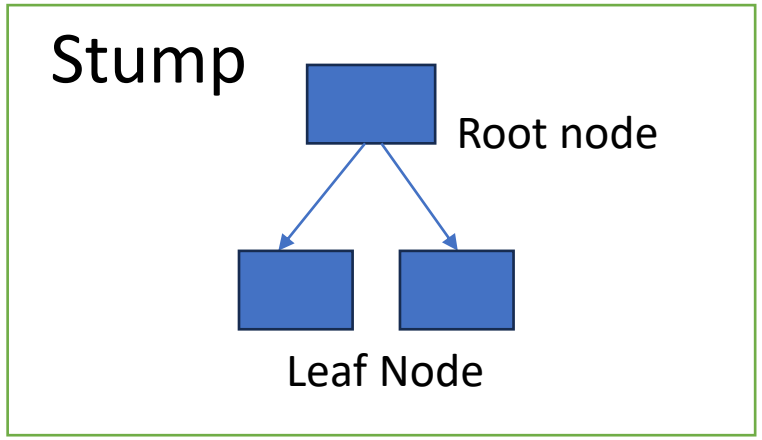
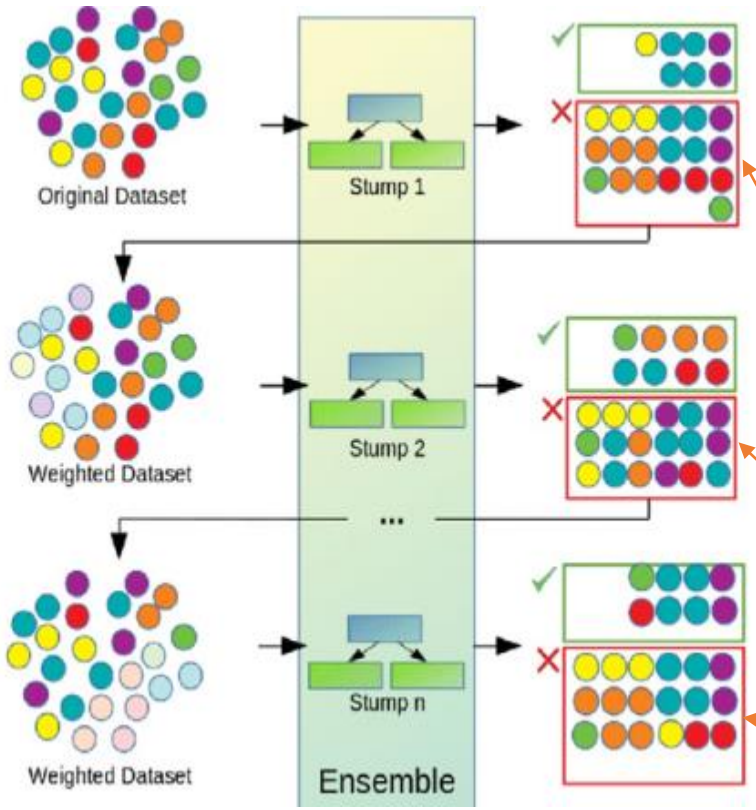
Ada Boost:

- Ada Boost or Adaptive Boosting
- It works same as normal Boosting algorithm.
- Transforms weak learners to strong learners
- Reassign weight to each instance, higher weight for incorrectly classified. This reduces bias and variance.
- Used for both classification and regression problems.



No fixed depth, AdaBoost takes only stumps

3 ideas behind Ada Boost:



Stumps can use only one variable to make a decision so it is a **Weak Learner**.

3 ideas of Ada Boost:

1. **Ada Boost** combines lot of weak learners to make classification. Weak learners are always **stumps**.
2. Some **stumps** get more say in classification than others.
3. Each **stump** is made by taking the previous **stump's** mistakes into account.

Misclassified data / Incorrectly Classified data

Example:

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease
Yes	Yes	205	Yes
No	Yes	180	Yes
Yes	No	210	Yes
Yes	Yes	167	Yes
No	Yes	156	No
No	Yes	125	No
Yes	No	168	No
Yes	Yes	172	No

Here we have to predict Heart Disease (Output) and other 3 columns are Input.

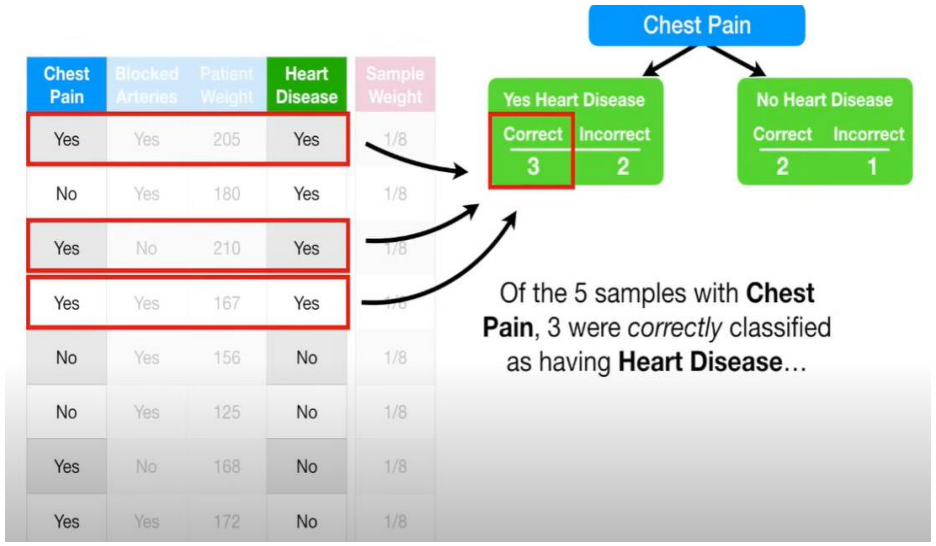
- First to predict the patient has heart disease or not , we give each sample a weight that indicates how important it is correctly classified.

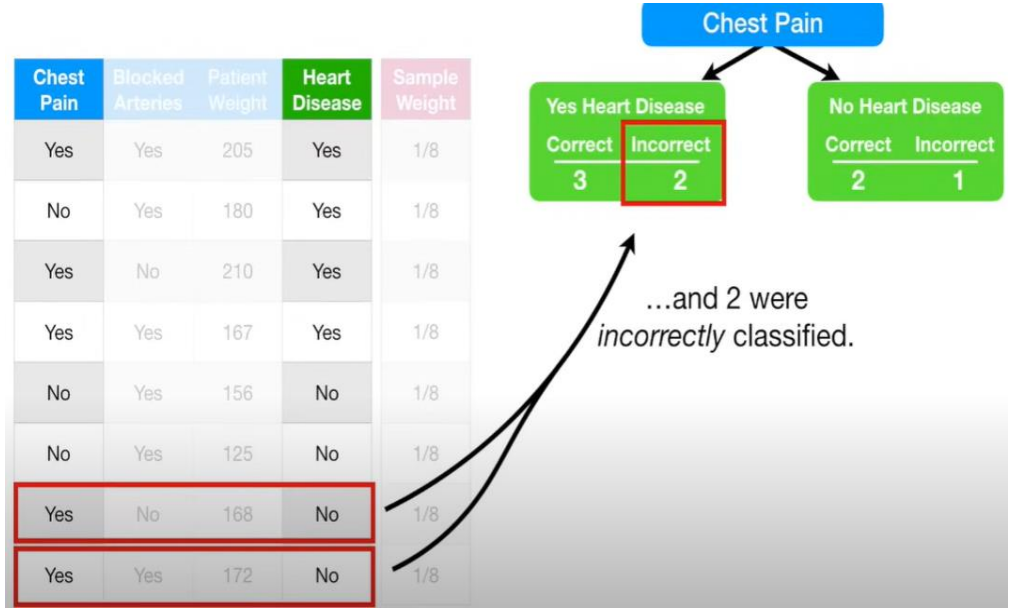
1. At the start , all samples get same weight

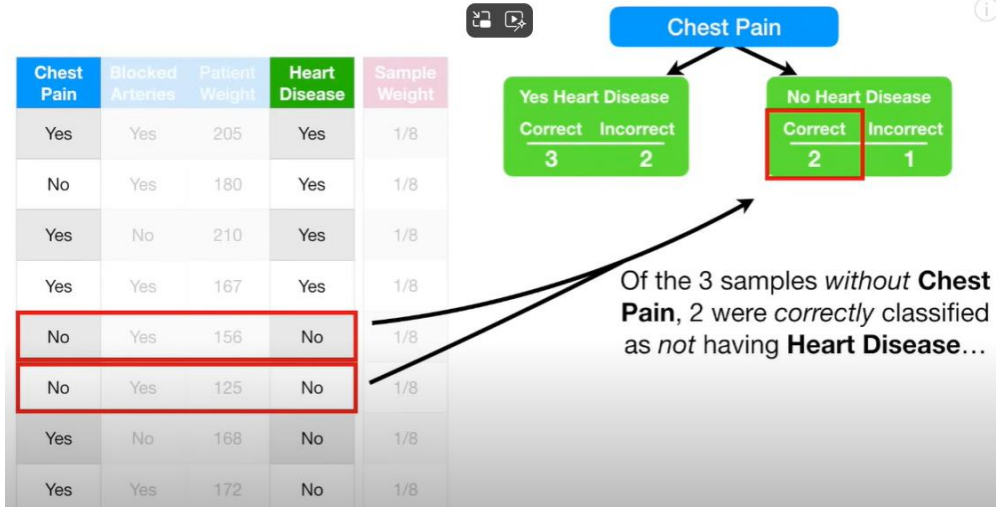
$$\text{Sample Weight} = \frac{1}{\text{Total no. of samples}} = 1/8$$

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	1/8
No	Yes	180	Yes	1/8
Yes	No	210	Yes	1/8
Yes	Yes	167	Yes	1/8
No	Yes	156	No	1/8
No	Yes	125	No	1/8
Yes	No	168	No	1/8
Yes	Yes	172	No	1/8

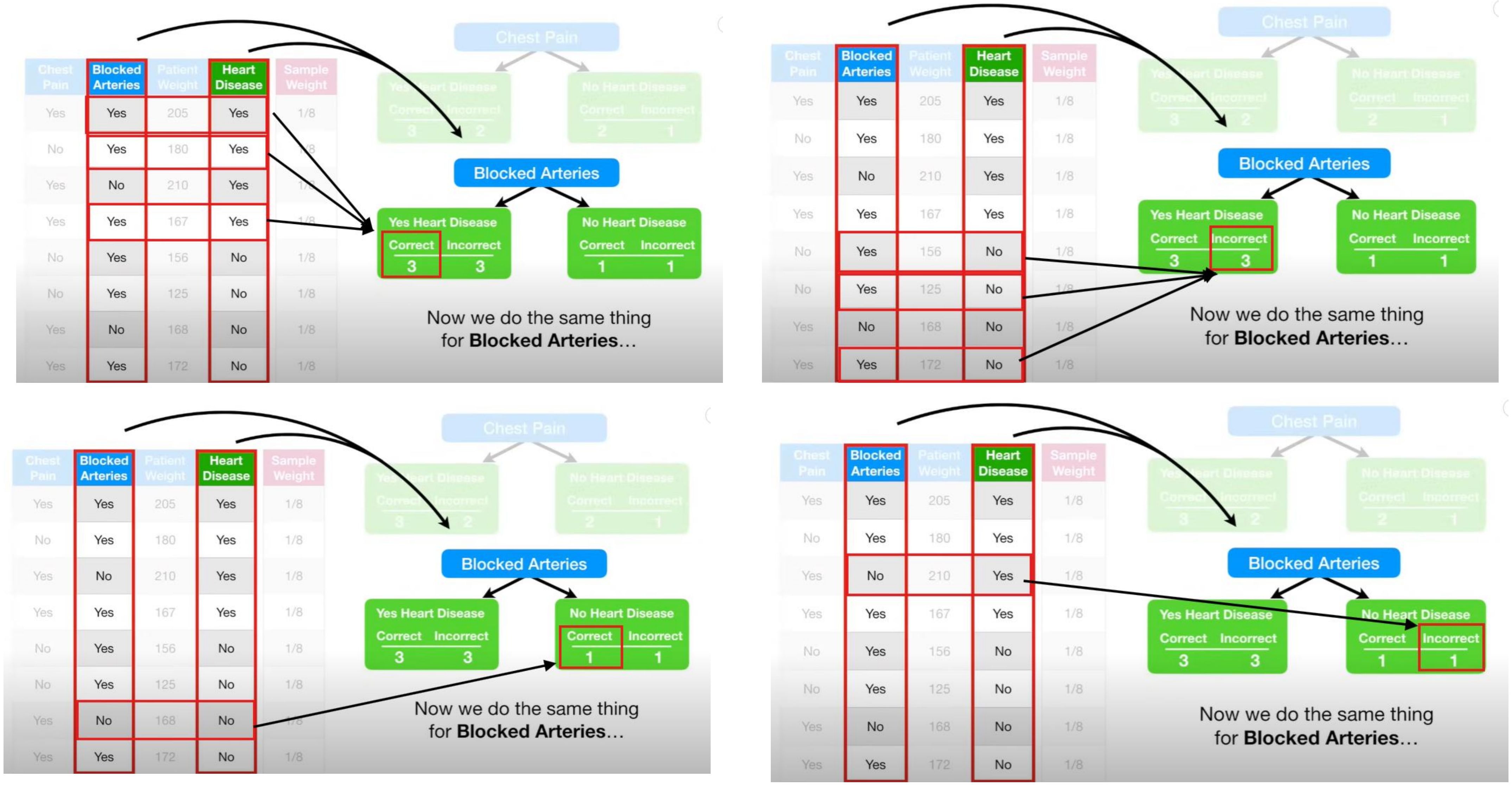
Chest Pain Classifies:





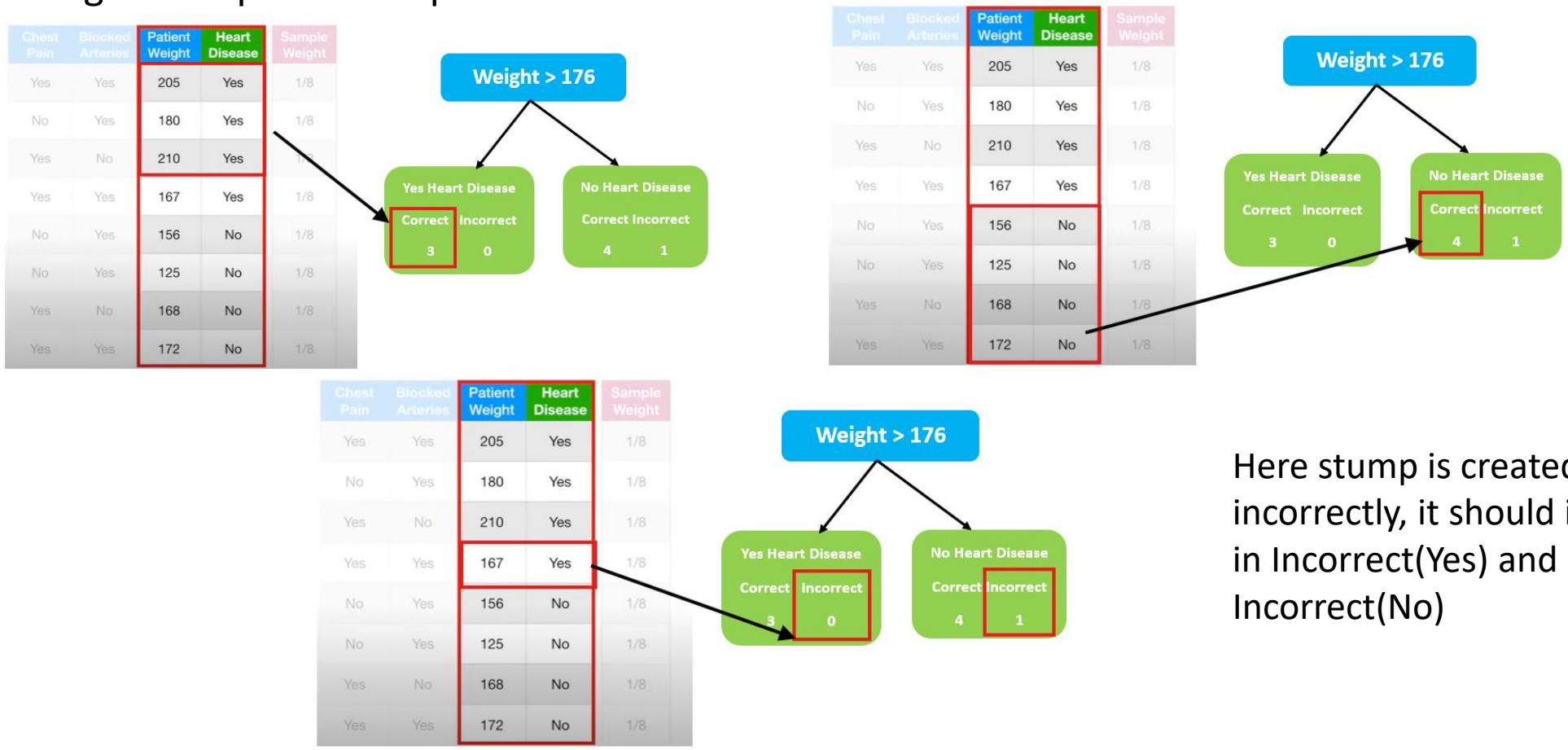


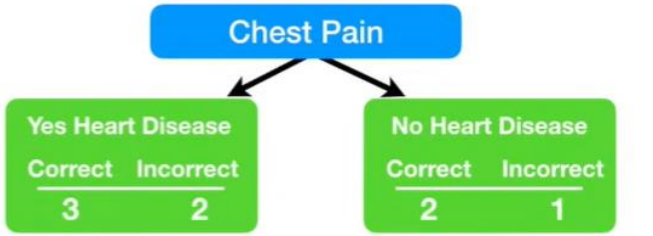
Blocked Arteries:



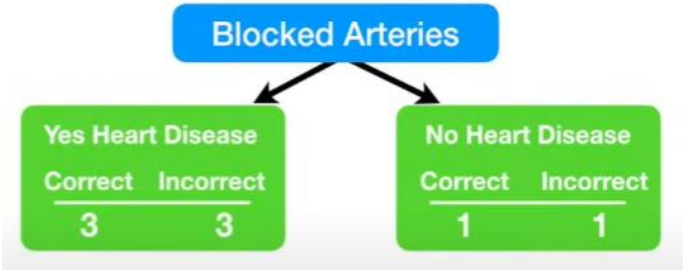
Patient Weight:

- We used the techniques described in **Decision tree StatQuest** to determine that **176** was the best weight to separate the patients.

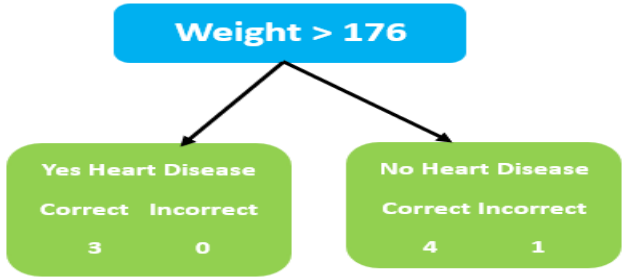




Gini Index 0.47



Gini Index 0.5



Gini Index 0.2

We calculate Gini index for all 3 stumps.

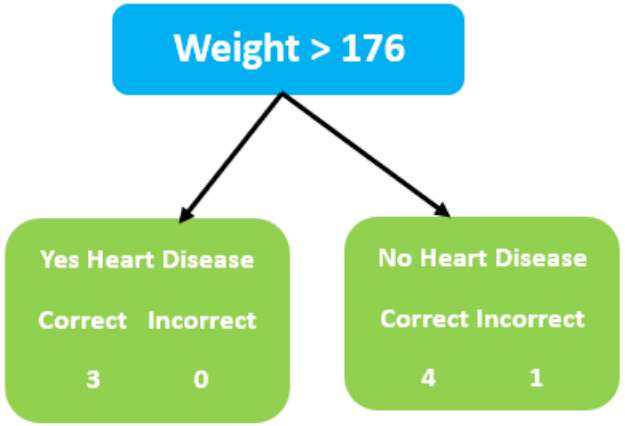
Gini Index for **Patient Weight** is the lowest

So this Patient Weight will be the first stump in the forest.

Total Error Calculation:

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	1/8
No	Yes	180	Yes	1/8
Yes	No	210	Yes	1/8
Yes	Yes	167	Yes	1/8
No	Yes	156	No	1/8
No	Yes	125	No	1/8
Yes	No	168	No	1/8
Yes	Yes	172	No	1/8

The **Total Error** for a stump is the sum of the weights associated with the *incorrectly* classified samples.



$$\text{Amount of Say} = \frac{1}{2}\log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

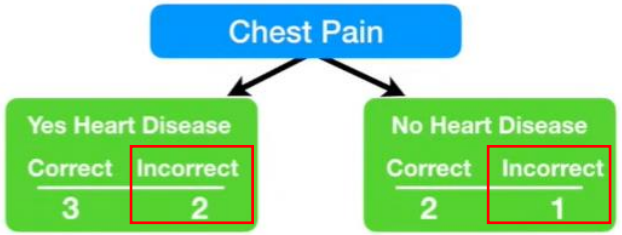
$$= \frac{1}{2}\log\left(\frac{1-1/8}{1/8}\right)$$

$$= \frac{1}{2}\log(7)$$

$$= 0.97$$

Final Classification is 0.97

Let's consider if Chest Pain would be the first stump then,

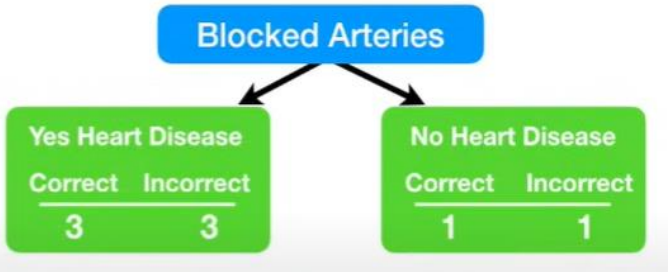


Total Error = 1/8 + 1/8 + 1/8 = 3/8

$$\text{Amount of Say} = \frac{1}{2}\log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

$$= \tfrac{1}{2}\log\left(\frac{1-3/8}{3/8}\right) = 0.42$$

If Blocked Arteries would be first stump then,



Total Error = 1/8 + 1/8 + 1/8 + 1/8 = 4/8

$$\text{Amount of Say} = \frac{1}{2}\log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

$$= \tfrac{1}{2}\log\left(\frac{1-4/8}{4/8}\right) = 0.2$$

- Initially all the sample weight has same weight.
- After first stump we found below row as incorrect sample weight so we need to increase that sample weight and decrease other sample weight.

Formula to increase/decrease sample weight:

$$\text{New Sample Weight} = \text{sample weight} \times e^{\text{amount of say}}$$

$$= \frac{1}{8} e^{\text{amount of say}}$$

$$= \frac{1}{8} e^{0.97} = \frac{1}{8} \times 2.64 = 0.33$$

This is larger than the old one (i.e) 1/8=0.125

Increase sample weight

$$\text{New Sample Weight} = \text{sample weight} \times e^{-\text{amount of say}}$$

$$= \frac{1}{8} e^{-\text{amount of say}}$$

$$= \frac{1}{8} e^{-0.97} = \frac{1}{8} \times 0.38 = 0.05$$

This is smaller than the old one (i.e) 1/8=0.125

Decrease sample weight

Adding New sample weight and Normalized weight:

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight	New Weight
Yes	Yes	205	Yes	1/8	0.05
No	Yes	180	Yes	1/8	0.05
Yes	No	210	Yes	1/8	0.05
Yes	Yes	167	Yes	1/8	0.33
No	Yes	156	No	1/8	0.05
No	Yes	125	No	1/8	0.05
Yes	No	168	No	1/8	0.05
Yes	Yes	172	No	1/8	0.05

If we add the new weight we get 0.68 and now we have to divide each new weight by 0.68 to get normalized weight

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight	New Weight	Norm. Weight
Yes	Yes	205	Yes	1/8	0.05	0.07
No	Yes	180	Yes	1/8	0.05	0.07
Yes	No	210	Yes	1/8	0.05	0.07
Yes	Yes	167	Yes	1/8	0.33	0.49
No	Yes	156	No	1/8	0.05	0.07
No	Yes	125	No	1/8	0.05	0.07
Yes	No	168	No	1/8	0.05	0.07
Yes	Yes	172	No	1/8	0.05	0.07

Create next stump by calculating Gini Index:

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	0.07
No	Yes	180	Yes	0.07
Yes	No	210	Yes	0.07
Yes	Yes	167	Yes	0.49
No	Yes	156	No	0.07
No	Yes	125	No	0.07
Yes	No	168	No	0.07
Yes	Yes	172	No	0.07

With this sample weight we calculate Gini Index to determine which variable should split the next stump.

If number is between 0 to 0.07 then we put this sample into new collection of sample

(0.07- 0.14)

(0.14- 0.21)

(0.21- 0.70)

(0.70- 0.77)

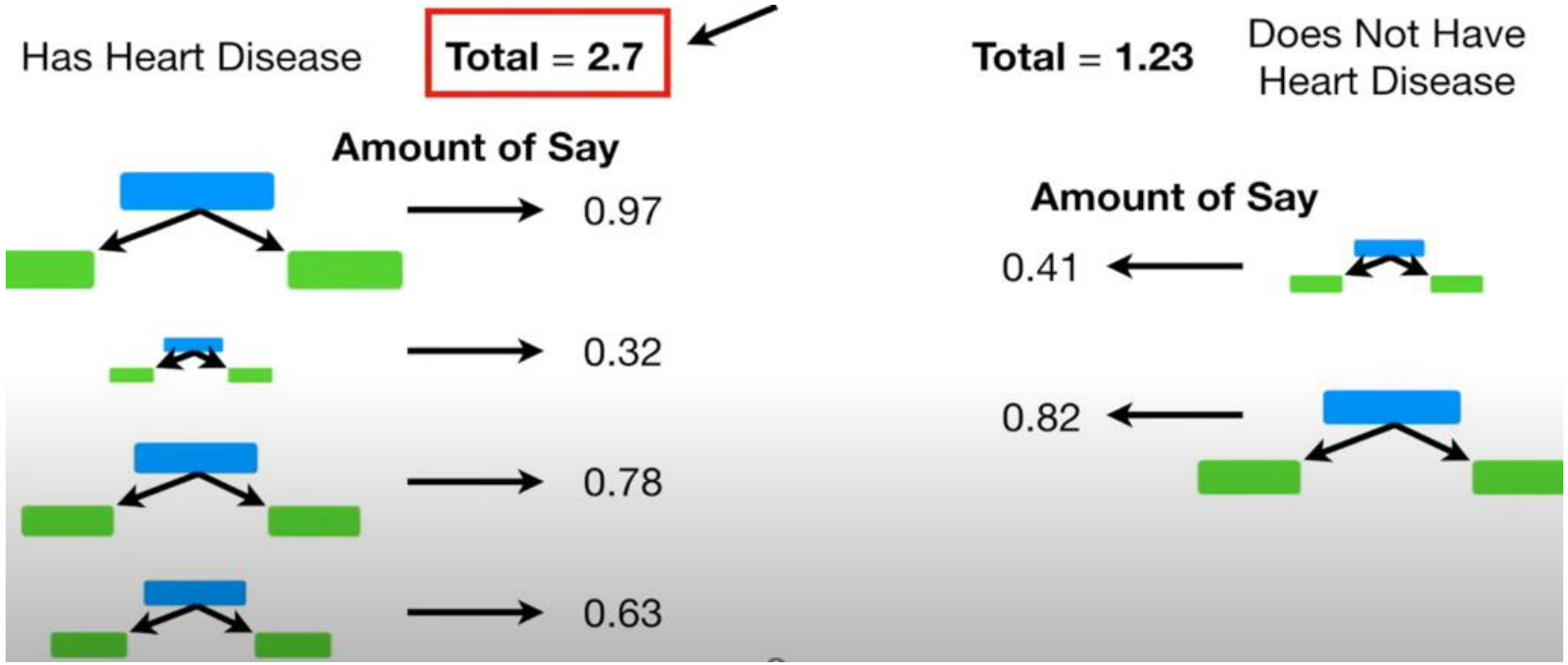
(0.77- 0.84)

(0.84- 0.91)

(0.91-0.98)

For example, the first number I picked was 0.42 then it takes 4th row (0.21-0.70) to the new dataset . Similarly it creates new table with original table size and follow the steps done earlier, to calculate Total error calculation and so on.

Patient as Heart disease since this as large sum



So this is how Ada boost convert weak learners to strong learners.