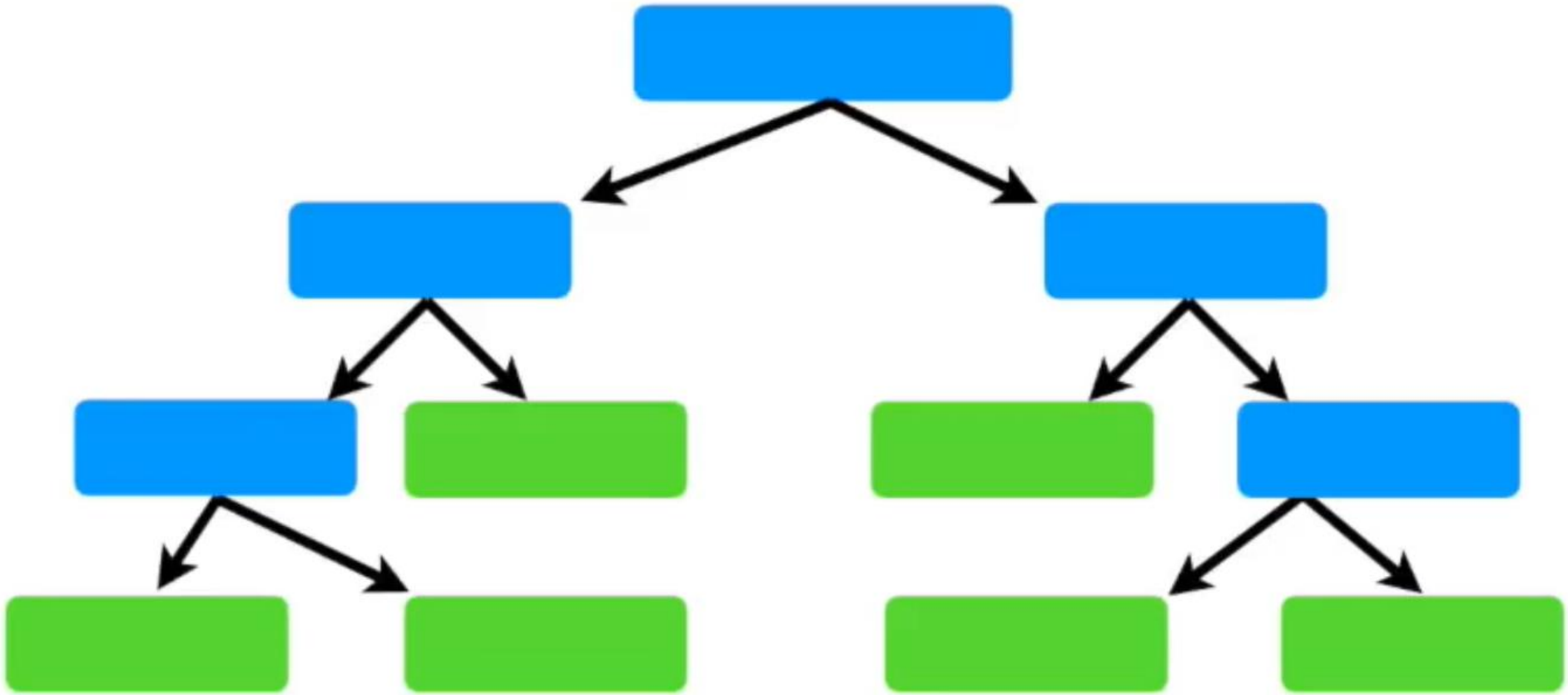
"Anticipare la crescita con le nuove competenze sui Big Data" Operazione Rif. PA 2023-19167/RER approvata con DGR n° 843 del 29 maggio 2023 e co-finanziata dal Fondo Sociale Europeo Plus 2021-2027 Regione Emilia-Romagna

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

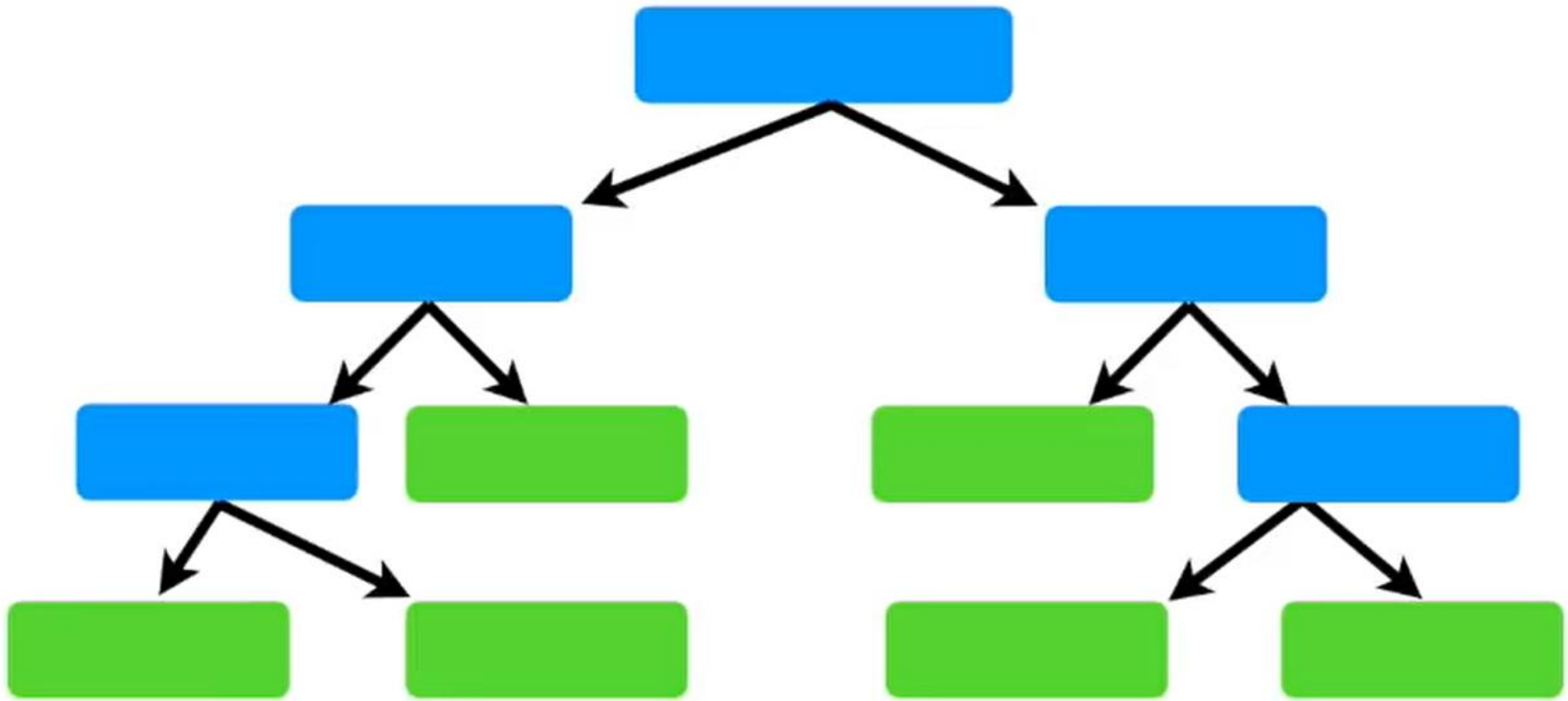
Operazione Rif. PA 2023-19167/RER/10/1, “ANTICIPARE LA CRESCITA CON LE NUOVE COMPETENZE SUI BIG DATA”, approvata dalla Regione Emilia-Romagna con DGR n° 843 del 29/05/2023 e co-finanziata dal Fondo Sociale Europeo Plus 2021-2027

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

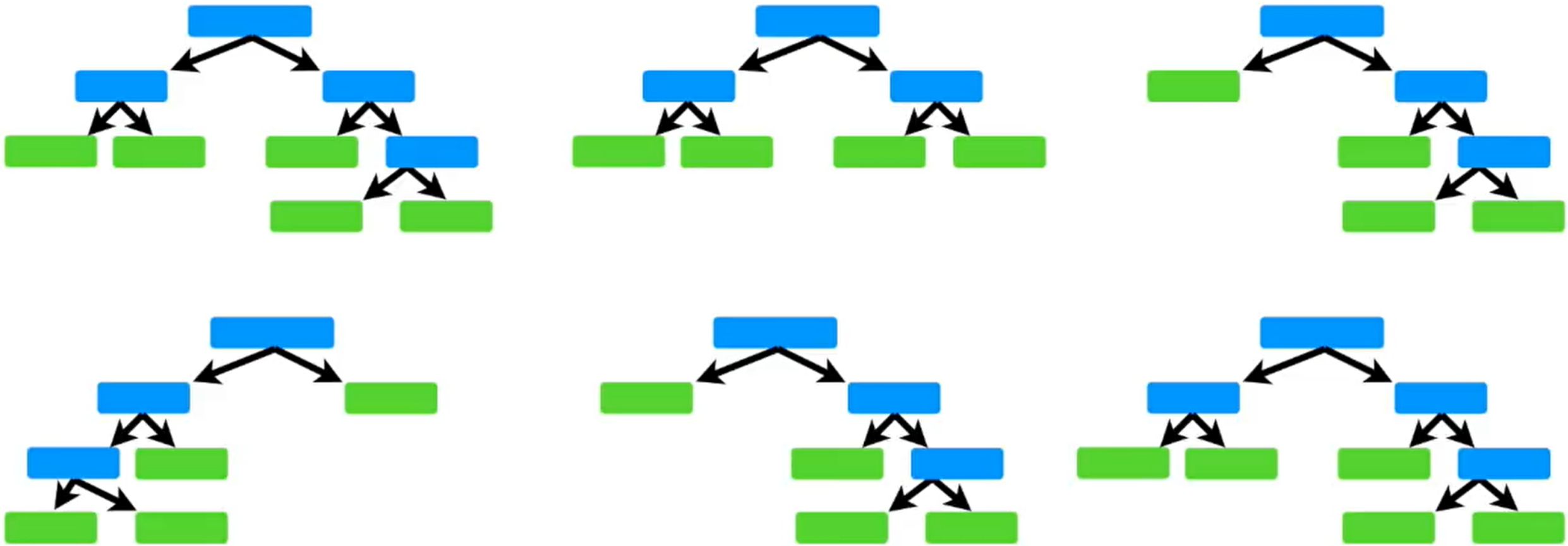


…but in practice they are not that awesome.

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.



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

…and here it is.

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.

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

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

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

Bootstrapped Dataset

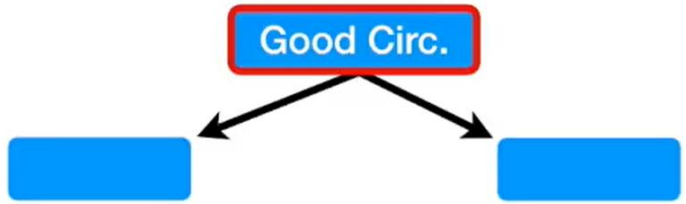
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes

Bootstrapped Dataset

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

In this case, we randomly selected **Good Blood Circulation** and **Blocked Arteries** as candidates for the root node.

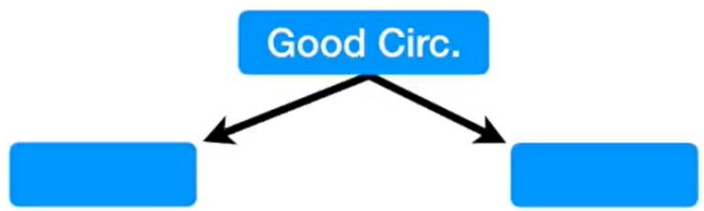
Just for the sake of the example, assume that **Good Blood Circulation** did the best job separating the samples.



Bootstrapped Dataset

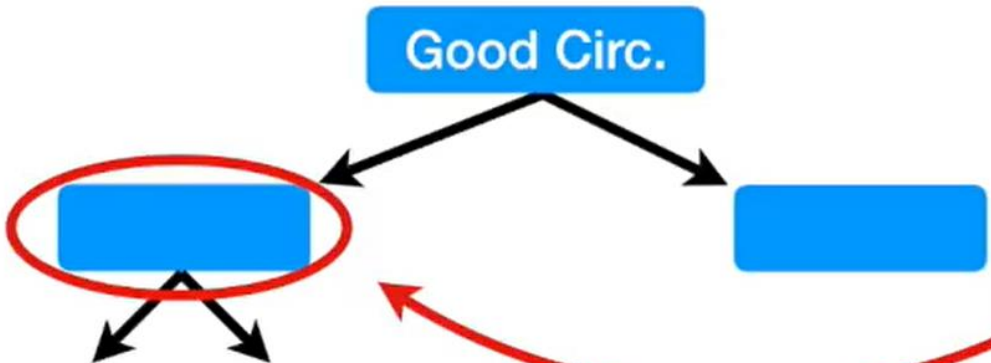
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No

Since we used **Good Blood Circulation**, I'm going to grey it out so that we focus on the remaining variables.



Bootstrapped Dataset

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



Just like for the root, we randomly select 2 variables as candidates, instead of all 3 remaining columns.

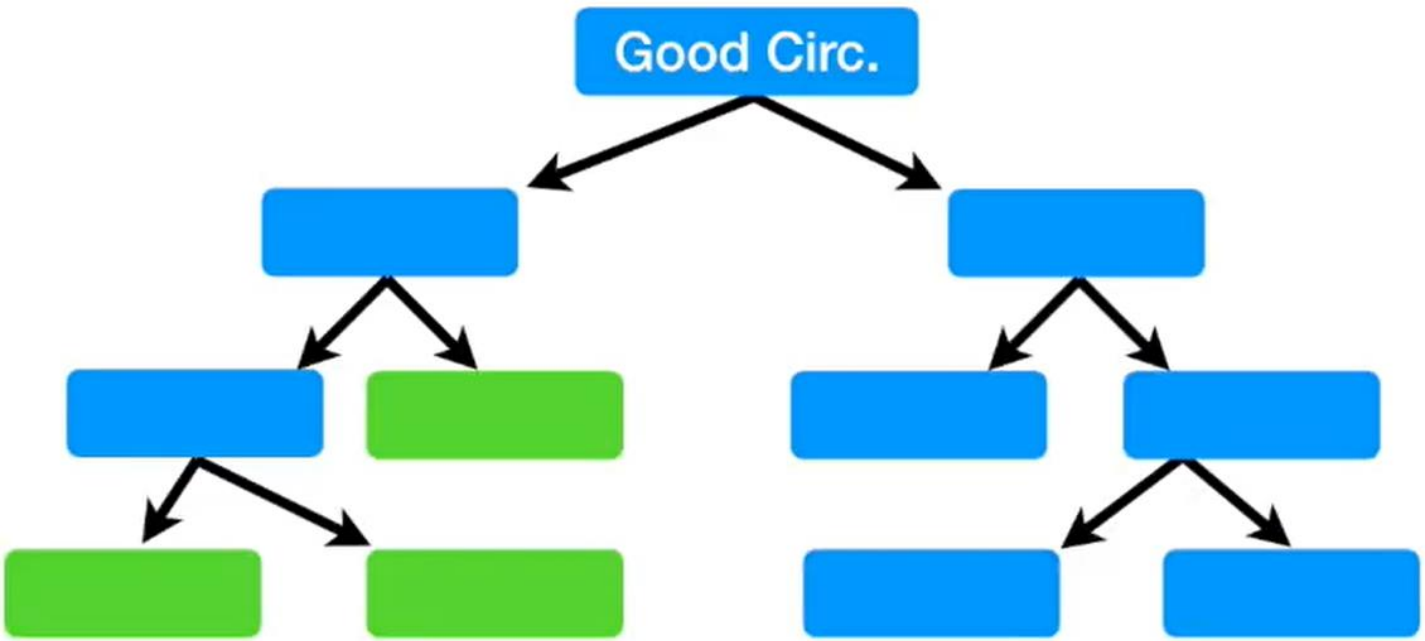
Bootstrapped Dataset

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.

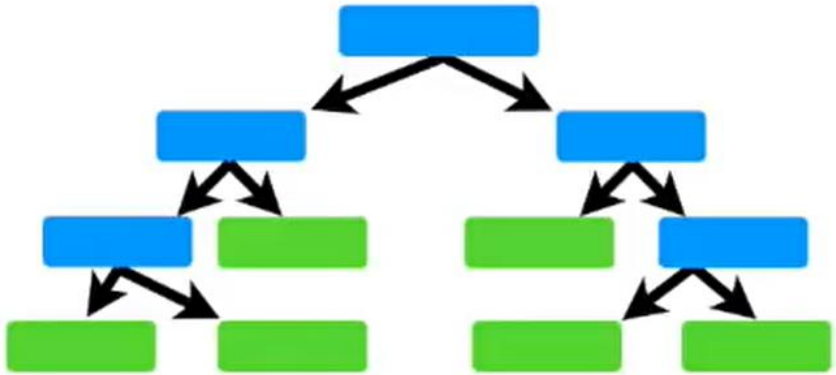
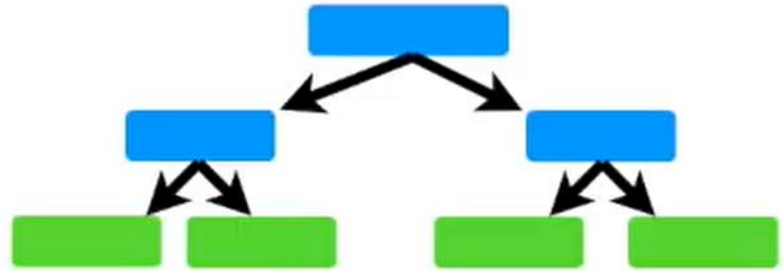
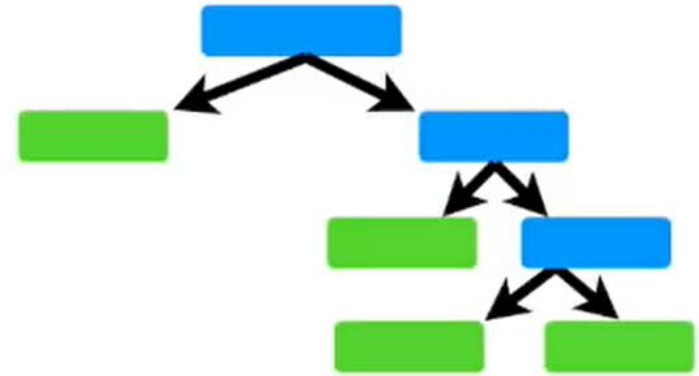
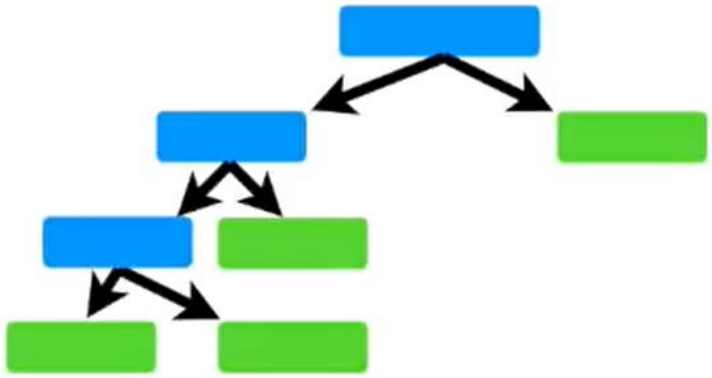
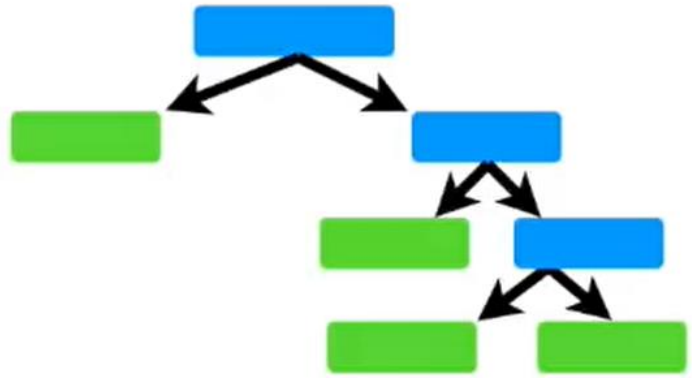
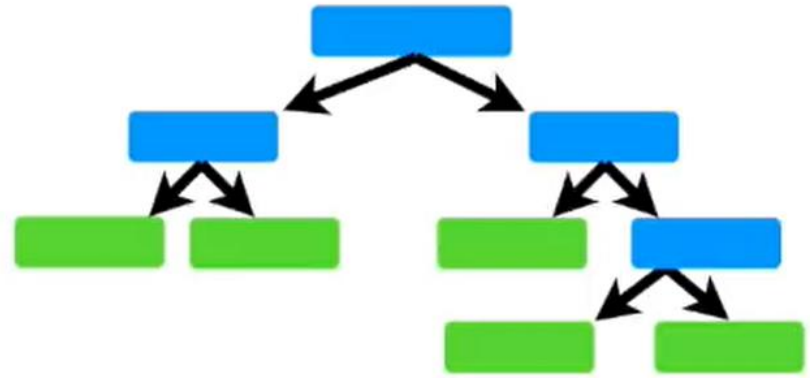


Bootstrapped Dataset

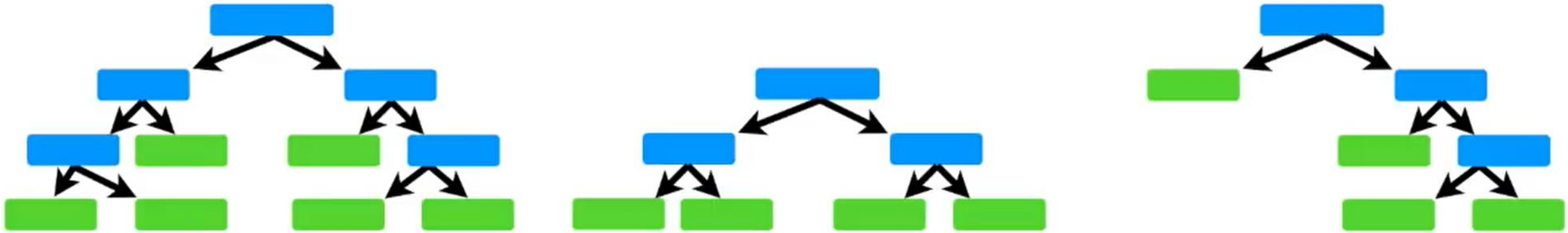
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
Yes	No	Yes	167	Yes

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

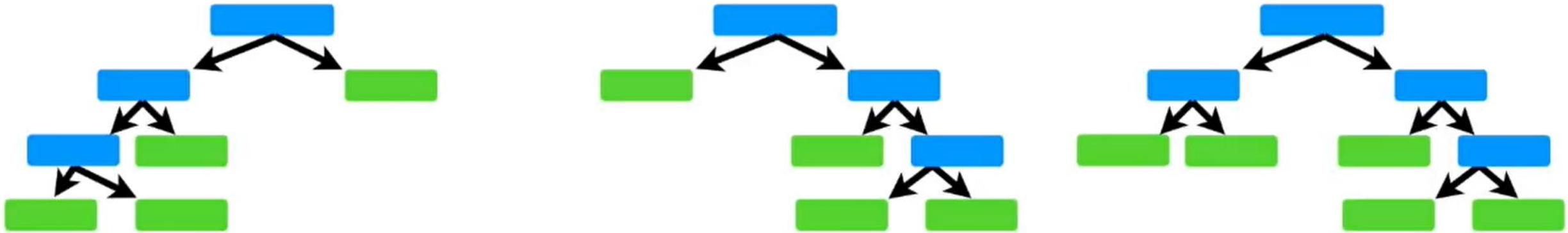
Now go back to Step 1 and repeat: Make a new bootstrapped dataset and build a tree considering a subset of variables at each step.

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.



OK, we now know how to:

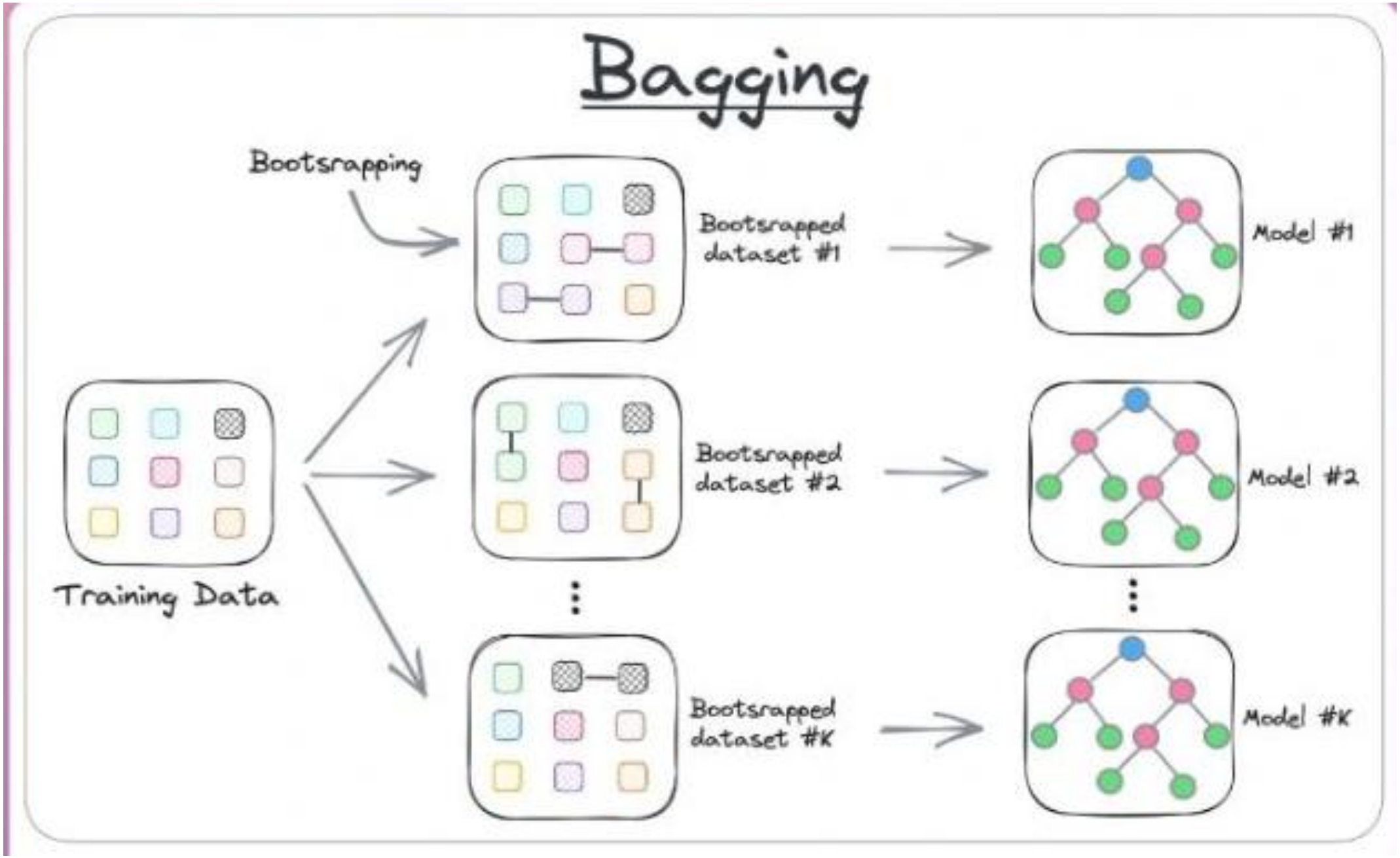
1) Build a Random Forest

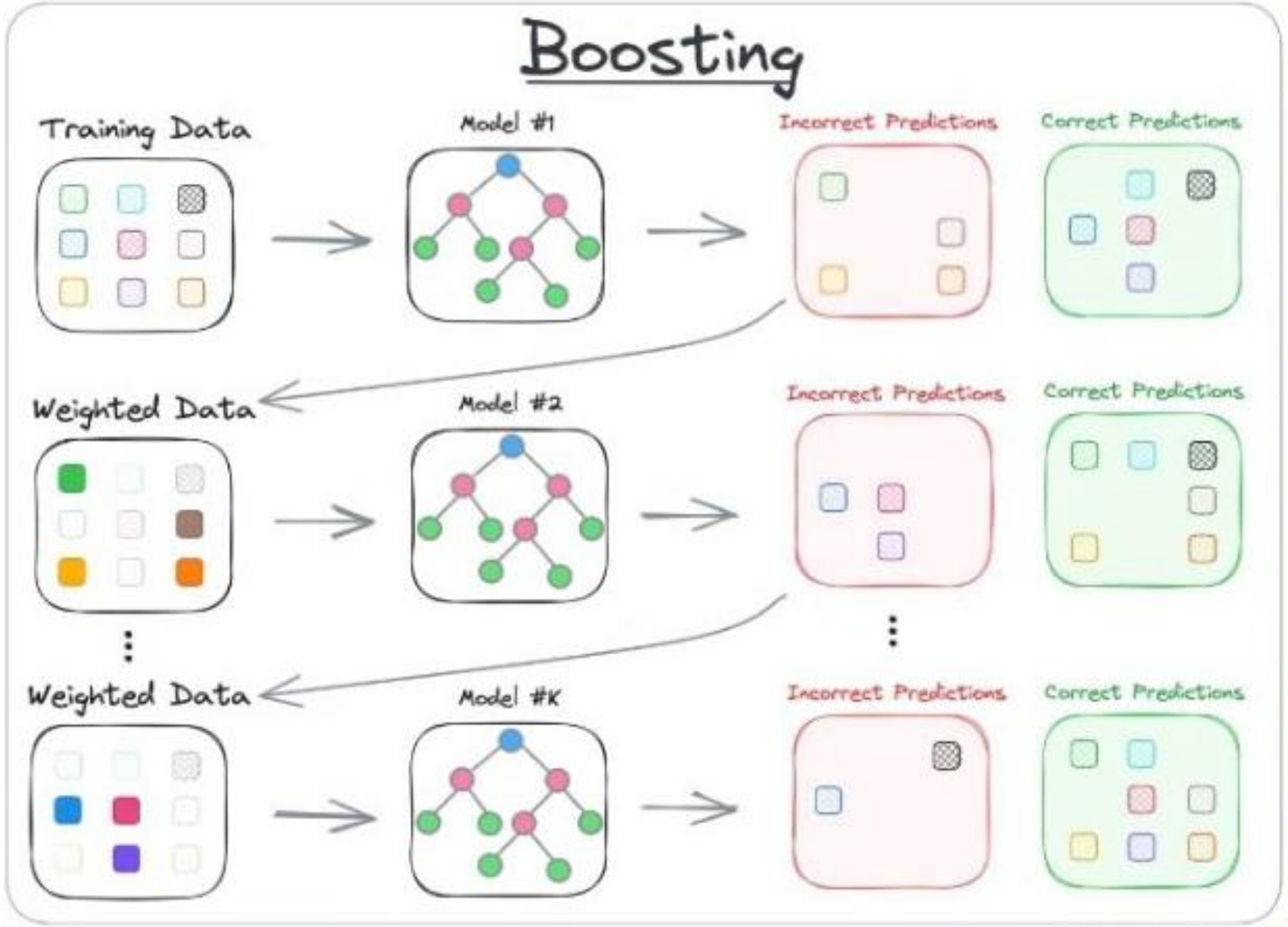
2) Use a Random Forest

However, now that we know how to do this…

3) Estimate the accuracy of a Random Forest.

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


Bagging
Bootsrapping
Bootsrapped dataset #1
Bootsrapped dataset #2
Bootsrapped dataset #K
Model #1
Model #2
Model #K
Training Data


Boosting
Training Data
Model #1
Incorrect Predictions
Correct Predictions
Weighted Data
Model #2
Incorrect Predictions
Correct Predictions
Weighted Data
Model #K
Incorrect Predictions
Correct Predictions

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.