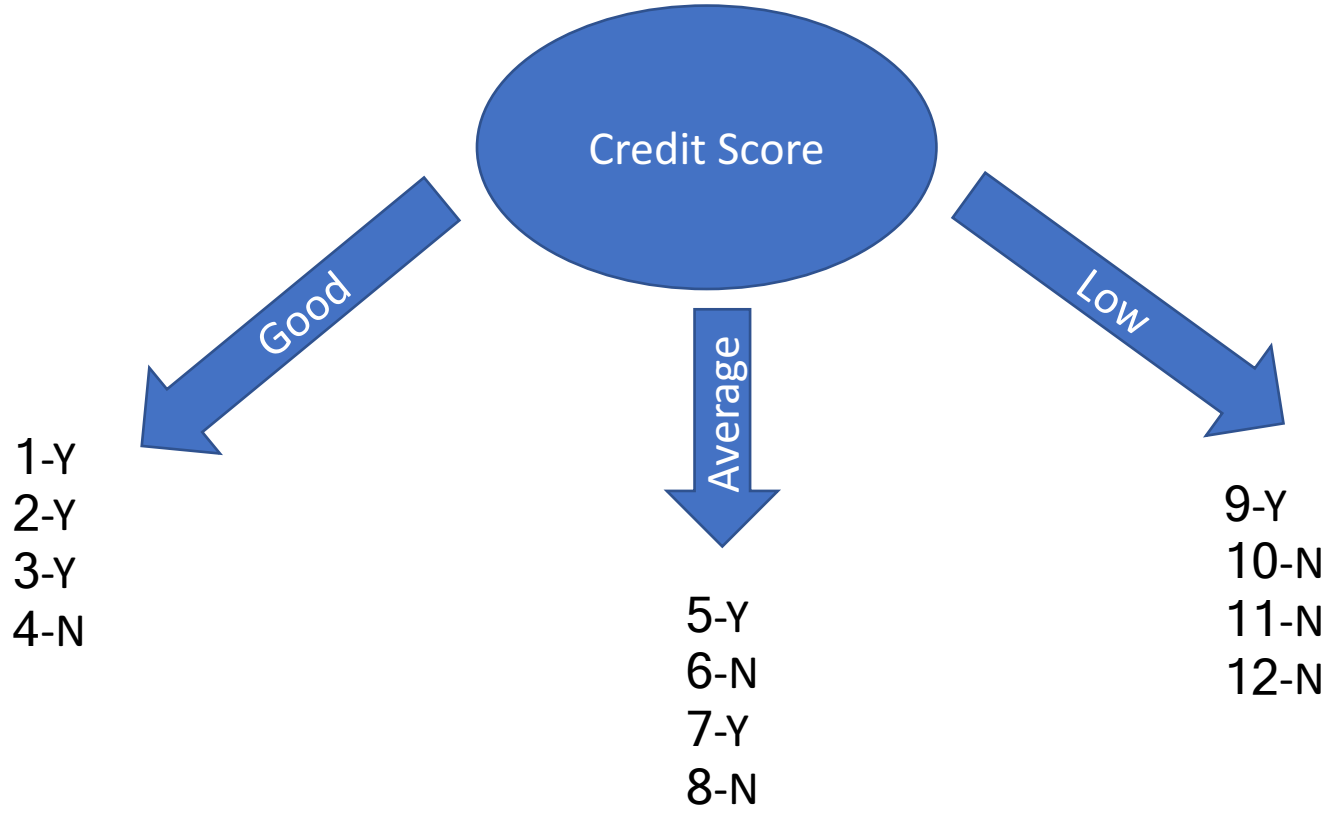
- Step 1 – Determine which attribute should be the root node.
  - This is done by trying each one, and seeing what the resulting split would look like. Calculate the entropy for each split, and then choose the attribute that results in the lowest entropy.

- As a team, work through this problem, then compare your answers with those on the following page.

Option 1 – Credit Score



Option 1 – Credit Score

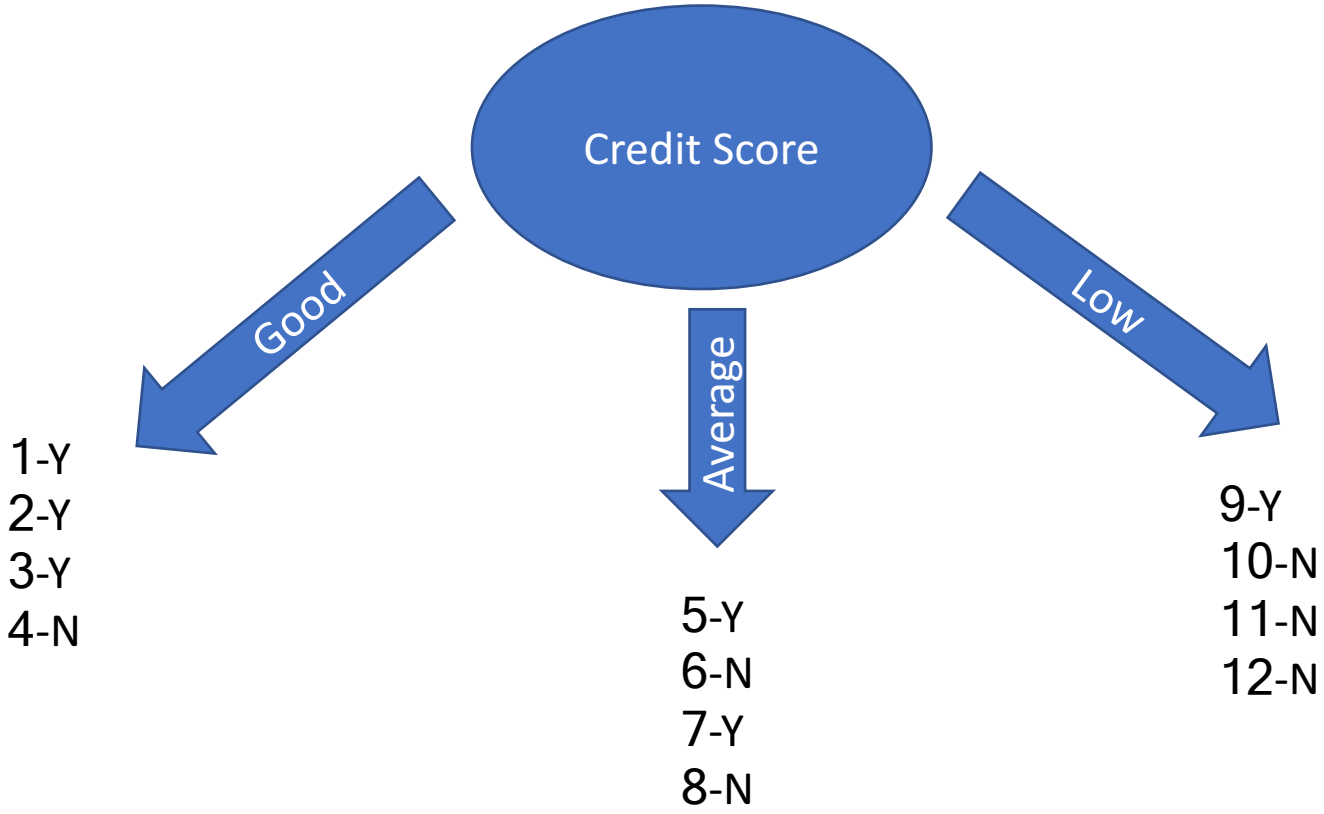


Entropy of “Good” Branch

E_good = -(3/4) * math.log2(3/4) - (1/4) * math.log2(1/4)

E_good = 0.811

Option 1 – Credit Score

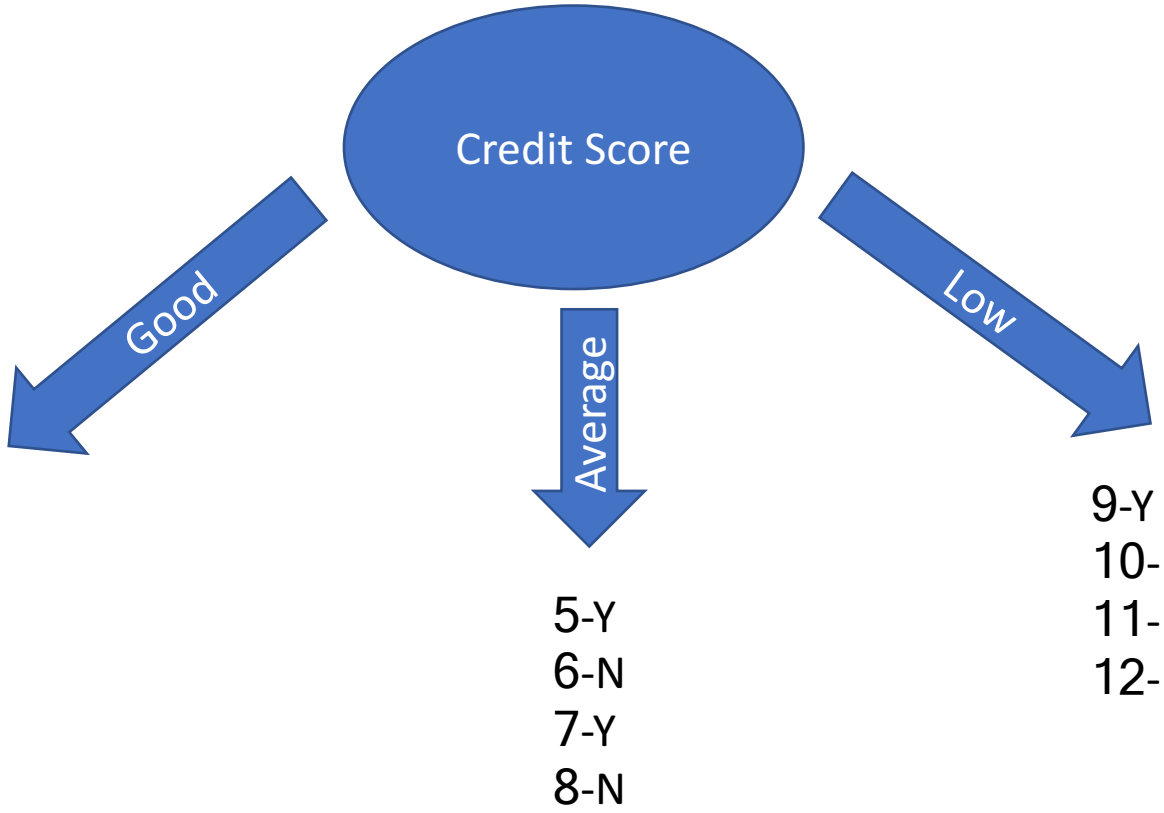


Entropy of “Average” Branch

E_average = -(2/4) * math.log2(2/4) - (2/4) * math.log2(2/4)

E_average = 1.0

Option 1 – Credit Score

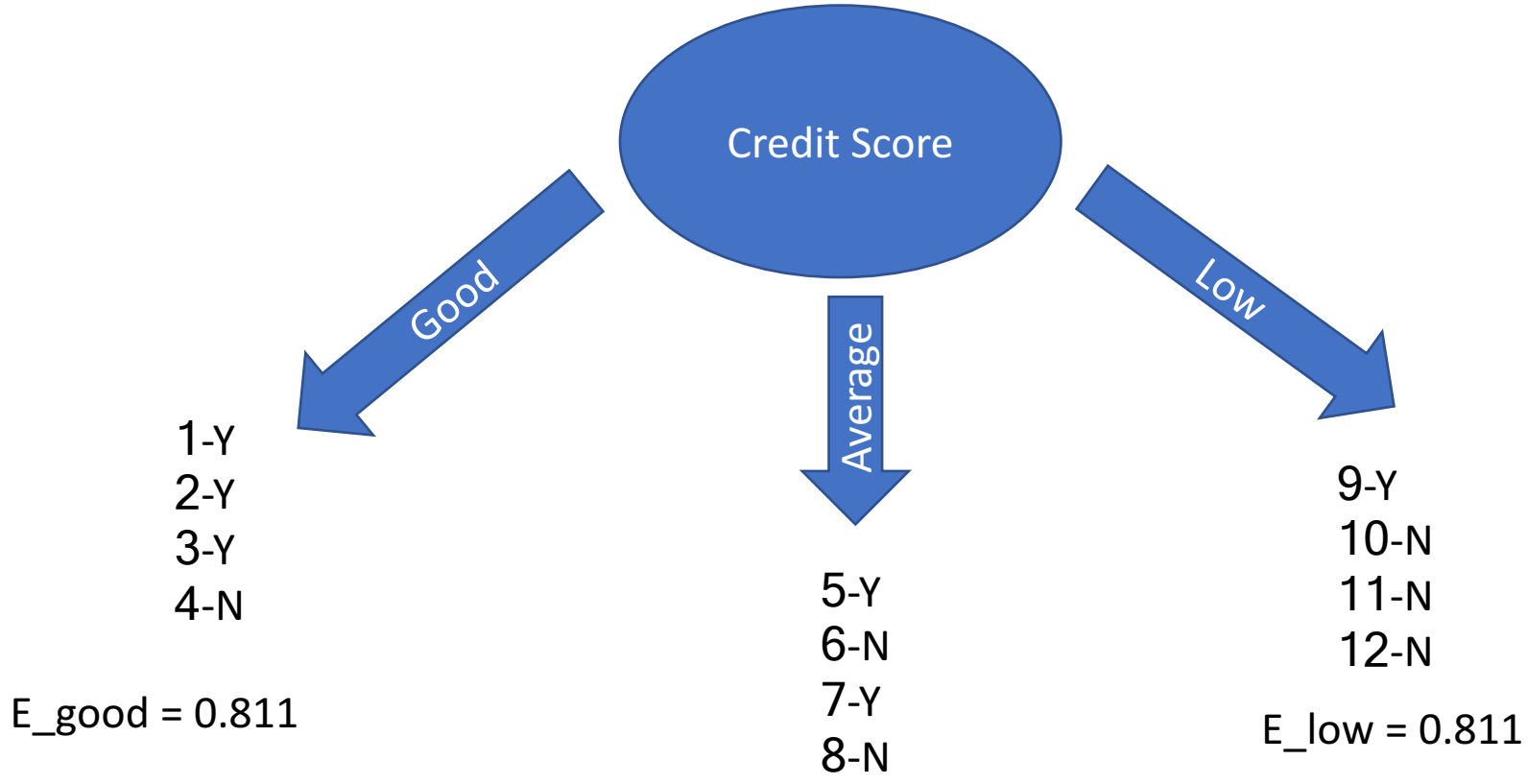


Entropy of “Low” Branch

E_low = -(1/4) * math.log2(1/4) - (3/4) * math.log2(3/4)

E_low = 0.811

Option 1 – Credit Score

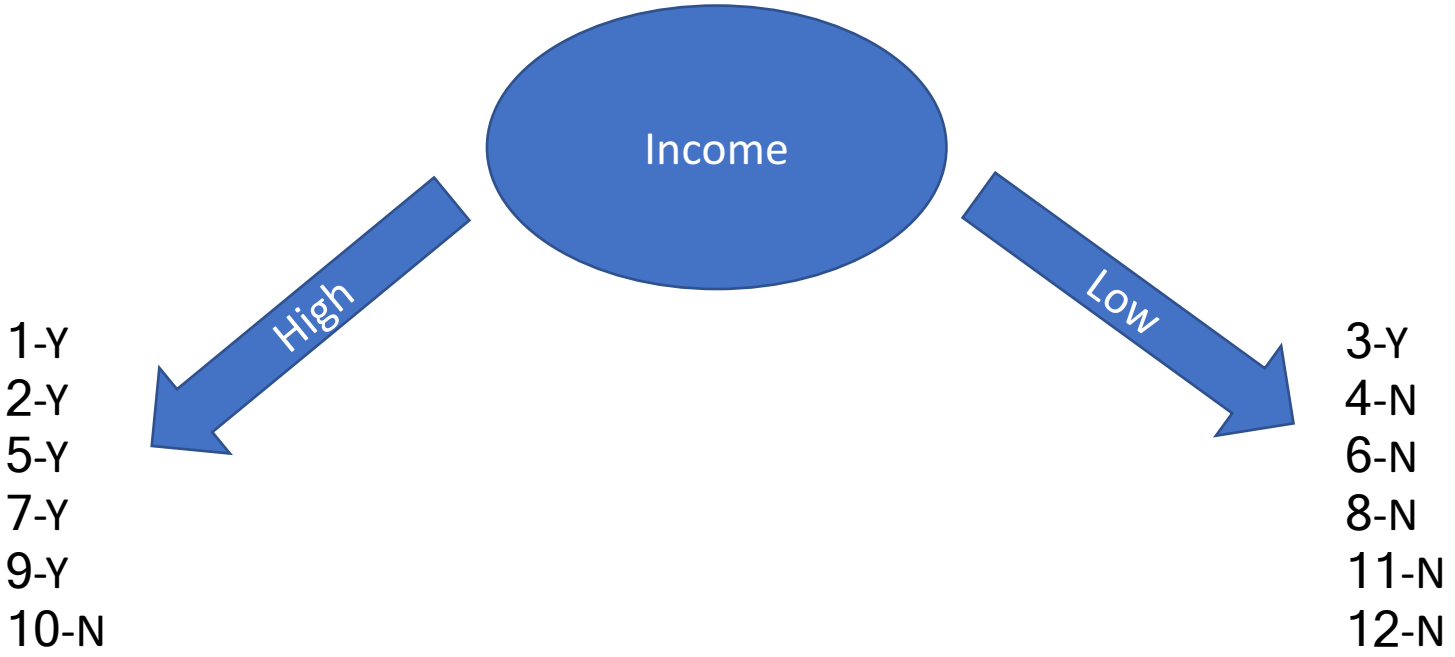


Now we compute the weighted average

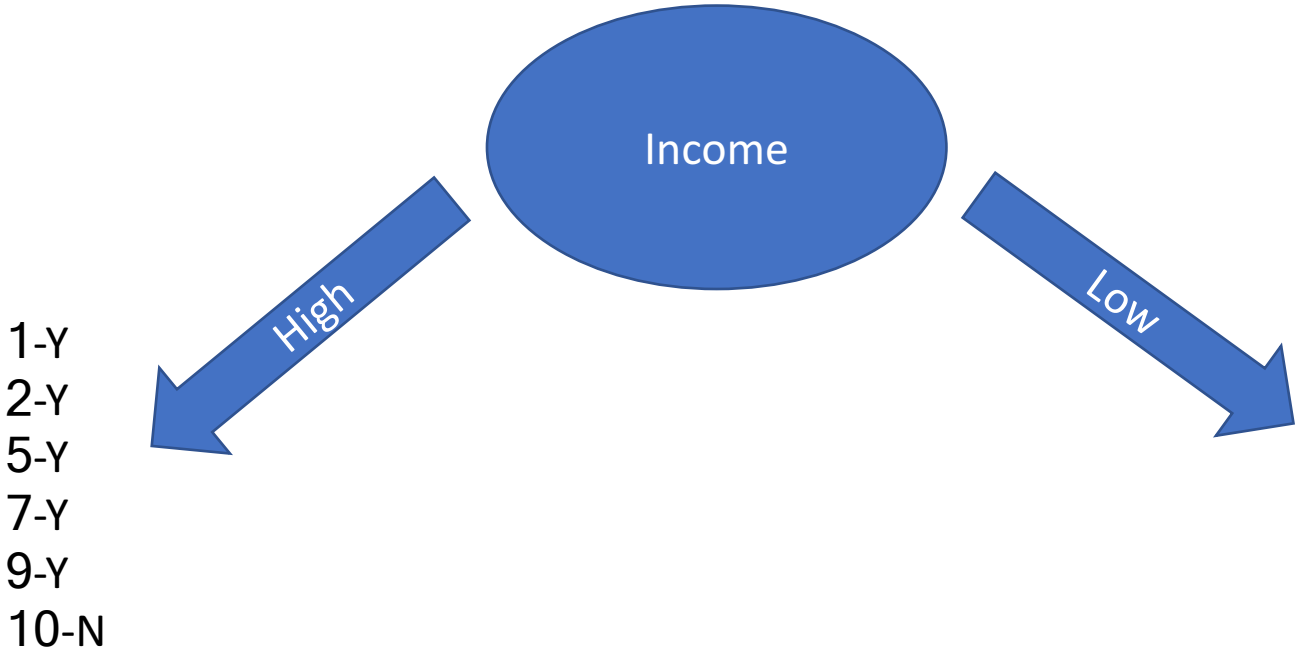
E_creditScore = $(4/12) * 0.811 + (4/12) * 1.0 + (4/12) * 0.811$

E_creditScore = 0.874

Option 2 - Income



Option 2 - Income

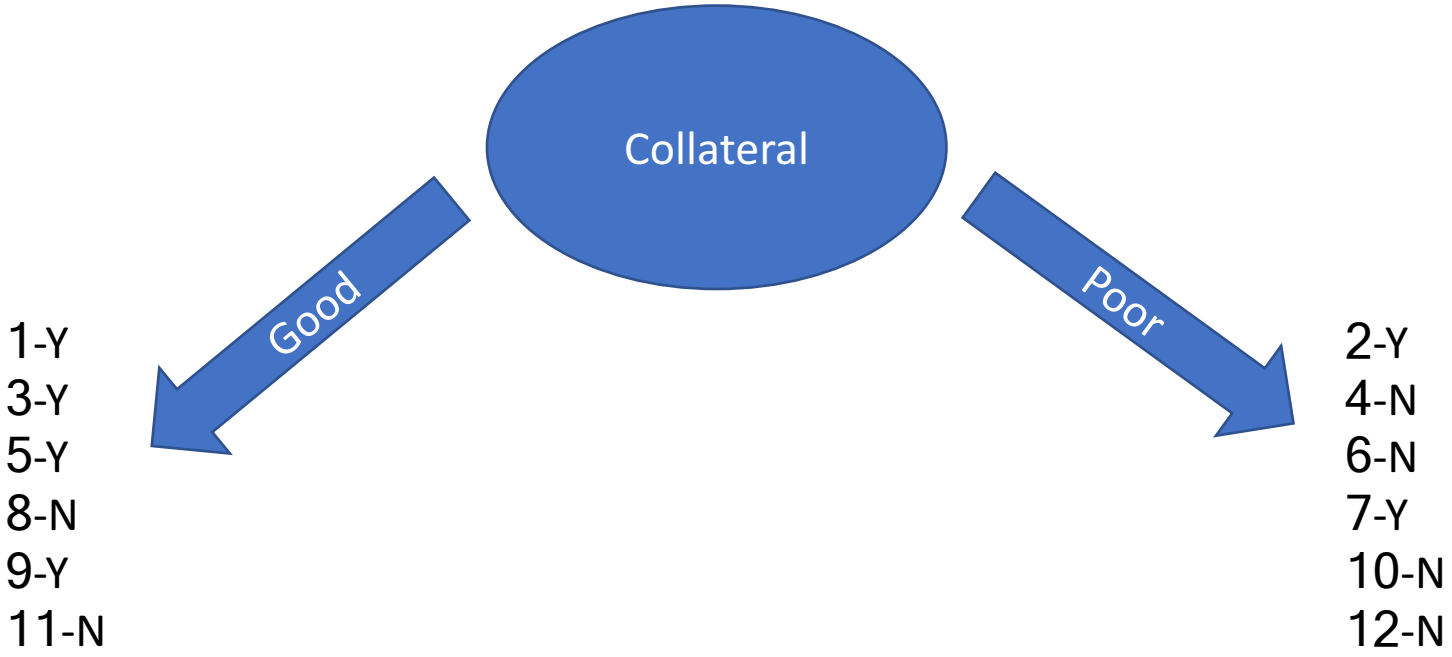


E_high = -(5/6) * math.log2(5/6) - (1/6) * math.log2(1/6)
E_high = 0.650

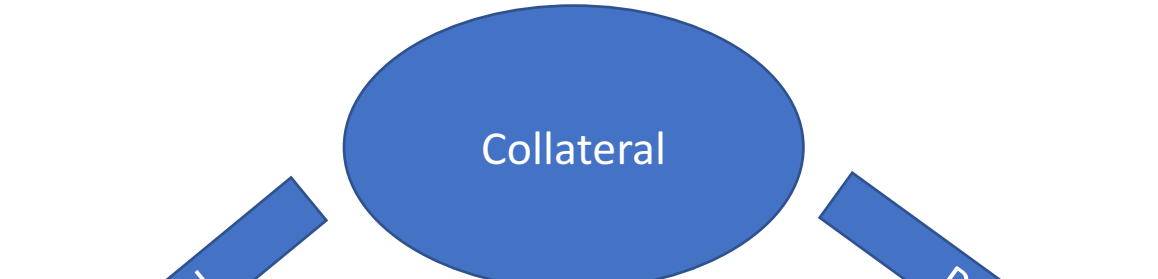
E_low = -(1/6) * math.log2(1/6) - (5/6) * math.log2(5/6)
E_low = 0.650

E_income = (6/12) * 0.650 + (6/12) * 0.650
E_income = 0.650

Option 3 - Collateral



Option 3 - Collateral



1-Y
3-Y
5-Y
8-N
9-Y
11-N

2-Y
4-N
6-N
7-Y
10-N
12-N

E_good = -(4/6) * math.log2(4/6) - (2/6) * math.log2(2/6)
E_good = 0.918

E_poor = -(2/6) * math.log2(2/6) - (4/6) * math.log2(4/6)
E_poor= 0.918

E_collateral = (6/12) * 0.918 + (6/12) * 0.918
E_collateral = 0.918

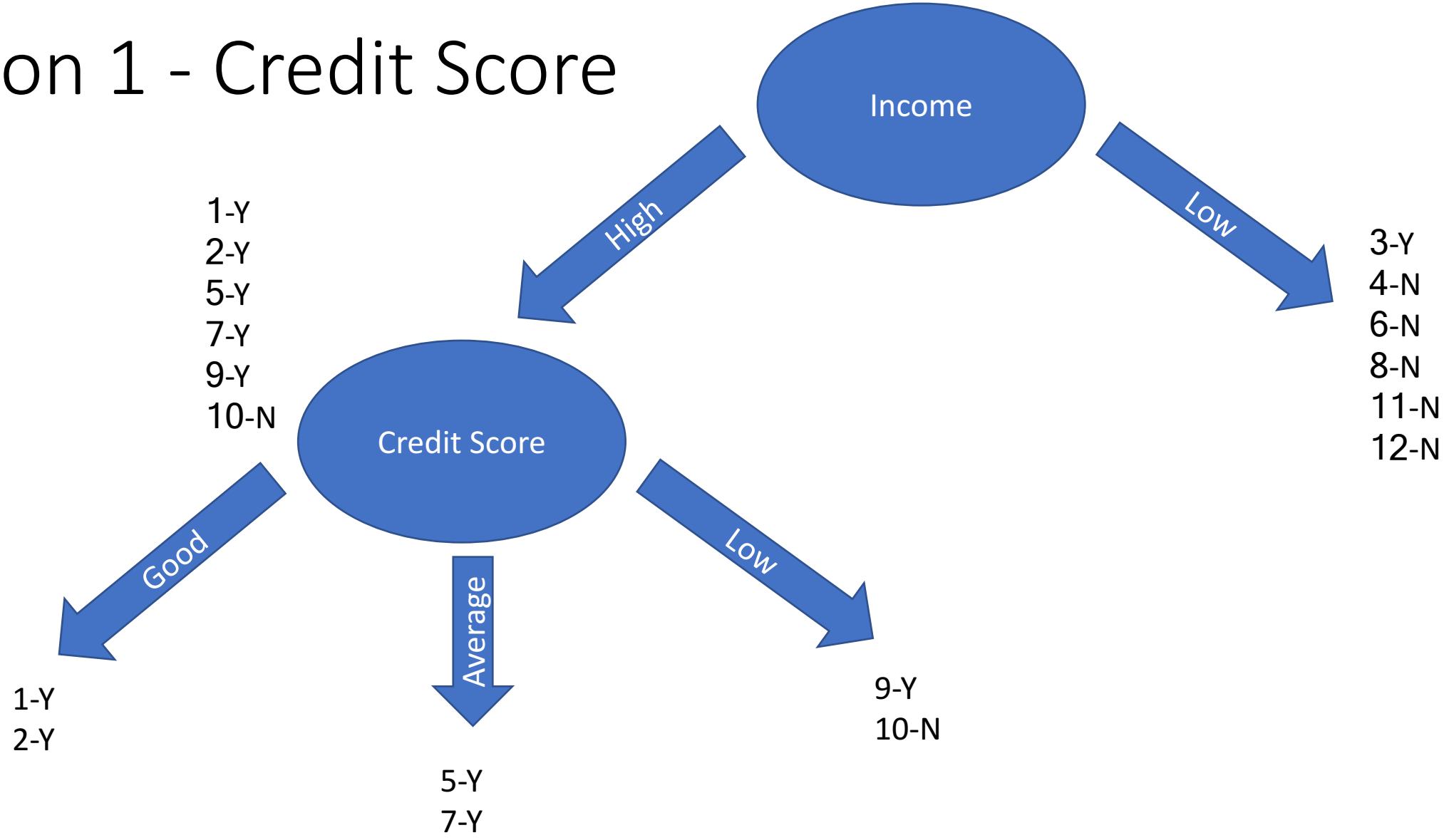
Options

1. Entropy for Credit Score = 0.874
2. Entropy for Income = 0.650
3. Entropy for Collateral = 0.918

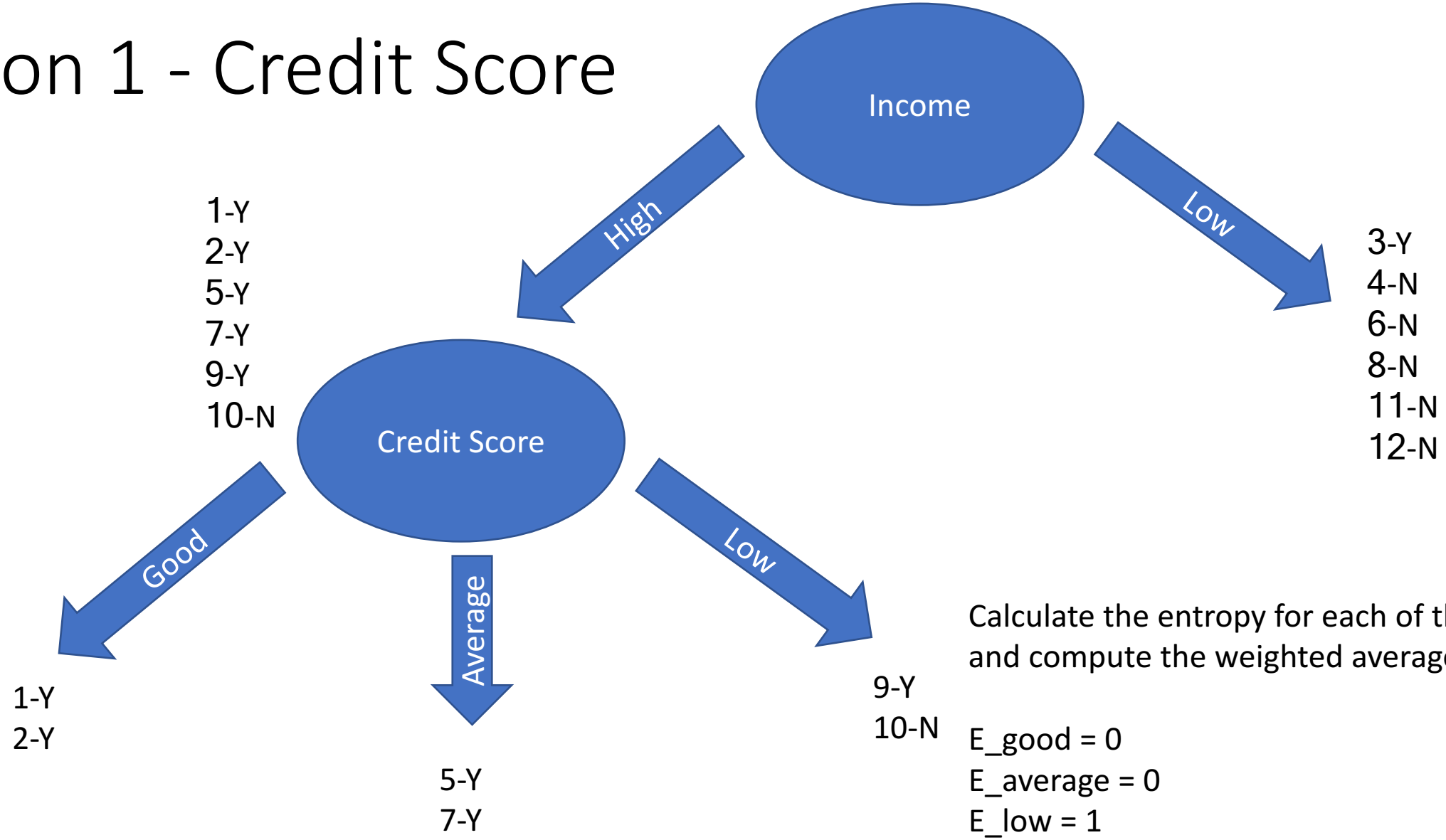
So the lowest Entropy (best Information Gain) is Income!

This means we pick Income as the root node, and work on the branches beneath it, starting with the left branch.

Option 1 - Credit Score



Option 1 - Credit Score

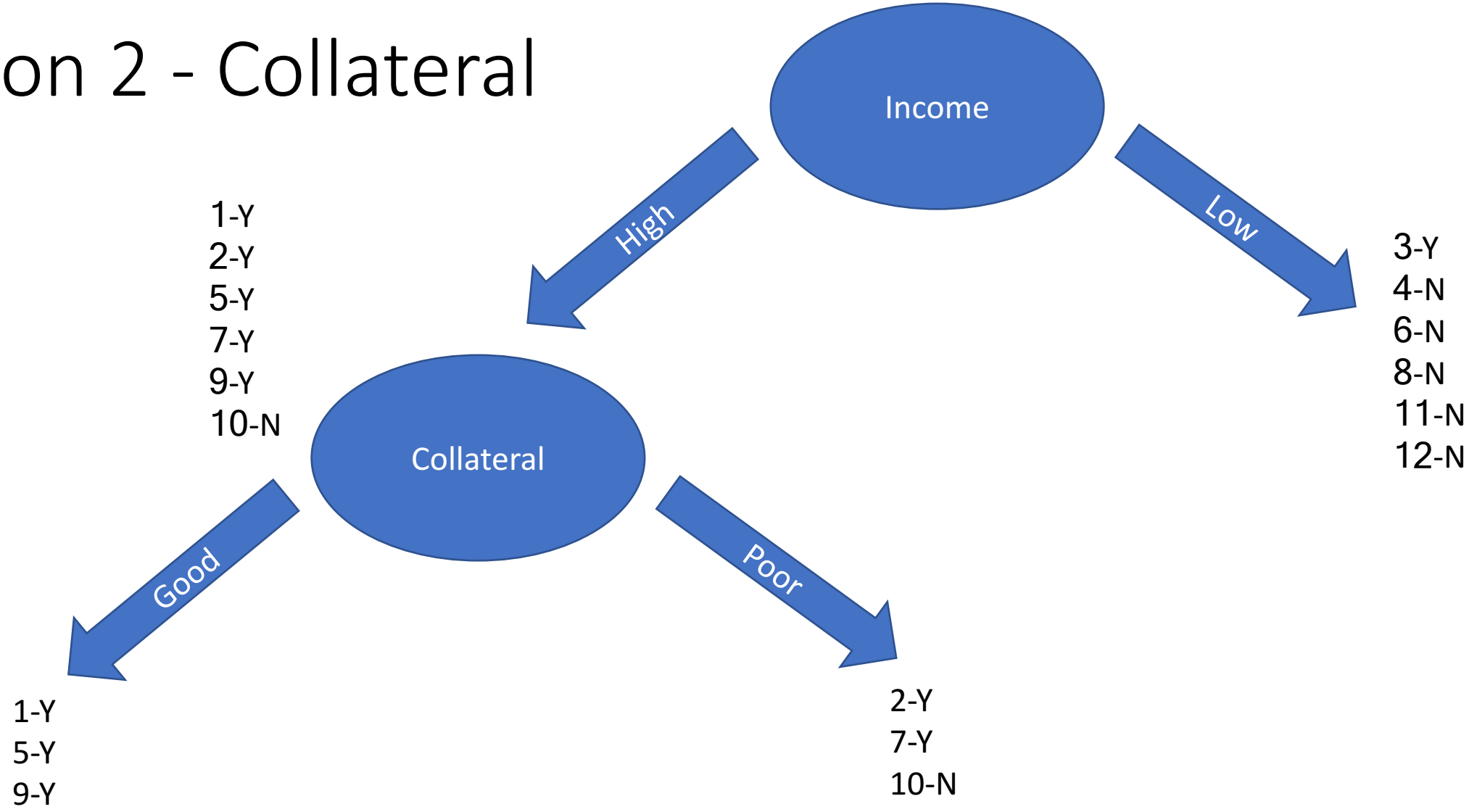


Calculate the entropy for each of these, and compute the weighted average:

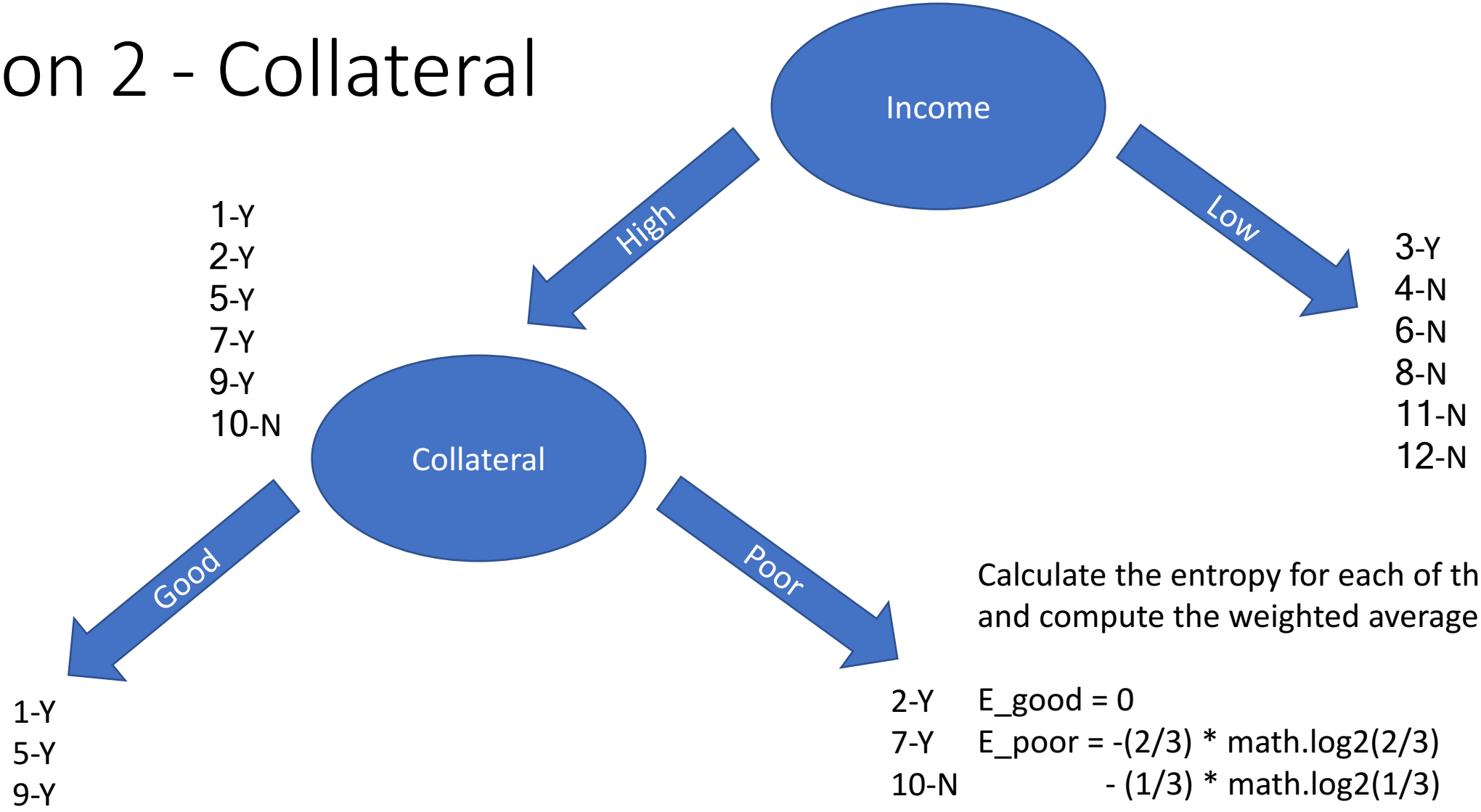
E_good = 0
E_average = 0
E_low = 1

E_creditScore = (2/6) * 0 + (2/6) * 0 + (2/6) * 1
E_creditScore = 0.333

Option 2 - Collateral



Option 2 - Collateral



Calculate the entropy for each of these, and compute the weighted average:

E_good = 0

E_poor = -(2/3) * math.log2(2/3)
- (1/3) * math.log2(1/3)
E_poor = 0.918

E_creditScore = (3/6) * 0 + (3/6) * 0.918
E_creditScore = 0.459

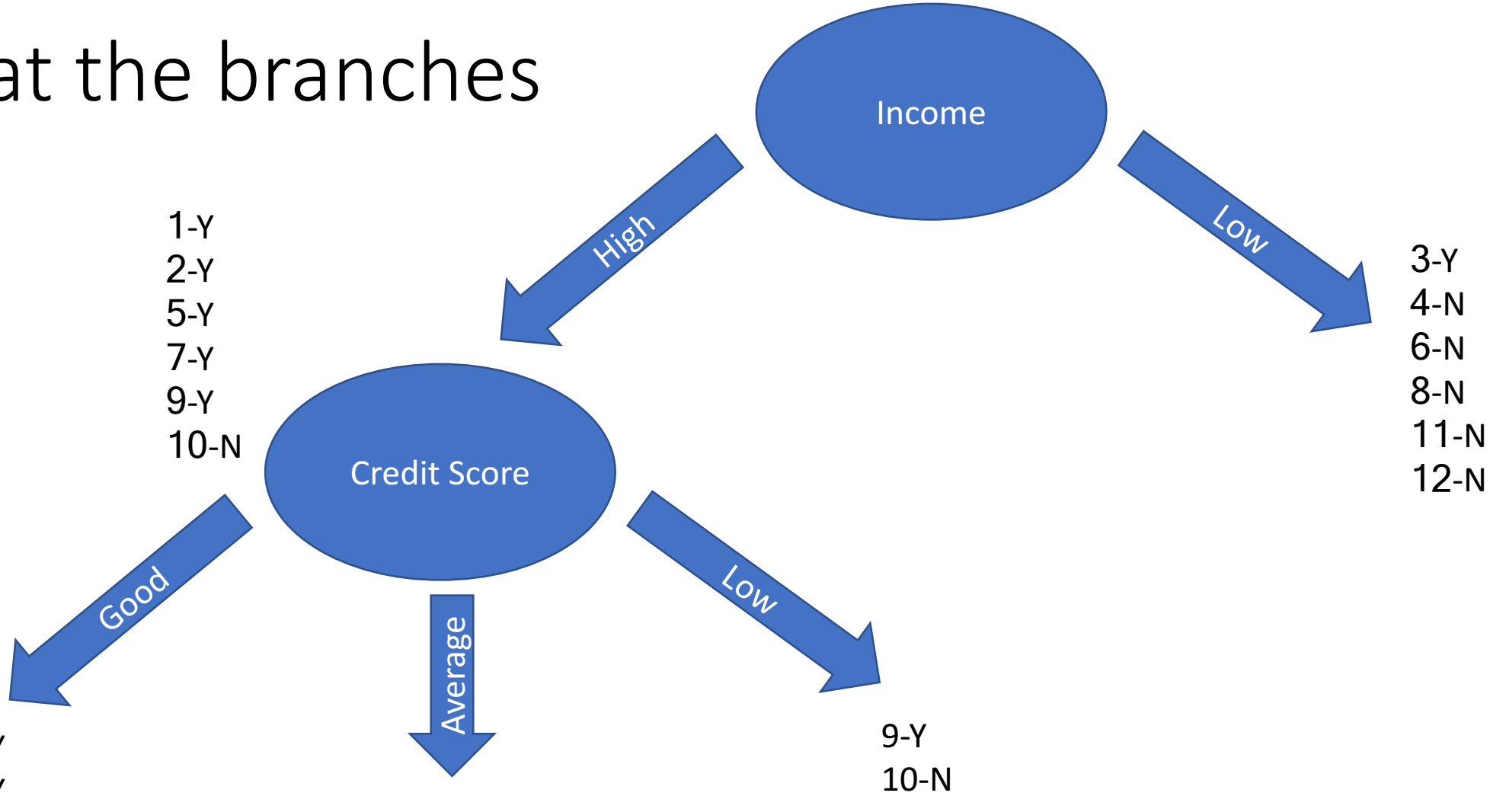
Options

1. Entropy for Income = 0.333
2. Entropy for Collateral = 0.459

So the lowest Entropy (best Information Gain) is: Income!

So we select this and work down its branches.

Look at the branches

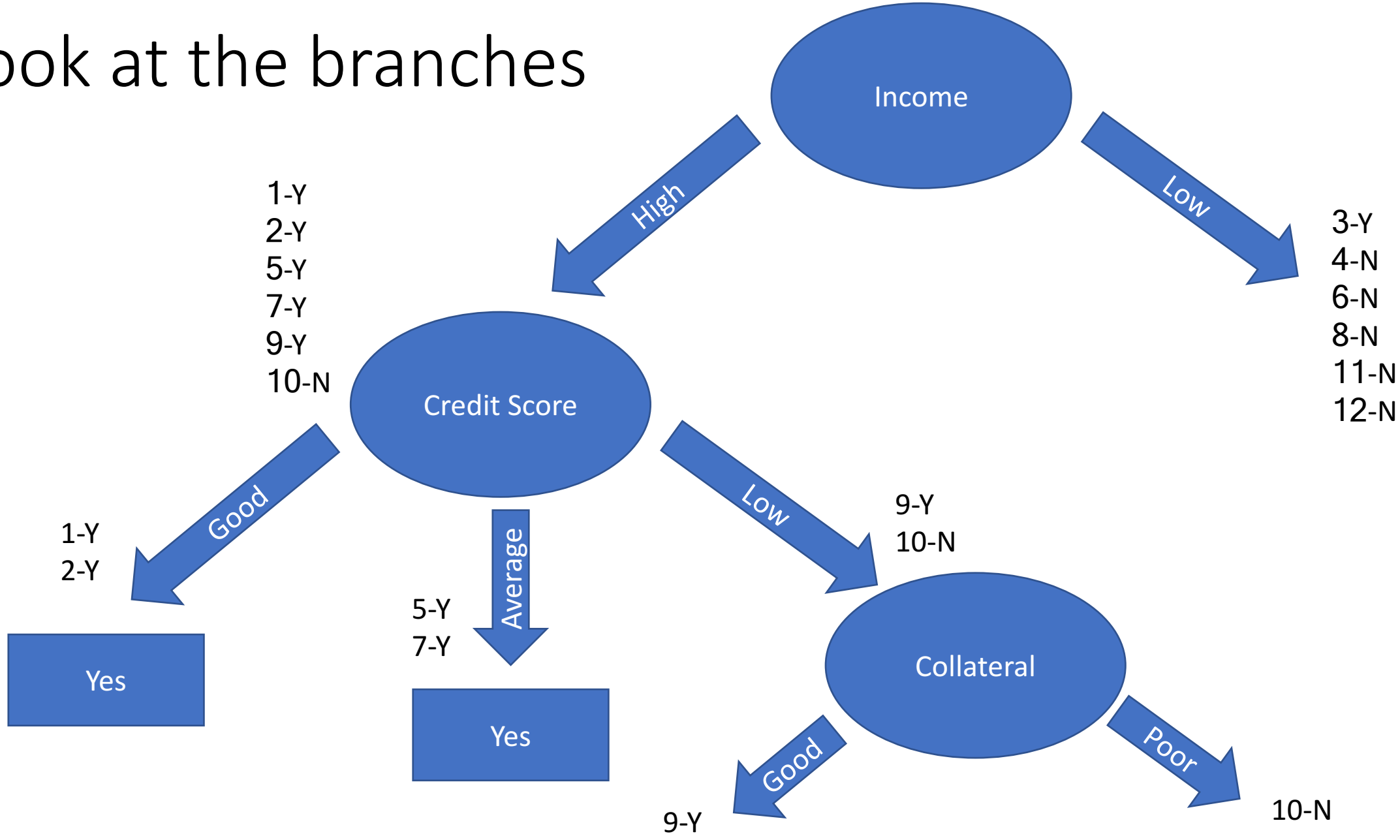


1. All the same class, so make a leaf node

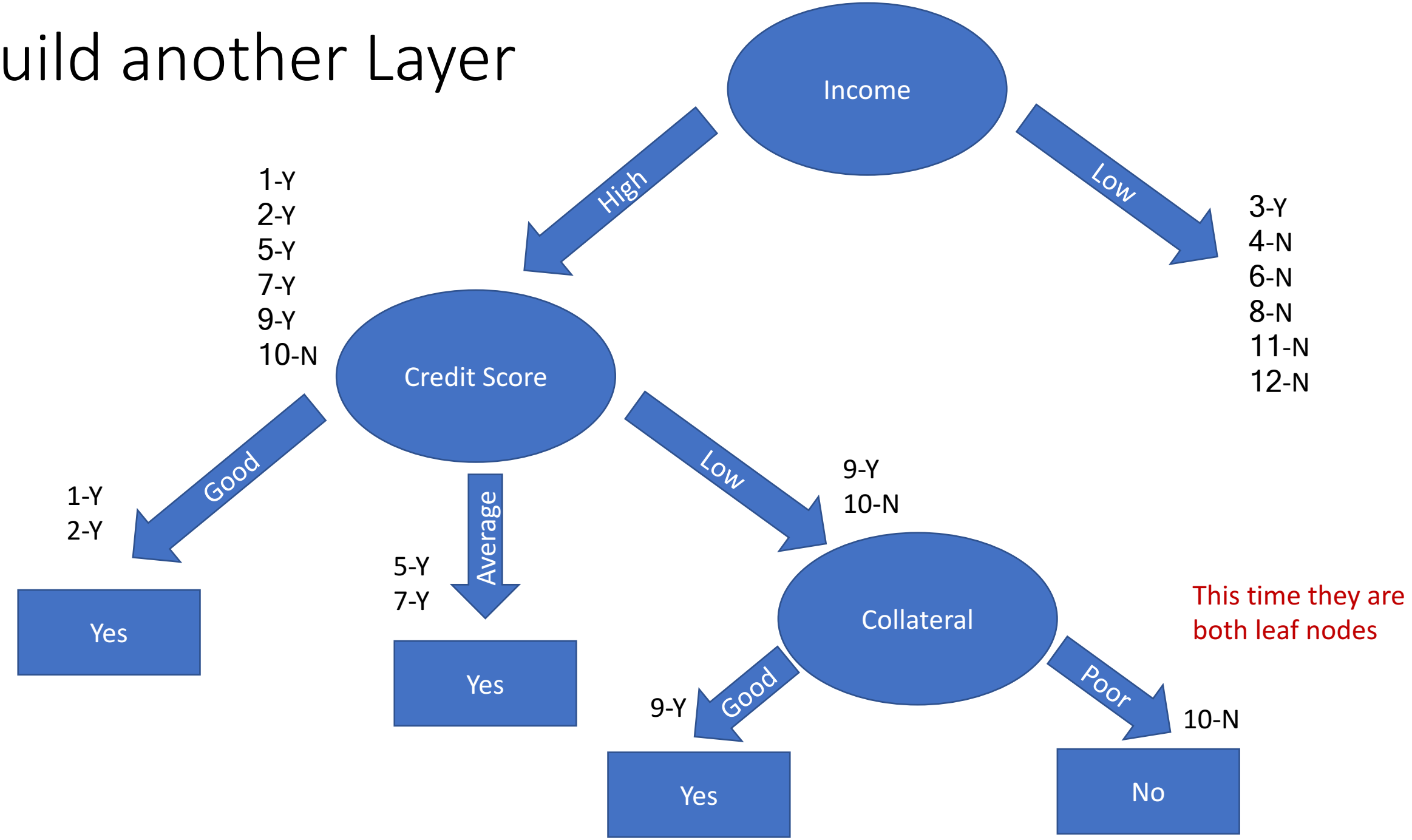
2. All the same class, so make a leaf node

3. Different class, and we have another attribute, so we recur…

Look at the branches



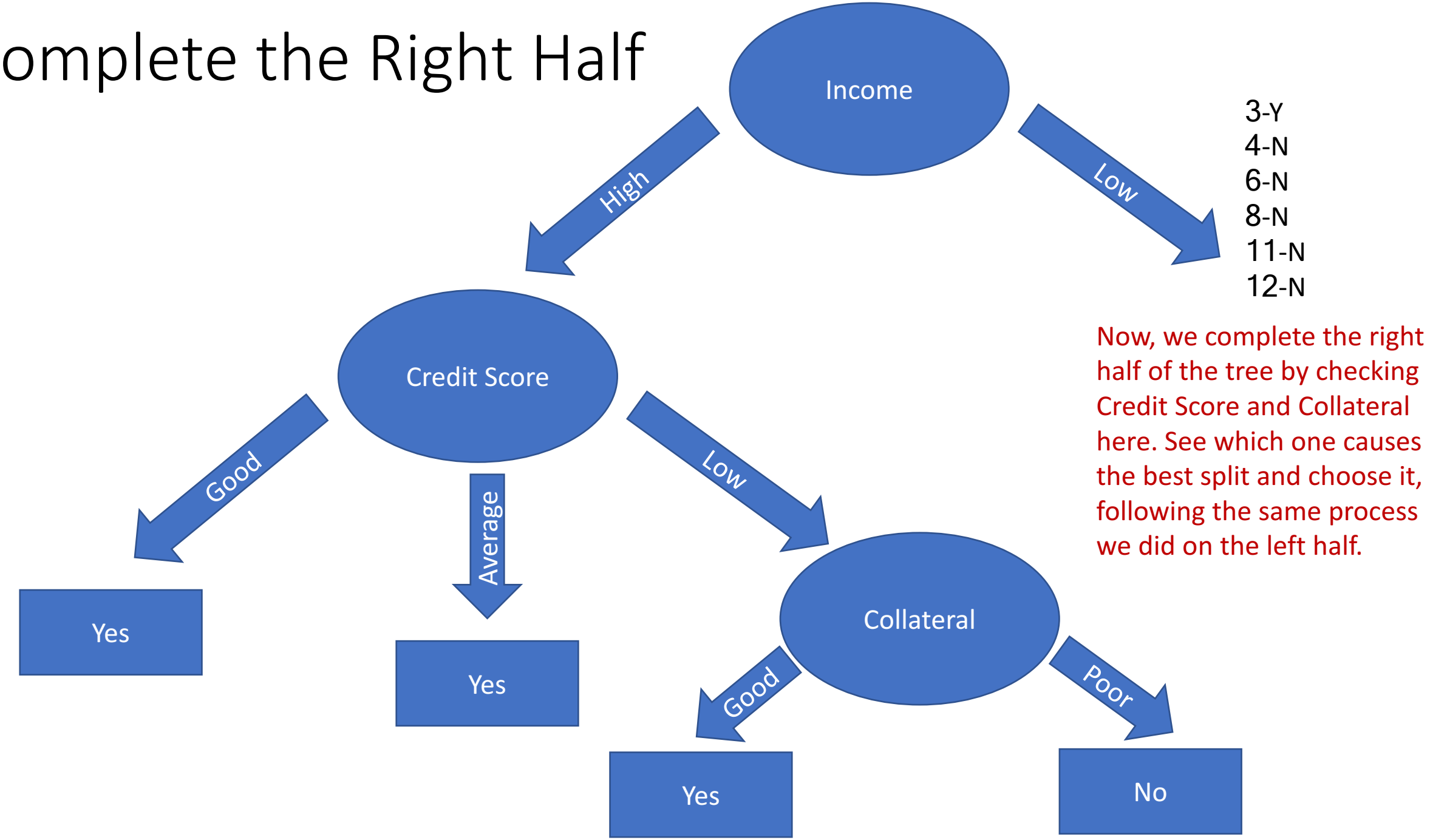
Build another Layer



Keep Building

Now we have completed the left half of the tree. We repeat the process for the right half of the tree.

Complete the Right Half



Now, we complete the right half of the tree by checking Credit Score and Collateral here. See which one causes the best split and choose it, following the same process we did on the left half.

Complete the Process

Please complete this process with your team.