

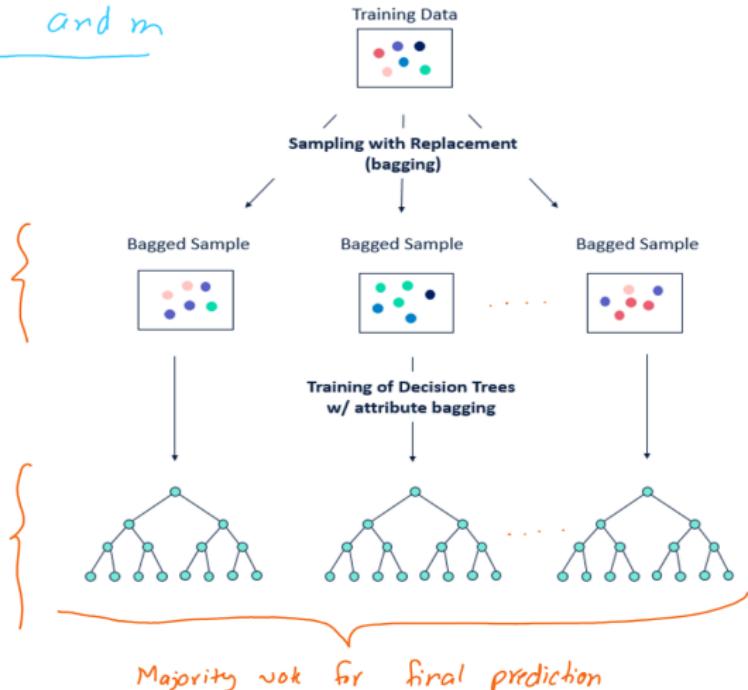
# Random Forest

# Big Picture of Random Forest

Hyper parameter:  $R$  and  $m$

$\checkmark$  % of  
the original  
data

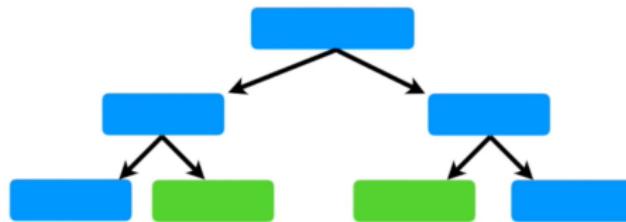
consider  
a random  $m$   
variable at  
each split



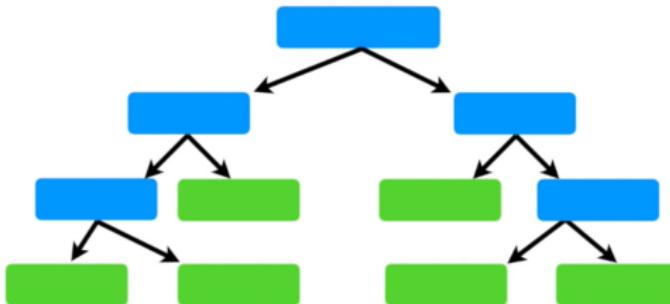
Random Forests are made out of decision trees



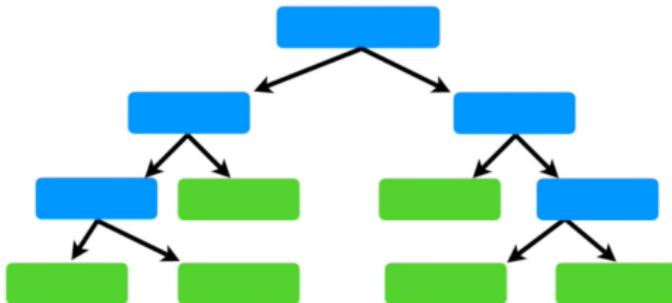
Random Forests are made out of decision trees



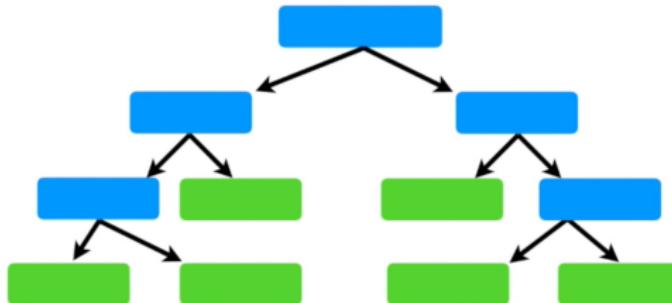
Random Forests are made out of decision trees



Decision Trees are easy to build, easy to use  
and easy to interpret...

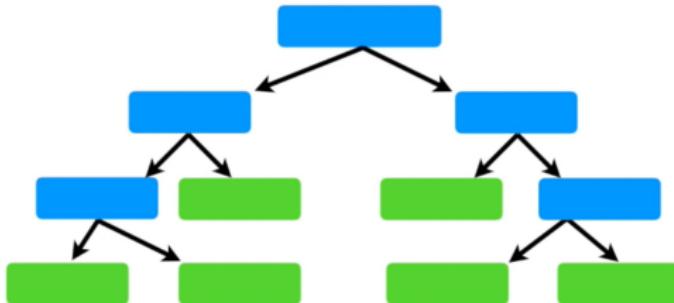


Decision Trees are easy to build, easy to use  
and easy to interpret...

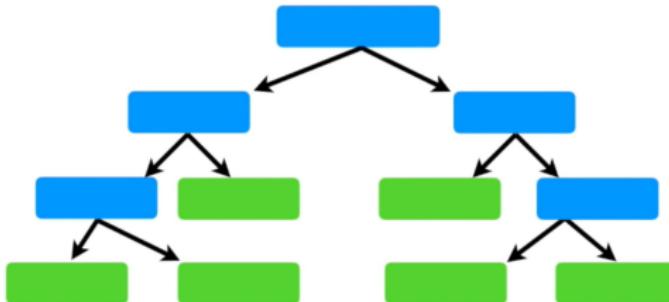


...but in practice they are not that awesome.

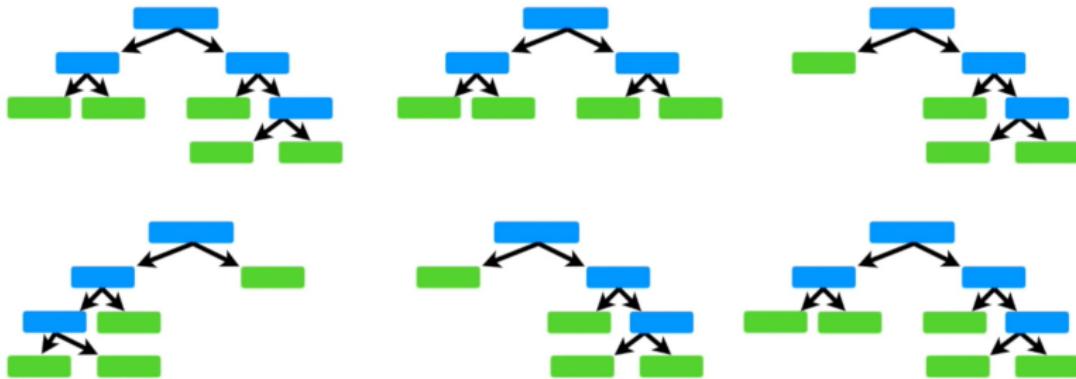
To quote from ***The Elements of Statistical Learning*** (aka The Bible of Machine Learning), “Trees have one aspect that prevents them from being the ideal tool for predictive learning, namely **inaccuracy**.”



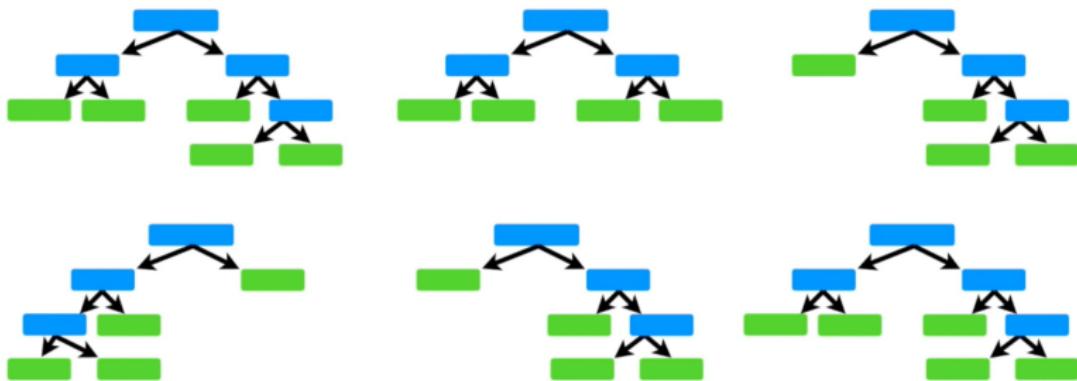
In other words, they work great with the data used to create them, but **they are not flexible when it comes to classifying new samples.**



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



So let's make a Random Forest!!!



**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

Imagine that these 4 samples are the entire dataset that we are going to build a tree from...



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
No	No	No	125	No

To create a bootstrapped dataset that is the same size as the original, we just randomly select samples from the original 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
No	No	No	125	No

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



This is the first  
sample that we  
randomly select...

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

...so it's the first sample in our 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
Yes	Yes	Yes	180	Yes

This is the second randomly selected sample from the original 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
Yes	Yes	Yes	180	Yes
No	No	No	125	No

...so it's the second sample in our 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
Yes	Yes	Yes	180	Yes
No	No	No	125	No

Here's the third  
randomly selected  
sample...



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



...so here it is in the  
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
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes

Here's the fourth randomly selected sample (**Note:** it's the same as the third)...



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



...and here it is.

We've created 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
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.

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.

Bootstrapped Dataset

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

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.

Bootstrapped Dataset

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

**NOTE:** We'll talk more about how to determine the optimal number of variables to consider later...

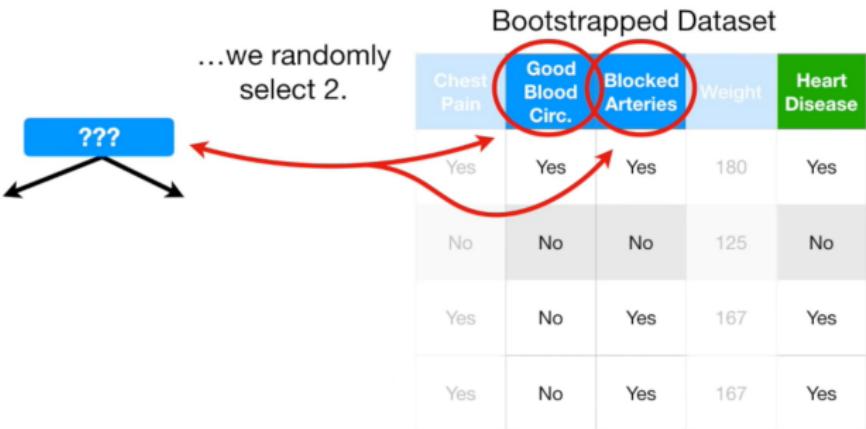
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

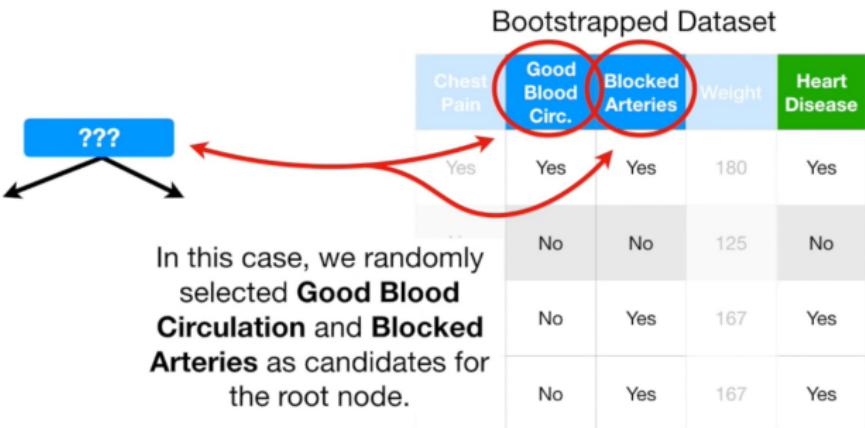
Thus, instead of  
considering all 4 variables  
to figure out how to split  
the root node...



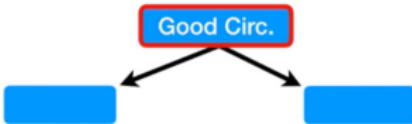
Bootstrapped Dataset

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





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

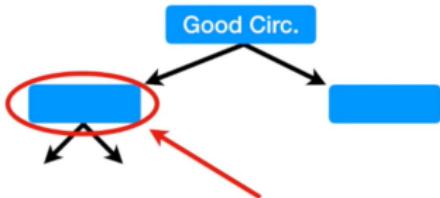


Bootstrapped Dataset

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

Bootstrapped Dataset

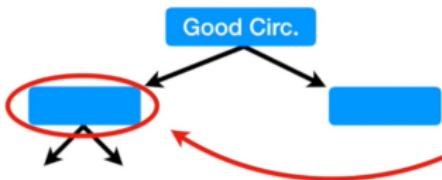
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes



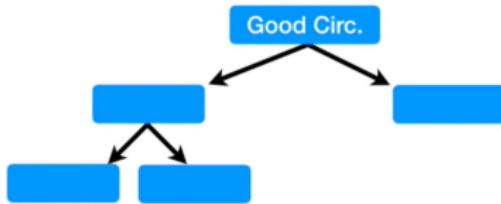
Now we need to figure out how to split samples at this node.

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



Just like for the root, we randomly select 2 variables as candidates, instead of all 4 columns.

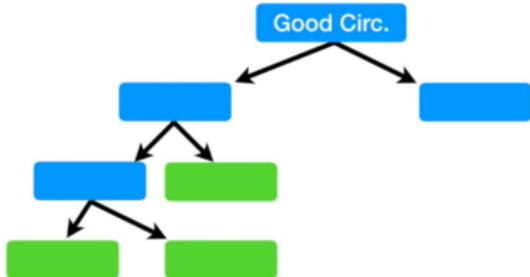


Bootstrapped Dataset

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes

And we just build the tree as usual,  
but only considering a random  
subset of variables at each step.

Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes



Bootstrapped Dataset

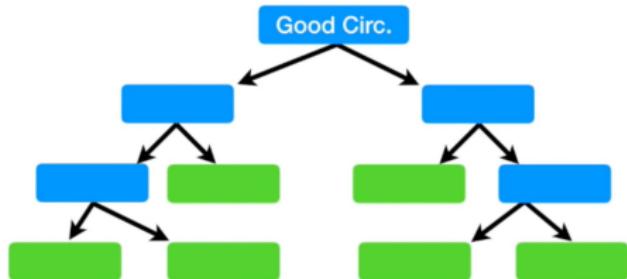
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes

And we just build the tree as usual,  
but only considering a random  
subset of variables at each step.

Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

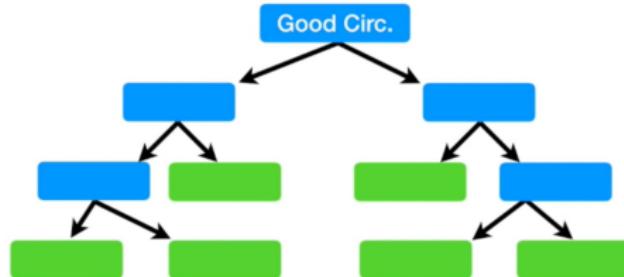
Bootstrapped Dataset

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



We built a tree...

### 1) Using a bootstrapped dataset

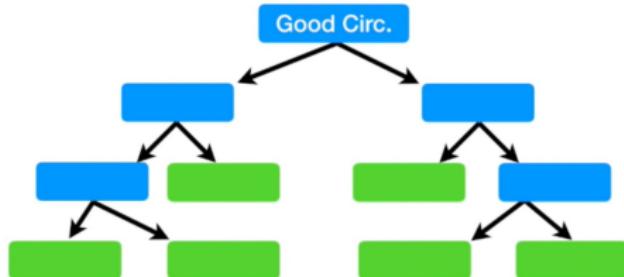


Bootstrapped Dataset

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 built a tree...

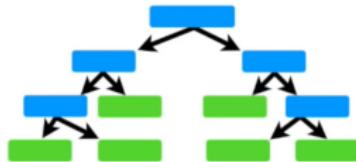
- 1) Using a bootstrapped dataset
- 2) Only considering a random a subset of variables at each step.



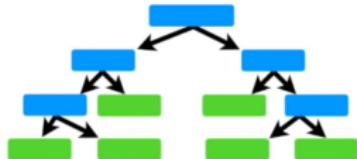
Bootstrapped Dataset

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

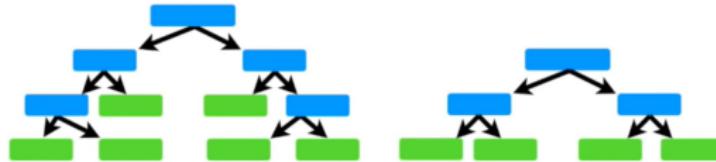
Here's the tree we just made...



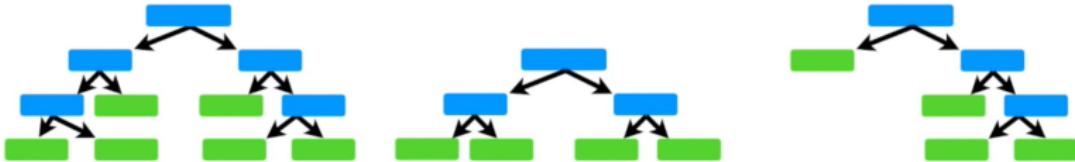
**Now go back to Step 1 and repeat:** Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.



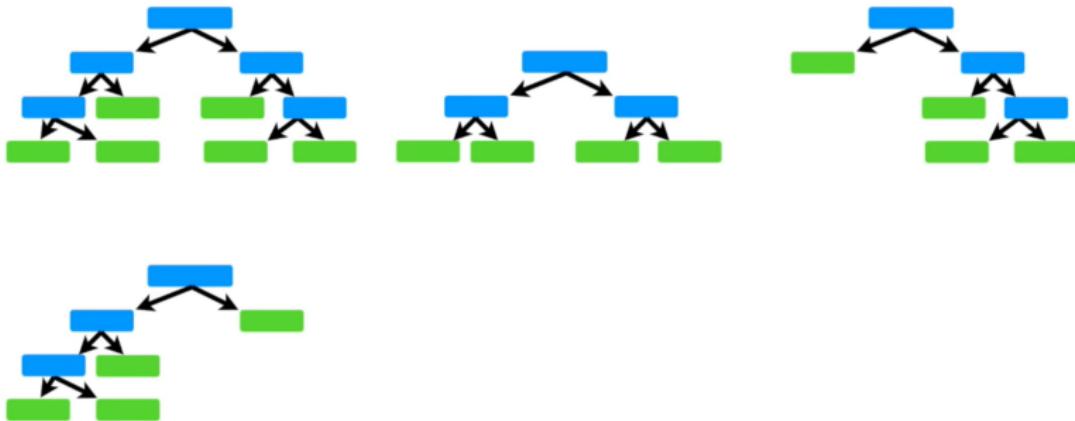
**Now go back to Step 1 and repeat:** Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.



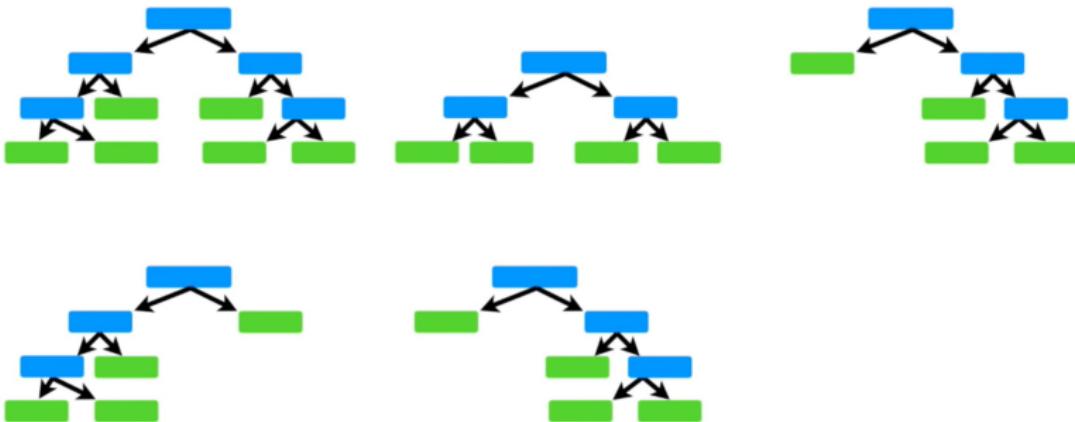
**Now go back to Step 1 and repeat:** Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.



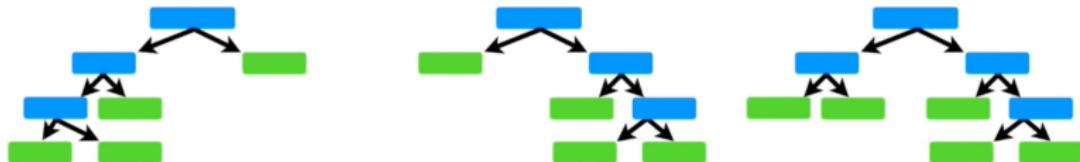
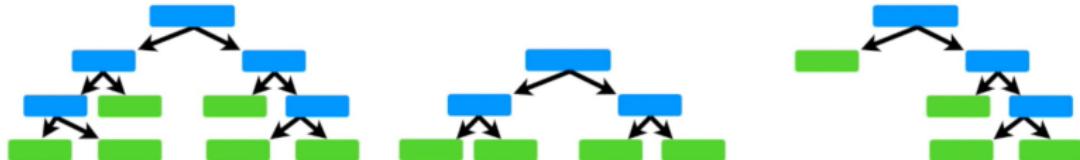
**Now go back to Step 1 and repeat:** Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.



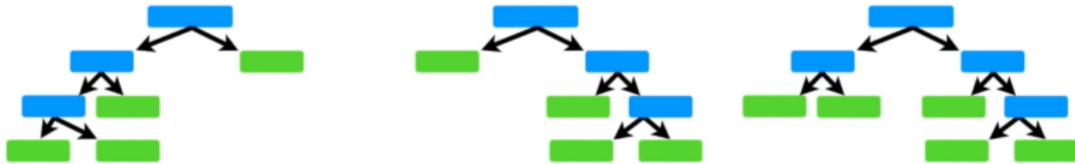
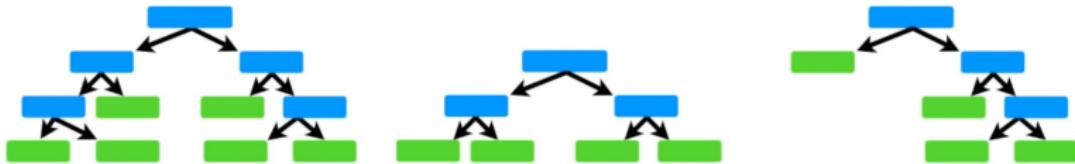
**Now go back to Step 1 and repeat:** Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.



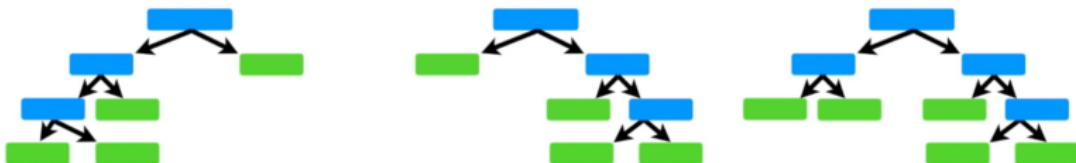
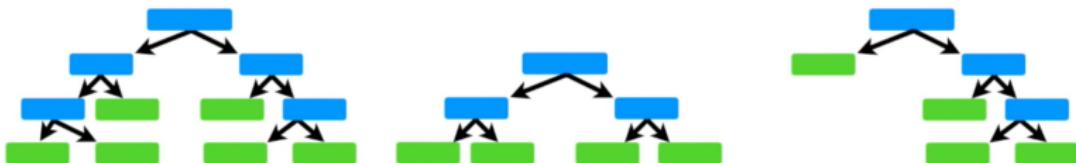
**Now go back to Step 1 and repeat:** Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.



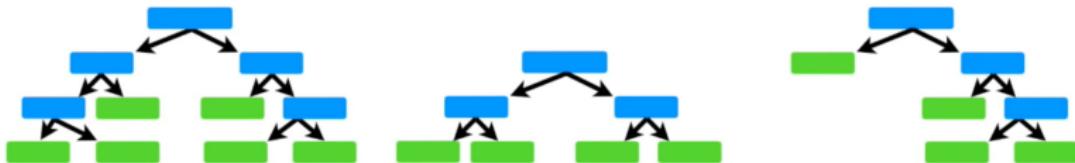
Ideally, you'd do this 100's of times, but we only have  
space to show 6... but you get the idea.



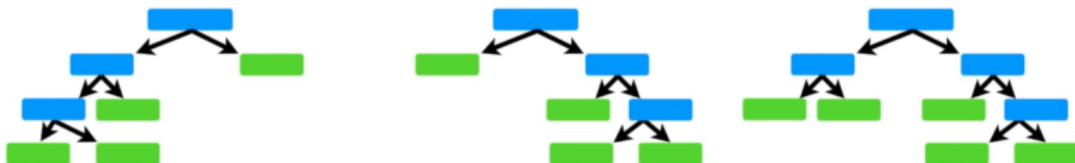
Using a bootstrapped sample and considering only a subset of the variables at each step results in a wide variety of trees.



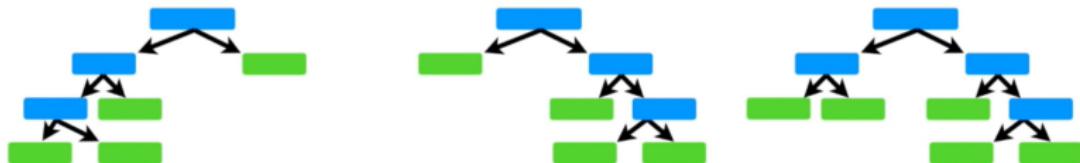
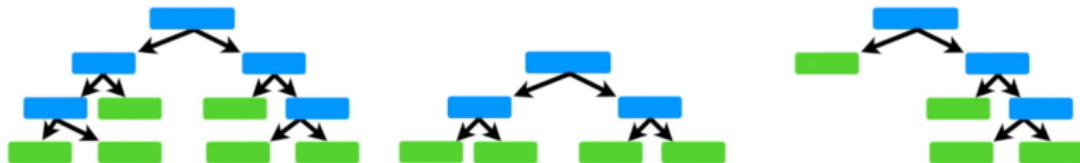
Using a bootstrapped sample and considering only a subset of the variables at each step results in a wide variety of trees.



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



Sweet!!! Now that we've created a random forest, how do we use it?



Well, first we get a new patient...

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

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

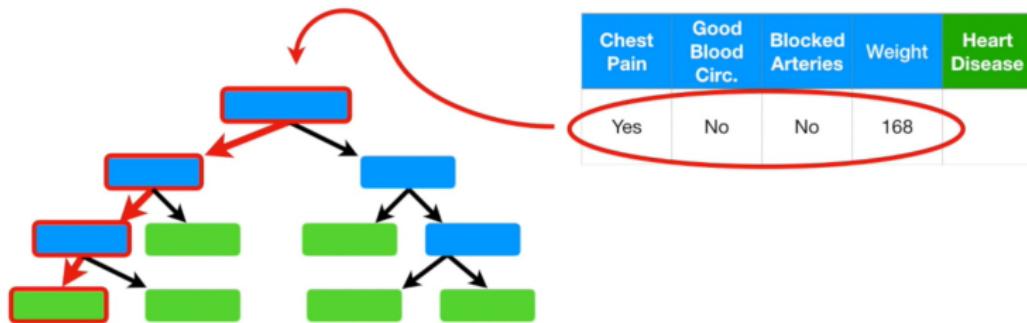
...we've got all the  
measurements...

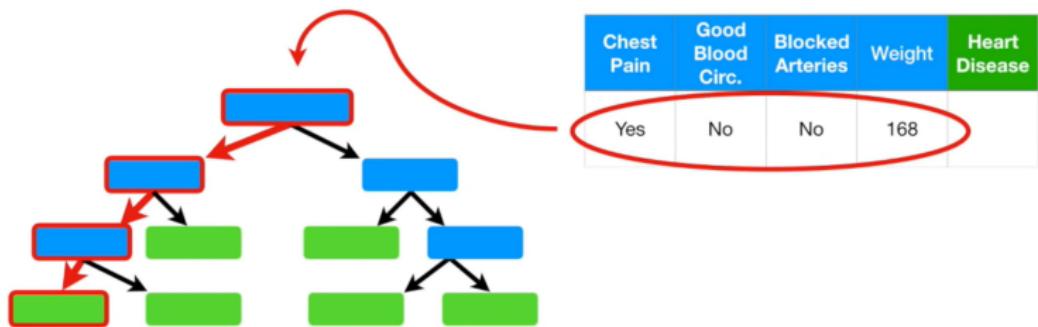
---

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	<input type="radio"/>

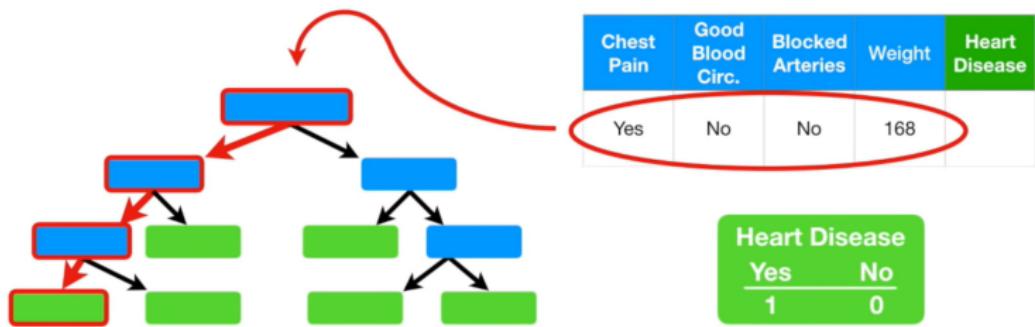
...and now we want to know if they have heart disease or not.

So we take the data  
and run it down the  
first tree that we  
made...





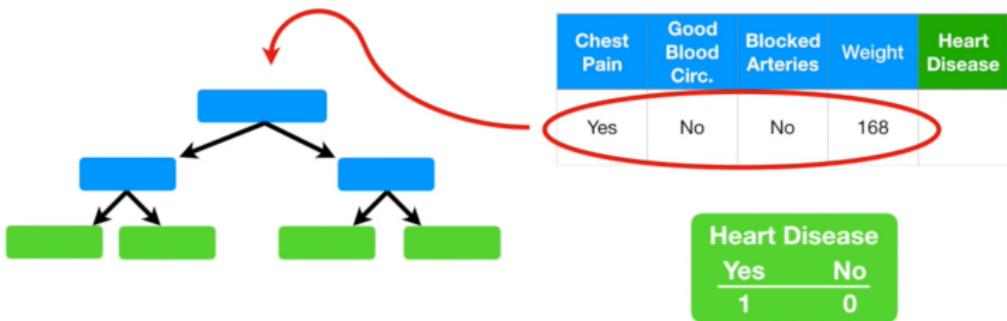
The first tree says  
“Yes”...



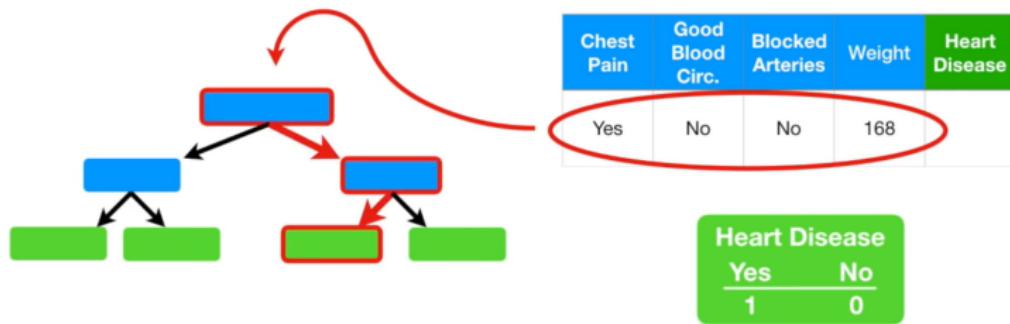
The first tree says  
“Yes”...

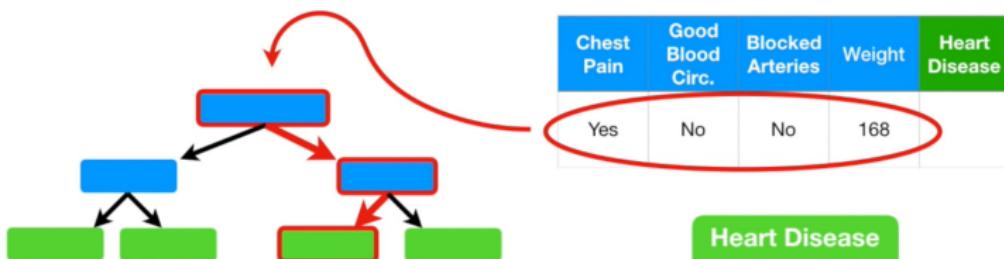
...and we keep track  
of that here.

Now we run the data  
down the second tree  
that we made...

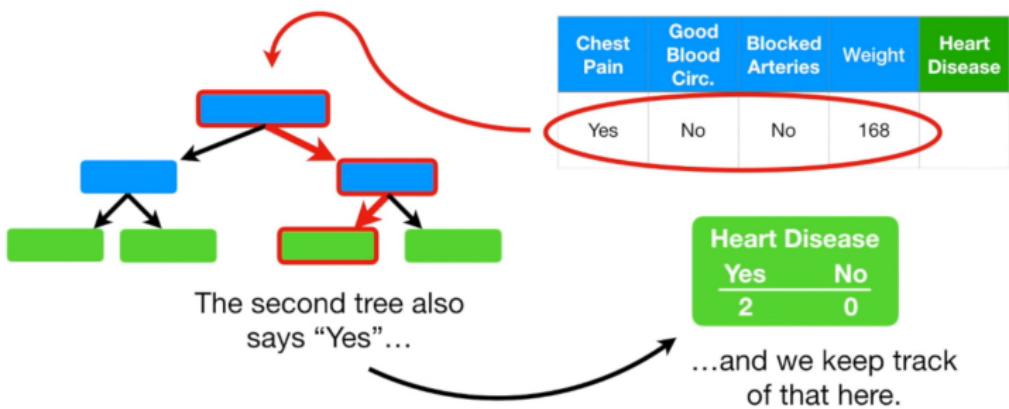


Now we run the data  
down the second tree  
that we made...

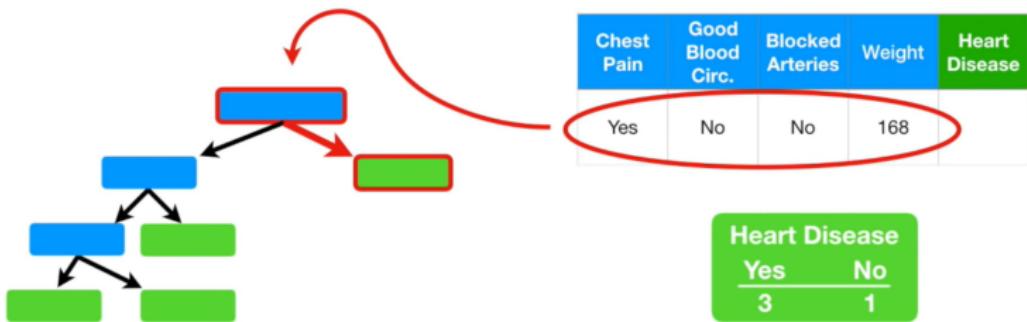




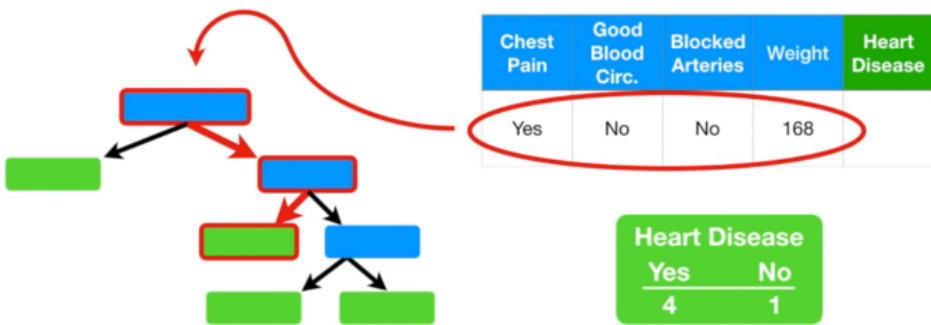
The second tree also  
says “Yes”...



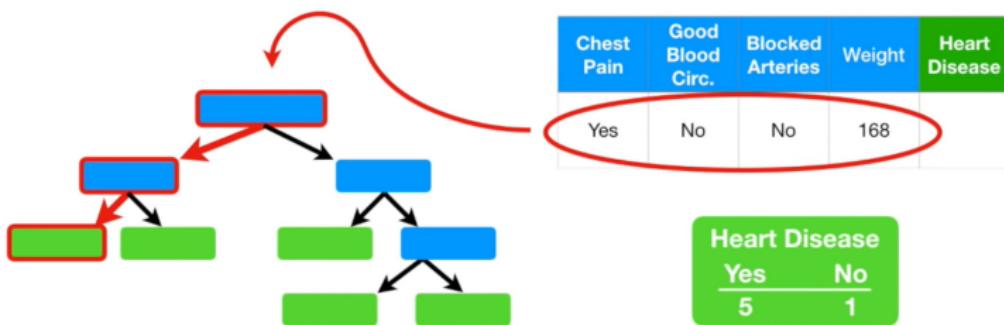
Then we repeat for all  
the trees that we  
made...



Then we repeat for all  
the trees that we  
made...



Then we repeat for all  
the trees that we  
made...



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

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

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	<b>YES</b>

In this case, “**Yes**” received the most votes, so we will conclude that this patient has heart disease.

Heart Disease	
Yes	No
5	1

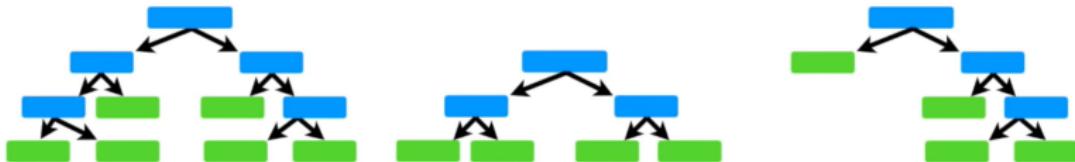
## **Terminology Alert!!!**

Bootstrapping the data plus using the aggregate to make a decision is called  
**“Bagging”**

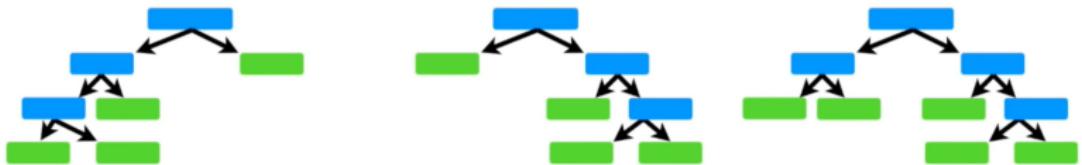
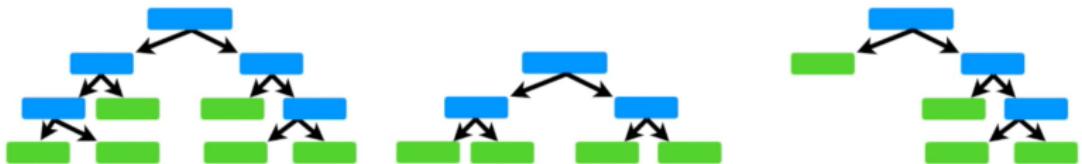
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	<b>YES</b>

Heart Disease	
Yes	No
5	1

OK, now we've seen how to create and  
use a random forest...



How do we know if it's any good?



## Out-of-bag errors

- ▶ Random Forest do not need a validation dataset!

Remember when we created the 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
No	No	No	125	No

We allowed duplicate entries in  
the 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
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

As a result, this entry was not included in the 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
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

Typically, about 1/3 of the original data does not end up in the 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
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

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

Here is the entry that didn't end up in the bootstrapped dataset..



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

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

(psst... if the original dataset were larger, we'd have more than just 1 entry over here...)



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

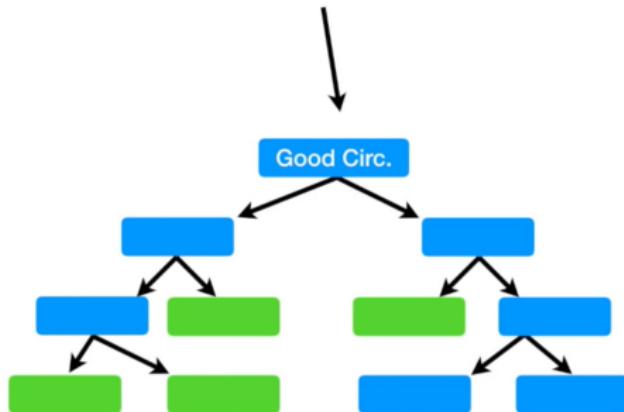
Original Dataset

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

This is called the  
**“Out-Of-Bag Dataset”**

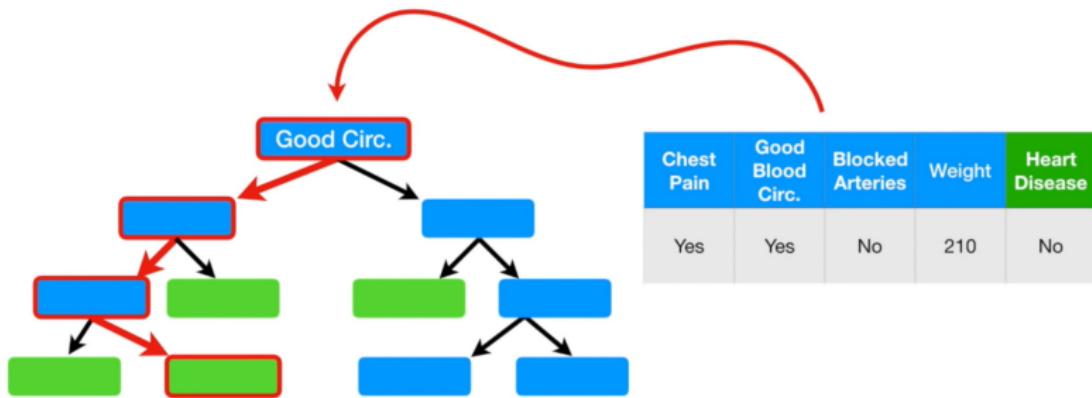
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	No	210	No

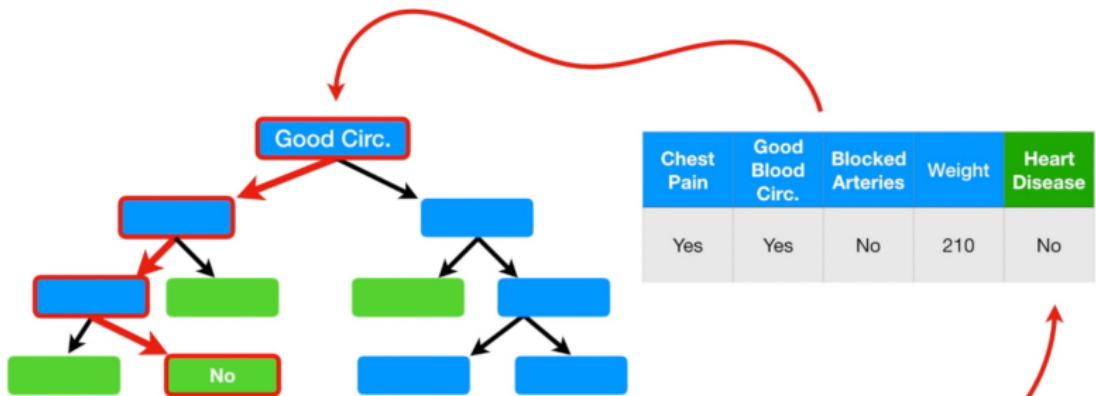
Since the Out-Of-Bag data was  
not used to create this tree...



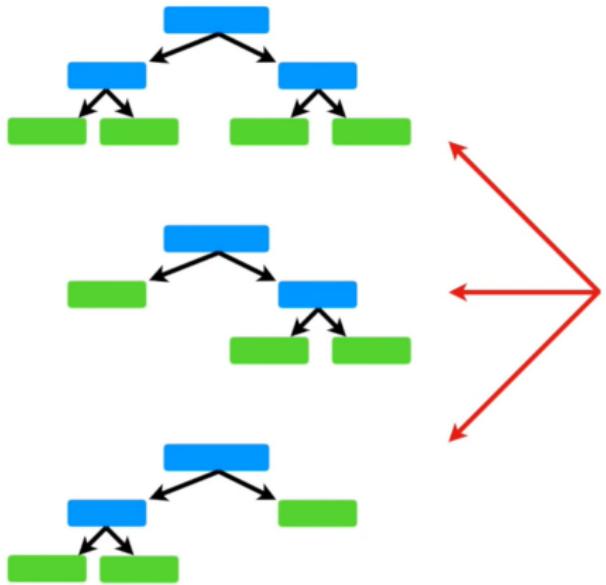
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	No	210	No

...we can run it through and see if it correctly classifies the sample as "No Heart Disease"



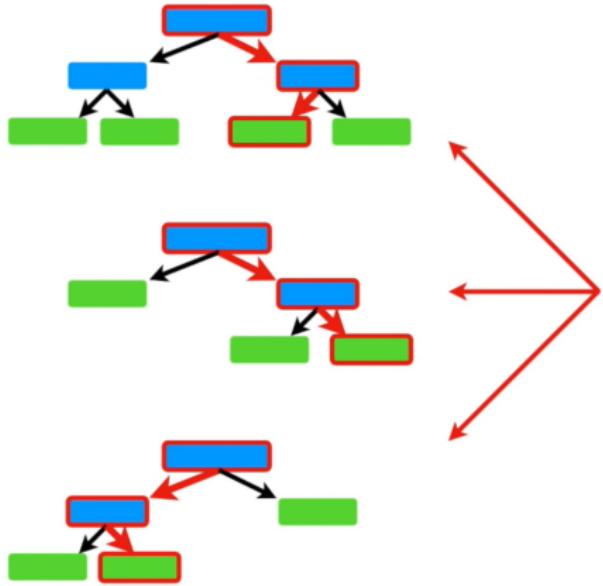


In this case, the tree correctly labels the Out-of-Bag sample “**No**.”



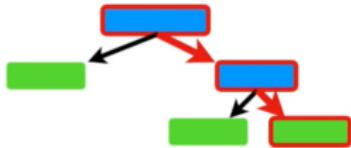
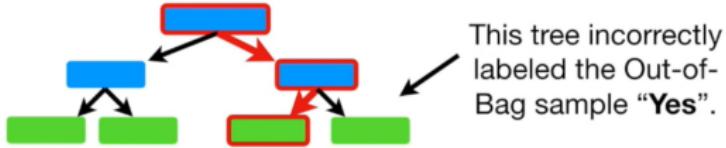
Then we run this Out-Of-Bag sample through all of the other trees that were built without it...

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

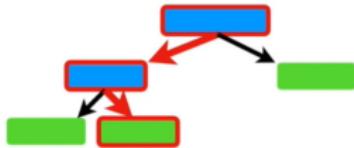


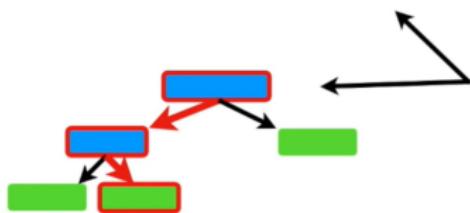
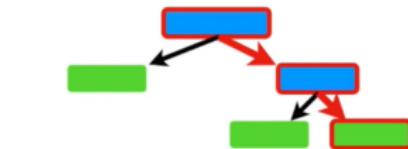
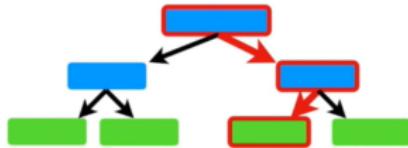
Then we run this Out-Of-Bag sample through all of the other trees that were built without it...

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



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





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

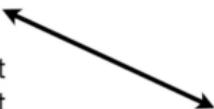
These trees correctly labeled the Out-of-Bag sample “No”.

**Classification of the  
Out-Of-Bag sample**

Yes	No
-----	----

1	3
---	---

Since the label with the most votes wins, it is the label that we assign this Out-of-Bag sample.



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

Classification of the Out-Of-Bag sample	
Yes	No
1	3

Since the label with the most votes wins, it is the label that we assign this Out-of-Bag sample.

In this case, the Out-of-Bag sample is correctly labeled by the Random Forest.

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

**Classification of the  
Out-Of-Bag sample**

Yes	No
1	3

We then do the same thing for all of the other Out-Of-Bag samples for all of the trees.

**Classification of the Out-Of-Bag sample**

Yes	No
-----	----

1	3
---	---

**Classification of the Out-Of-Bag sample**

Yes	No
-----	----

4	0
---	---

This Out-of-Bag sample was also correctly labeled...

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes

**Classification of the Out-Of-Bag sample**

Yes	No
-----	----

---

1	3
---	---

**Classification of the Out-Of-Bag sample**

Yes	No
-----	----

---

4	0
---	---

**Classification of the Out-Of-Bag sample**

Yes	No
-----	----

---

3	1
---	---

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

This Out-of-Bag sample was  
incorrectly labeled...



**Classification of the  
Out-Of-Bag sample**

Yes	No
1	3

**Classification of the  
Out-Of-Bag sample**

Yes	No
4	0

**Classification of the  
Out-Of-Bag sample**

Yes	No
3	1

etc... etc... etc...

Ultimately, we can measure how accurate our random forest is by the proportion of Out-Of-Bag samples that were correctly classified by the Random Forest.

**Classification of the  
Out-Of-Bag sample**

Yes	No
-----	----

1	3
---	---

**Classification of the  
Out-Of-Bag sample**

Yes	No
-----	----

4	0
---	---

**Classification of the  
Out-Of-Bag sample**

Yes	No
-----	----

3	1
---	---

etc... etc... etc...

Ultimately, we can measure how accurate our random forest is by the proportion of Out-Of-Bag samples that were correctly classified by the Random Forest.

The proportion of Out-Of-Bag samples that were *incorrectly* classified is the “**Out-Of-Bag Error**”

OK, we now know how to:

- 1) Build a Random Forest
- 2) Use a Random Forest
- 3) Estimate the accuracy of a Random Forest.

OK, we now know how to:

- 1) Build a Random Forest
- 2) Use a Random Forest

However, now that  
we know how to do  
this...

- 3) Estimate the accuracy of a Random Forest.

However, now that  
we know how to do  
this...

OK, we now know how to:

1) Build a Random Forest

2) Use a Random Forest

3) Estimate the accuracy of a Random Forest.

...we can talk a little  
more about how to  
do this!



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

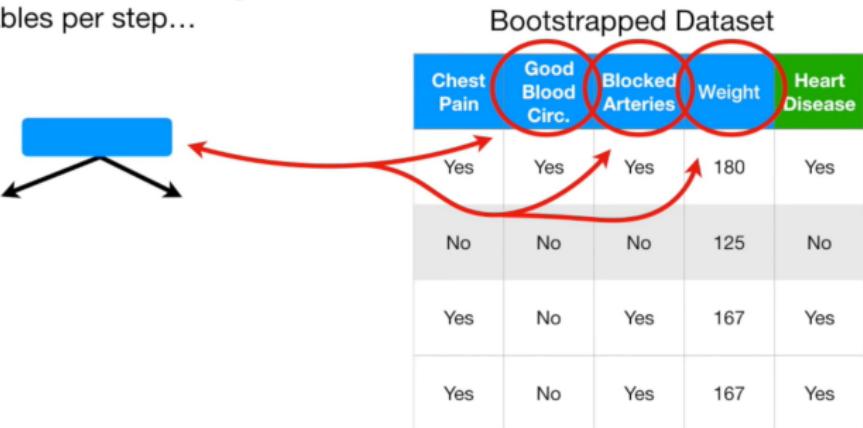
Now we can compare the Out-Of-Bag error for a random forest built using only 2 variables per step...



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

...to random forest built using 3 variables per step...



...to random forest built using 3 variables per step...



...and we test a bunch of different settings and choose the most accurate random forest.

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

In other words...

- 1) Build a Random Forest

In other words...

- 1) Build a Random Forest
- 2) Estimate the accuracy of a Random Forest.

In other words...

...change the number of  
variables used per step...

1) Build a Random Forest

2) Estimate the accuracy of a Random Forest.



In other words...

...change the number of  
variables used per step...

1) Build a Random Forest

2) Estimate the accuracy of a Random Forest.



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

In other words...

...change the number of  
variables used per step...

1) Build a Random Forest

2) Estimate the accuracy of a Random Forest.



Typically, we start by using the square of the number of variables and then try a few settings above and below that value.