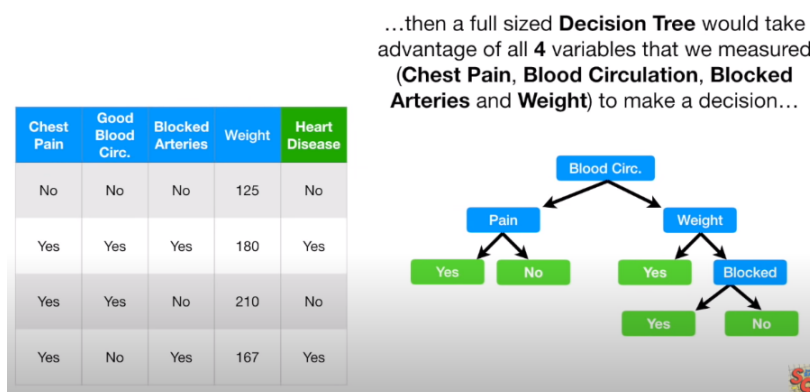


AdaBoost

AdaBoost 使用了 Decision Trees 和 Random Forests 的一些概念，再加上一些变化。在使用 Random Forest 的时候，每一次 Generate 的树都是一颗 Full Sized Tree，训练出来的树的 Tree Height 都是不一样的，因为使用 Random Forests 并没有设定 Maximum Tree Height。

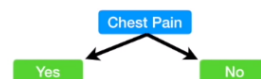
但是在 AdaBoost 里，每棵树只有一个 Root 和两个 leaves，一颗只有一个 Node 和 2 Leaves 的树被称之为 Stump，所以 AdaBoost 也可以称之为 Forest of Stumps。Stumps 在 Classification 里不能给出很好的决定。



在 Decision Tree 里，每一个 Parameters 都被考虑到，考虑的 Parameters 越多，能给出的答案就比只考虑一个 Parameters 的树更正确。

...but a **Stump** can only use one variable to make a decision.

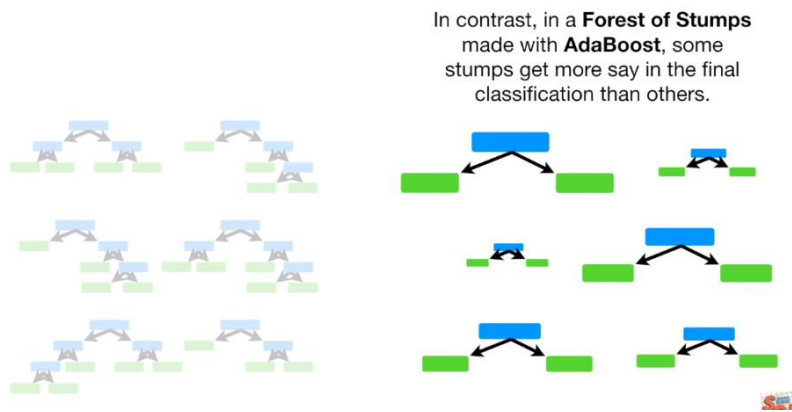
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes



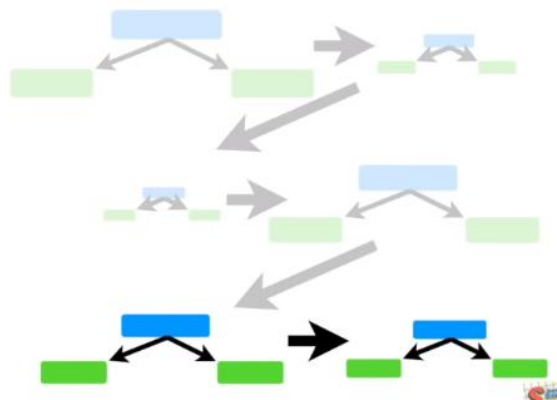
Thus, **Stumps** are technically "weak learners".

However, that's the way **AdaBoost** likes it, and it's one of the reasons why they are so commonly combined.

而每一个 Stumps 只考虑一个 Parameters，所以 Stumps 也被称之为 Weak Learners，但这也是 AdaBoost 的特点。



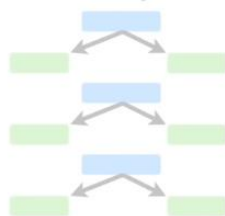
在 Random Forests 里，每一棵树 Predict 的结果都是同等对待，但是在 AdaBoost 里，每棵树是有不同的 Weight。



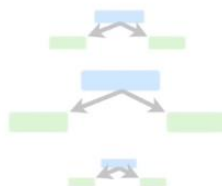
在 Random Forests 里，先 Generate 出那一棵树都不是很重要，但是在 AdaBoost 里，先 Generate 的那一棵树会影响之后被 Generate 出来的树的结果。

小总结

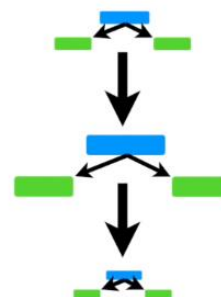
1) **AdaBoost** combines a lot of “weak learners” to make classifications. The weak learners are almost always **stumps**.



2) Some **stumps** get more say in the classification than others.



3) Each **stump** is made by taking the previous **stump's** mistakes into account.



Create Forest of Stumps

要 Create Stump 之前，需要给每一笔 Data 一个 Sample Weight (Sample Weight 是一个几率，加起来等于 1)，每一个 Data 初始的 Sample Weight 都是一样的。

$$\text{Sample Weight} = \frac{1}{\text{Total Data}(s)}$$

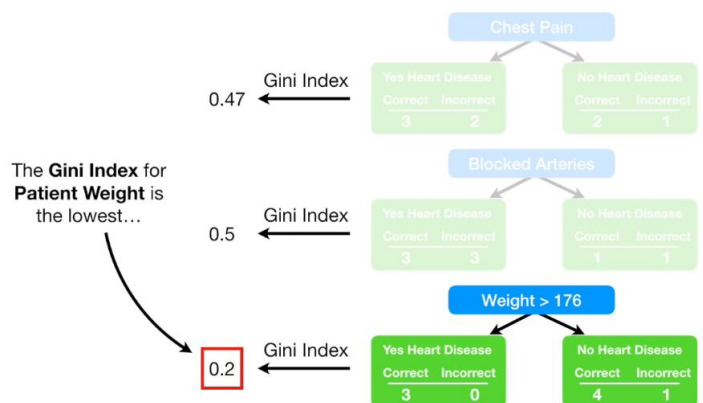
Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	1/8
No	Yes	180	Yes	1/8
Yes	No	210	Yes	1/8
Yes	Yes	167	Yes	1/8
No	Yes	156	No	1/8
No	Yes	125	No	1/8
Yes	No	168	No	1/8
Yes	Yes	172	No	1/8

At the start, all samples get the same weight...

$$\frac{1}{\text{total number of samples}} = \frac{1}{8}$$

...and that makes the samples all equally important.

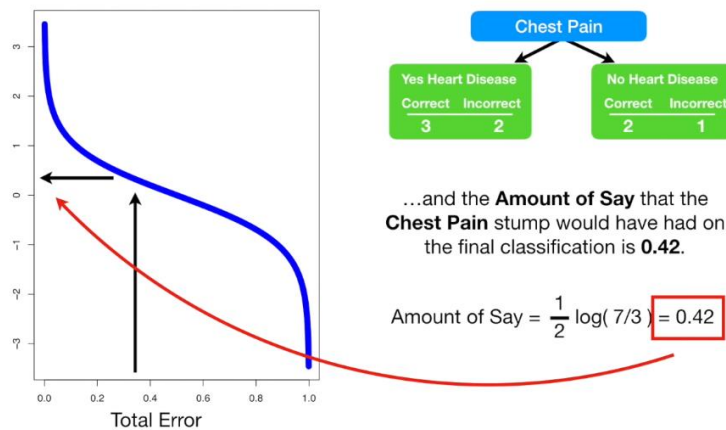
上图的 Dataset 里总共有 8 个 Data，所以每一个 Data 的 Sample Weight 初始的时候都是一样的。当设定好初始 Sample Weight 之后，需要开始创建第一个 Stump，这时候需要开始从所以 Parameters 里面选出最好的 Parameter。



在这里对每一个 Parameters 进行 Gini Impurity 的计算，有着最小 Gini Impurity 值的树就会被设定成第一颗 Stump。之后需要定义这棵 Stump 的重要程度。

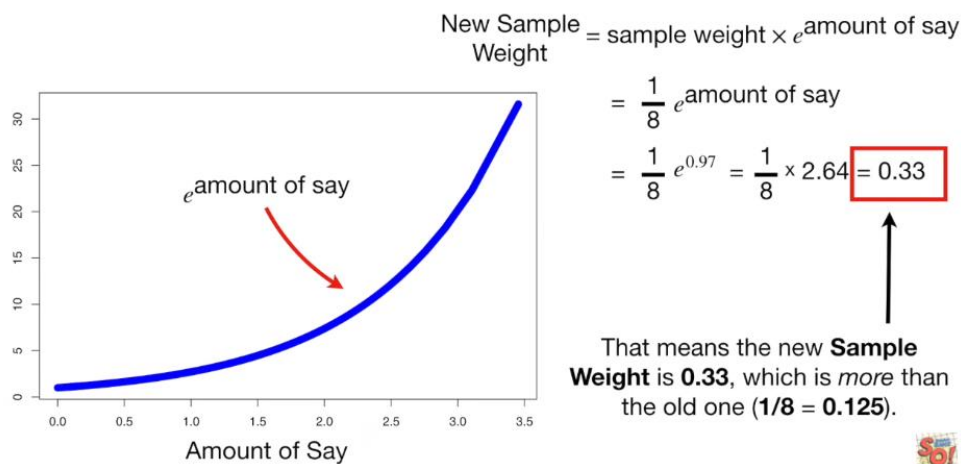


这时候把所有预测错误的 Data(s) 的 Sample Weight 加起来。这些总和就是 Total Error。



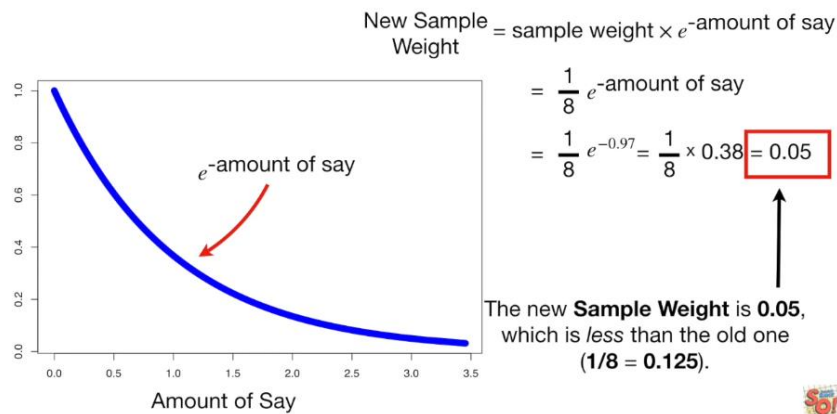
$$\text{Amount of Say} = \frac{1}{2} \log \left(\frac{1 - \text{Total Error}}{\text{Total Error}} \right)$$

当 Total Error 小的时候, Amount of Say 的值就会是比较大, 当 Total Error 逐渐变大的时候, Amount of Say 就会变小。也就是说, 当 Error 越大, 这棵树的可信度就越低。当计算好 Amount of Say 值之后, 需要对 Sample Weight 进行 Update。AdaBoost 的想法是被错误 Predict 的 Data 的 Sample Weight 需要被提高, 而被正确 Predict 的 Sample Weight 需要被降低。



$$\text{New Sample Weight} = \text{Sample Weight of Incorrect Prediction Data} * e^{\text{Amount of Say}}$$

上面的 Formula 是用来计算被错误预测的 Data。当 Amount of Say 值是大的时候, 计算出的 New Sample Weight 也会变大。



$$\text{New Sample Weight} = \text{Sample Weight of Correct Prediction Data} * e^{-\text{Amount of Say}}$$

上面这个 Formula 是用来计算 Predict 正确的 Data。当 Amount of Say 越大，计算出来的 Weight 的值就会越小。

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

So we divide each **New Sample Weight** by **0.68** to get the normalized values.

$$\text{Norm Weight} = \frac{\text{Weight}}{\text{Sum of New Weight}}$$

当计算完所有 Data 的 New Weight 之后，需要把这个 Weight 做 Normalize 的处理，为了让这些 Weight 加起来等于 1 (可以有些许 Rounding Error)，因为是概率。这时候就能用这个 Normalize 过后的 Weight 来创建下一棵 Stump。

当要创建下一棵 Stump 的时候，会重新创建一个新的 Dataset。创建这个 Dataset 有几个步骤。首先先随机 Generate 一个号码 (0 到 1 之间)，然后使用这个号码来选出对应的 Data。在这里不需要担心没被正确 Predict 的数据会被遗忘，当在 Update New Weight 的时候，Predict 错误的 Data 的 New Weight 会变大，这会让这笔 Data 更容易被选到。

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

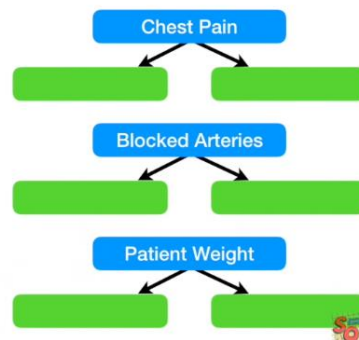
Chest Pain	Blocked Arteries	Patient Weight	Heart Disease

...and if the number is between **0.21** and **0.70** ($0.21 + 0.49 = 0.70$), then we would put this sample into the new collection of samples...

如果 Generate 出来的号码是 0 到 0.07 之间，第一笔 Data 将会被选出来，如果是 0.21 到 0.70 之间，如上图，那笔 Data 就会被选出来。会重复 Generate 一个随机的号码，然后选出对应的 Data 直到这个新的 Dataset 有着与原来 Dataset 同样数目的 Data。

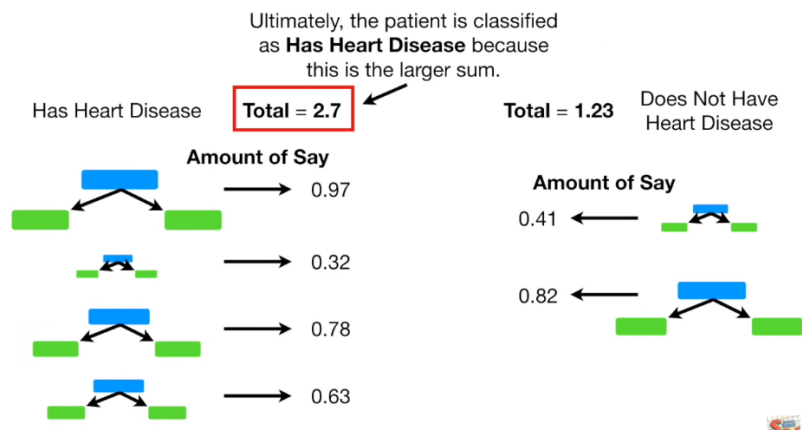
Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
No	Yes	156	No	1/8
Yes	Yes	167	Yes	1/8
No	Yes	125	No	1/8
Yes	Yes	167	Yes	1/8
Yes	Yes	167	Yes	1/8
Yes	Yes	172	No	1/8
Yes	Yes	205	Yes	1/8
Yes	Yes	167	Yes	1/8

Now we go back to the beginning and try to find the stump that does the best job classifying the new collection of samples.



当选好这组新的 Dataset 之后，一样给每一笔 Data 分配同样的 Sample Weight，重新对每一个 Parameters 进行 Gini Impurity 的计算，选出有着最低 Gini Impurity 的 Stump 然后 Update New Weight 然后再继续同样的步骤。

How AdaBoost Make Classification Decision



AdaBoost 在给出预测的时候，会把这笔 Input Data 用来跑完所有 Stumps 之后，会把有着同样 Class 的树归类在一起，然后进行 Amount of Say 的计算，有着最高 Amount of Say 的 Class，那个 Class 就是答案。

小总结

