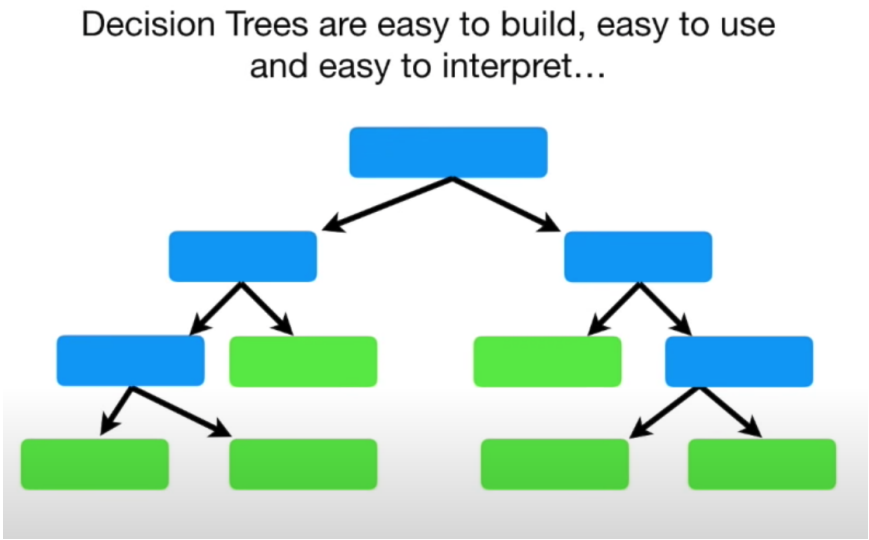
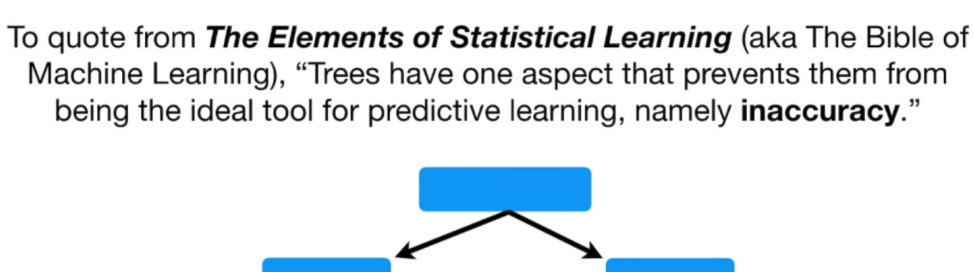
3 - Random Forest - sq

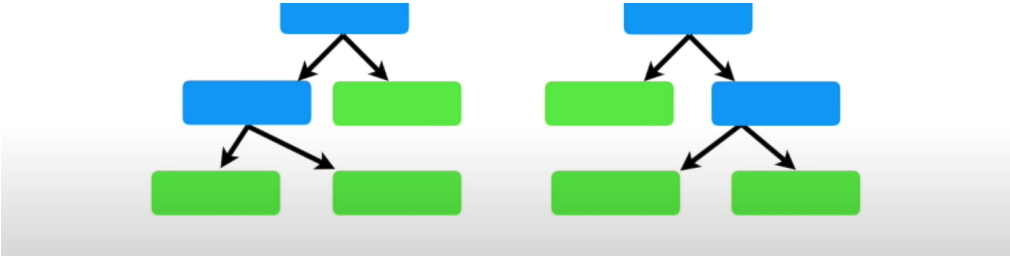
Tuesday, 17 November BE 2563 17:40

External Resources:

1. StatQuest: Random Forests Part 1 - Building, Using and Evaluating
2. StatQuest: Random Forests Part 2: Missing data and clustering



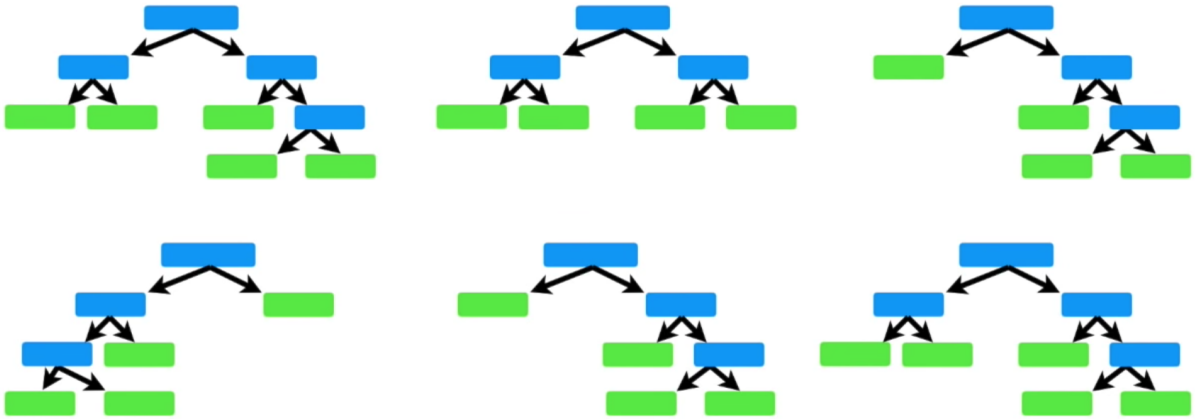




Let's talk about inaccuracy..

In other words, they work great with the data used to create them, but **they are not flexible when it comes to classifying new samples**.

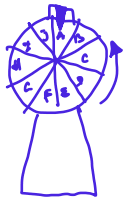
The good news is that **Random Forests** combine the simplicity of decision trees with flexibility resulting in a vast improvement in accuracy.



Step 1: Create a “bootstrapped” dataset.

Original Dataset

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



Bootstrapped Dataset

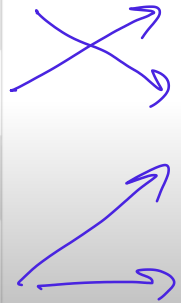
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease

To create a bootstrapped dataset that is the same size as the original, we just randomly select samples from the original dataset.

The important detail is that we’re allowed to pick the same sample more than once.

Original Dataset

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



Bootstrapped Dataset

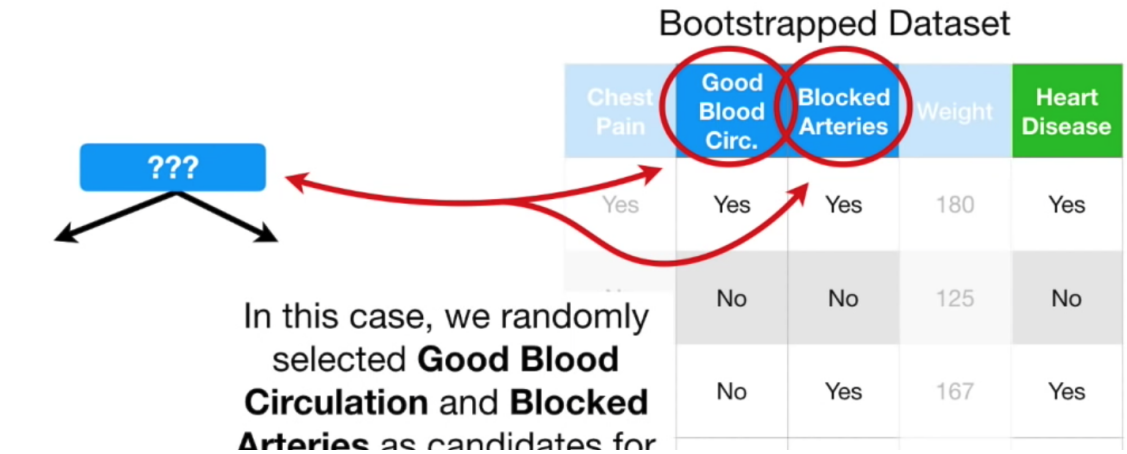
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

Step 2: Create a decision tree using the bootstrapped dataset, but only use a random subset of variables (or columns) at each step.

In this example, we will only consider 2 variables (columns) at each step.

Bootstrapped Dataset

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



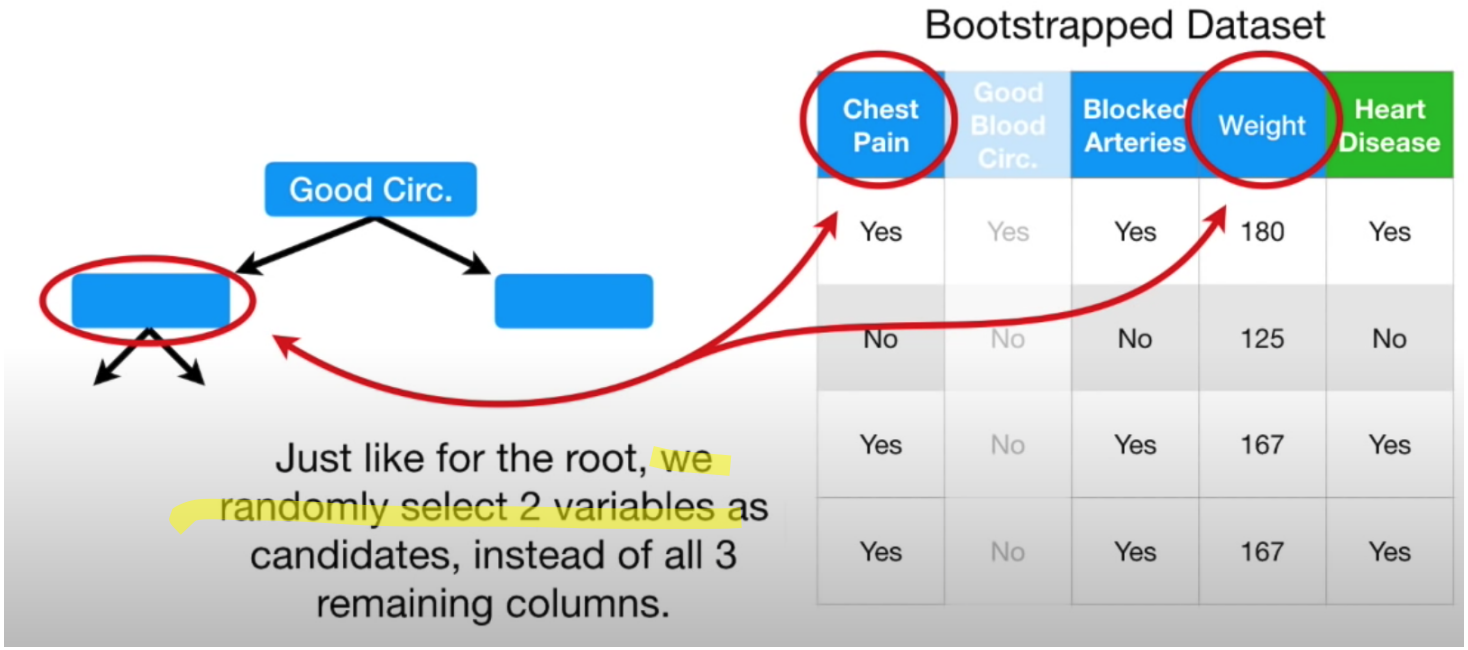
the root node.	No	Yes	167	Yes

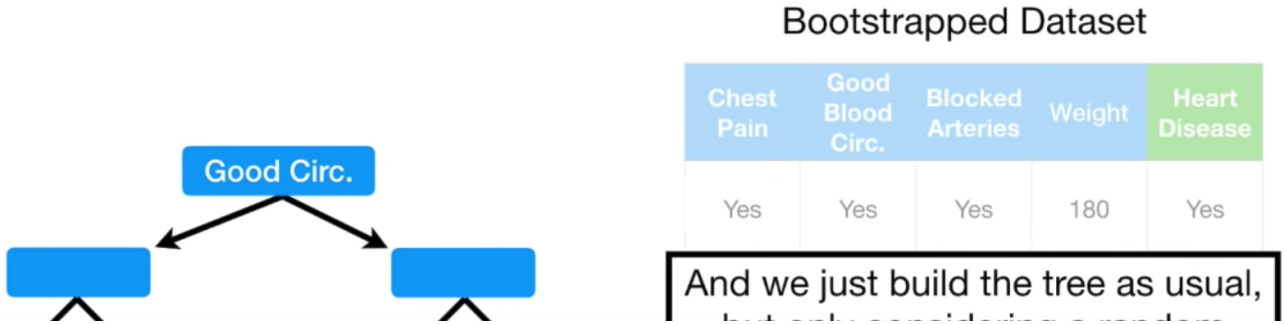
Just for the sake of the example, assume that **Good Blood Circulation** did the best job separating the samples.

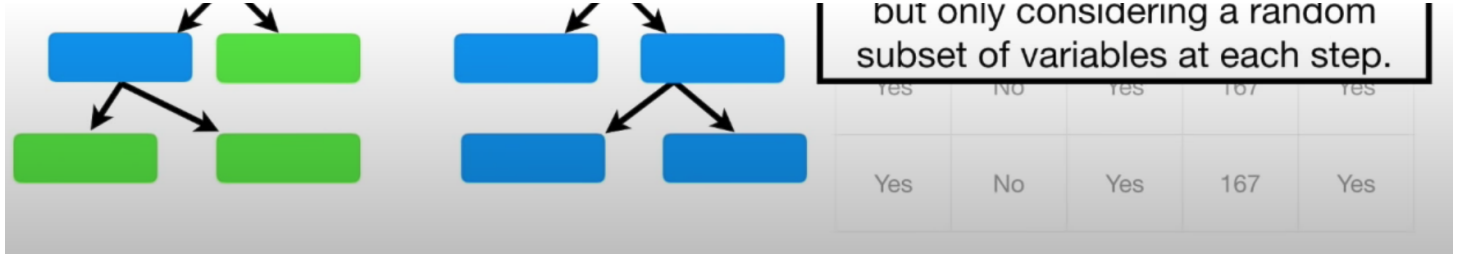
Bootstrapped Dataset

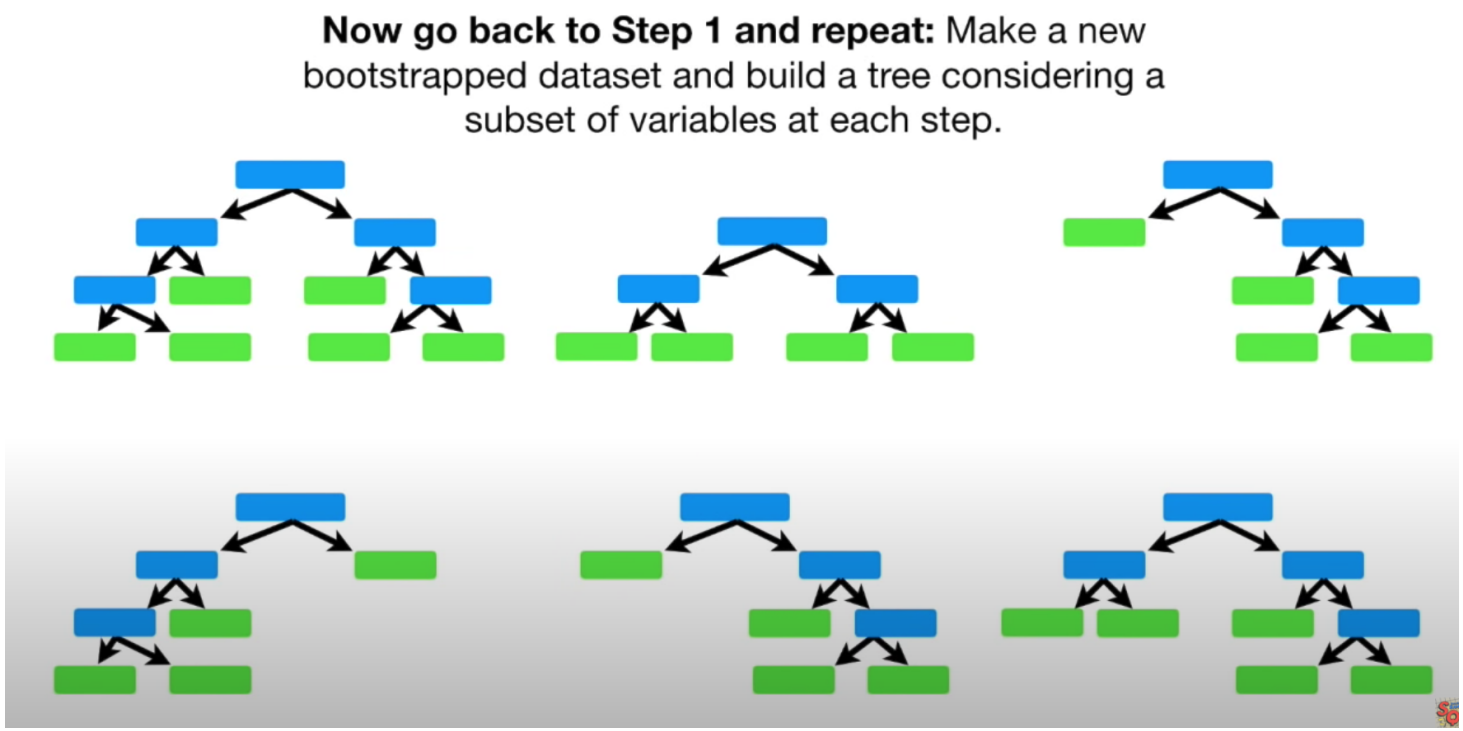
Chest Pain	[illegible] Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	[illegible]	Yes	180	Yes
No	[illegible]	No	125	No
Yes	[illegible]	Yes	167	Yes

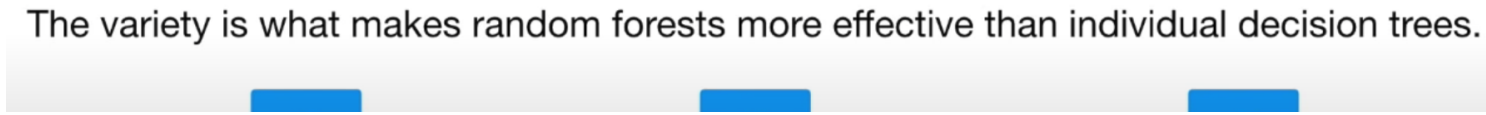
Yes		Yes	167	Yes

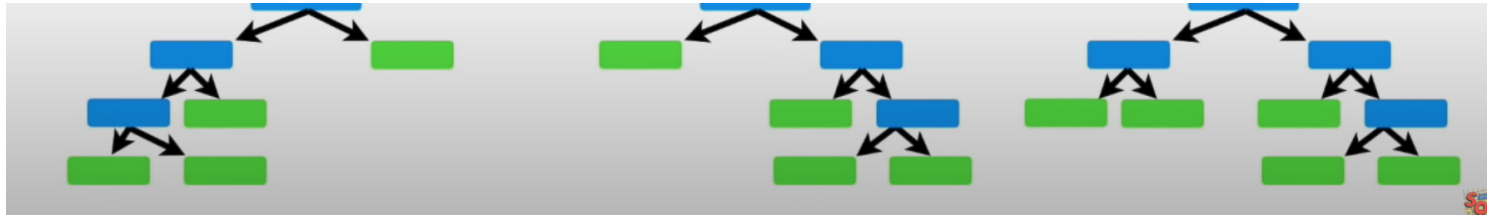





but only considering a random
subset of variables at each step.
Yes No Yes 167 Yes


Now go back to Step 1 and repeat: Make a new
bootstrapped dataset and build a tree considering a
subset of variables at each step.

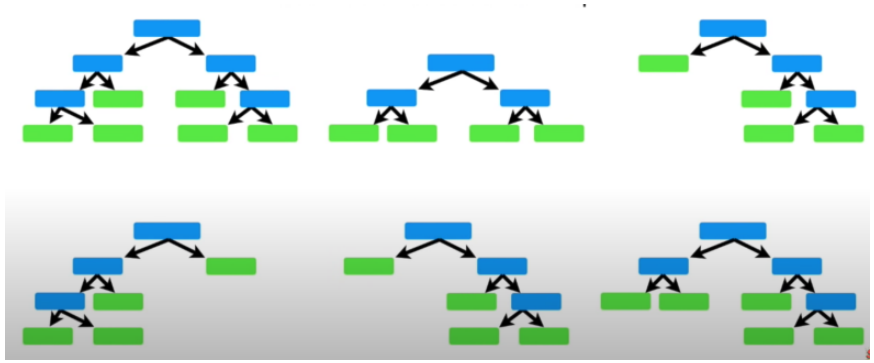

The variety is what makes random forests more effective than individual decision trees.



Let's use RF to predict the class of the (unseen) data.

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	

…we've got all the measurements…



After running the data down all of the trees in the random forest, we see which option received more votes.

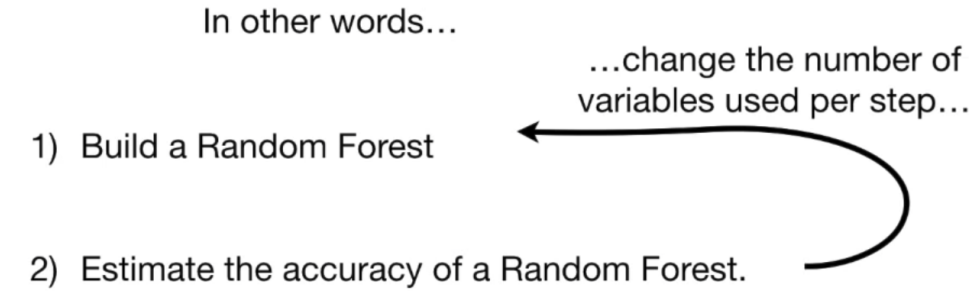
Heart Disease	
Yes	No
5	1

Remember when we built our first tree and we only used 2 variables (columns of data) to make a decision at each step?

Bootstrapped Dataset

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

We test with different settings and choose the most accurate ones.



Do this for a bunch of times and then choose the one that is most accurate.

Normally, the number we pick is the square root of n.