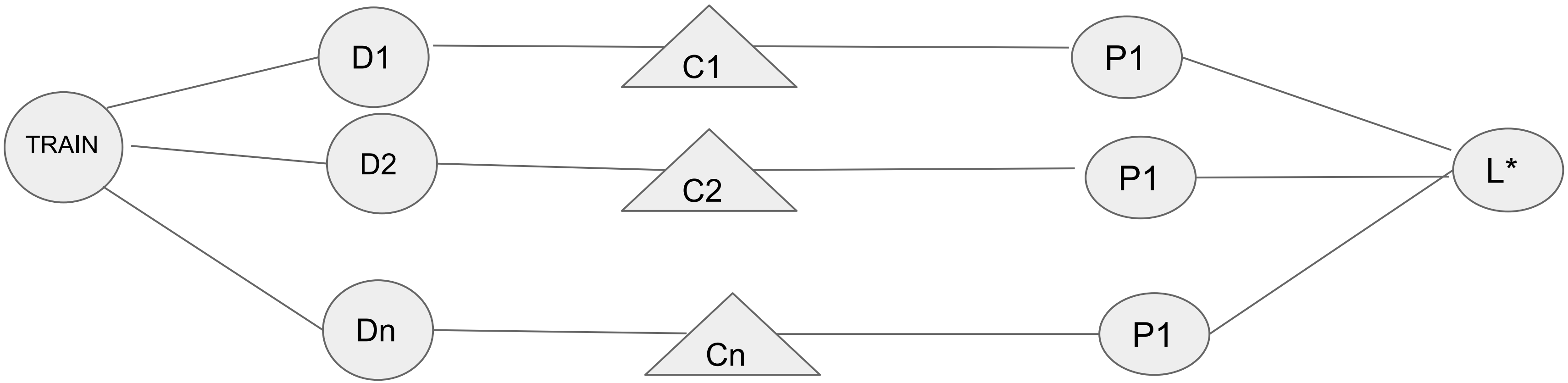
Ensemble Learning

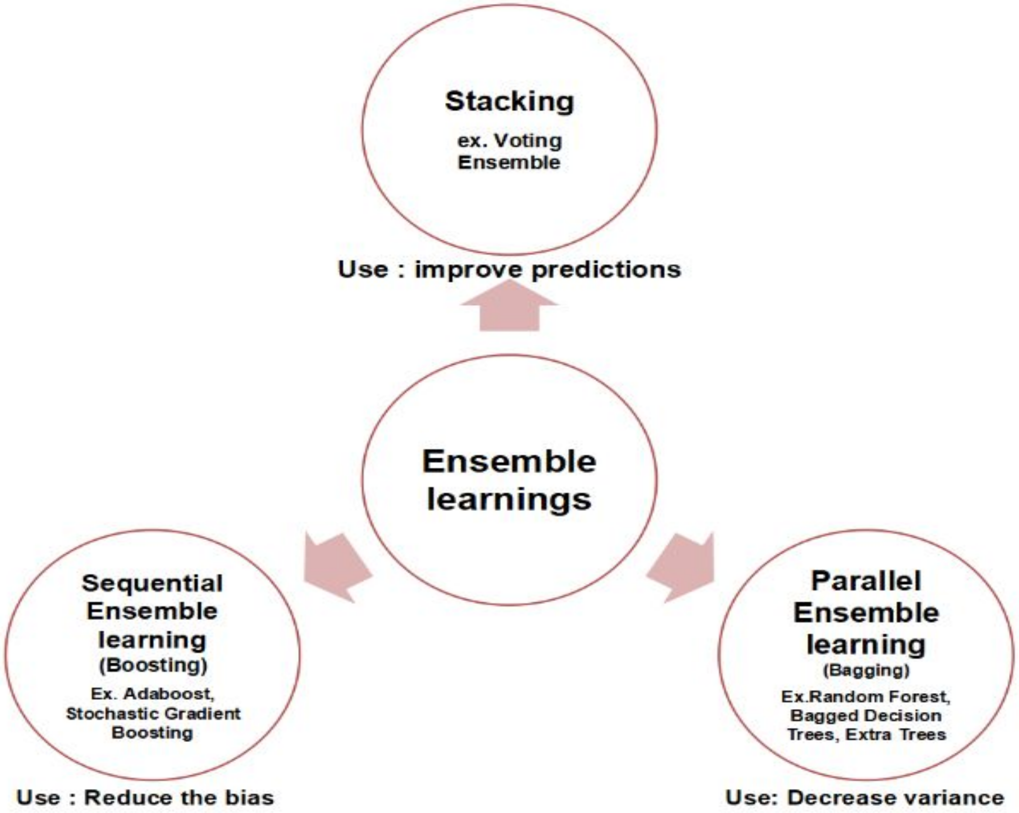
DEFINITION

- Meta-data algorithms which combines multiple machine learning techniques into one predictive model to achieve:

1. Decrease variance(bagging)
2. Decrease bias(boosting)
3. Improve prediction(stacking)



Ensemble methods are mainly divided into:



Why we need ensemble methods?

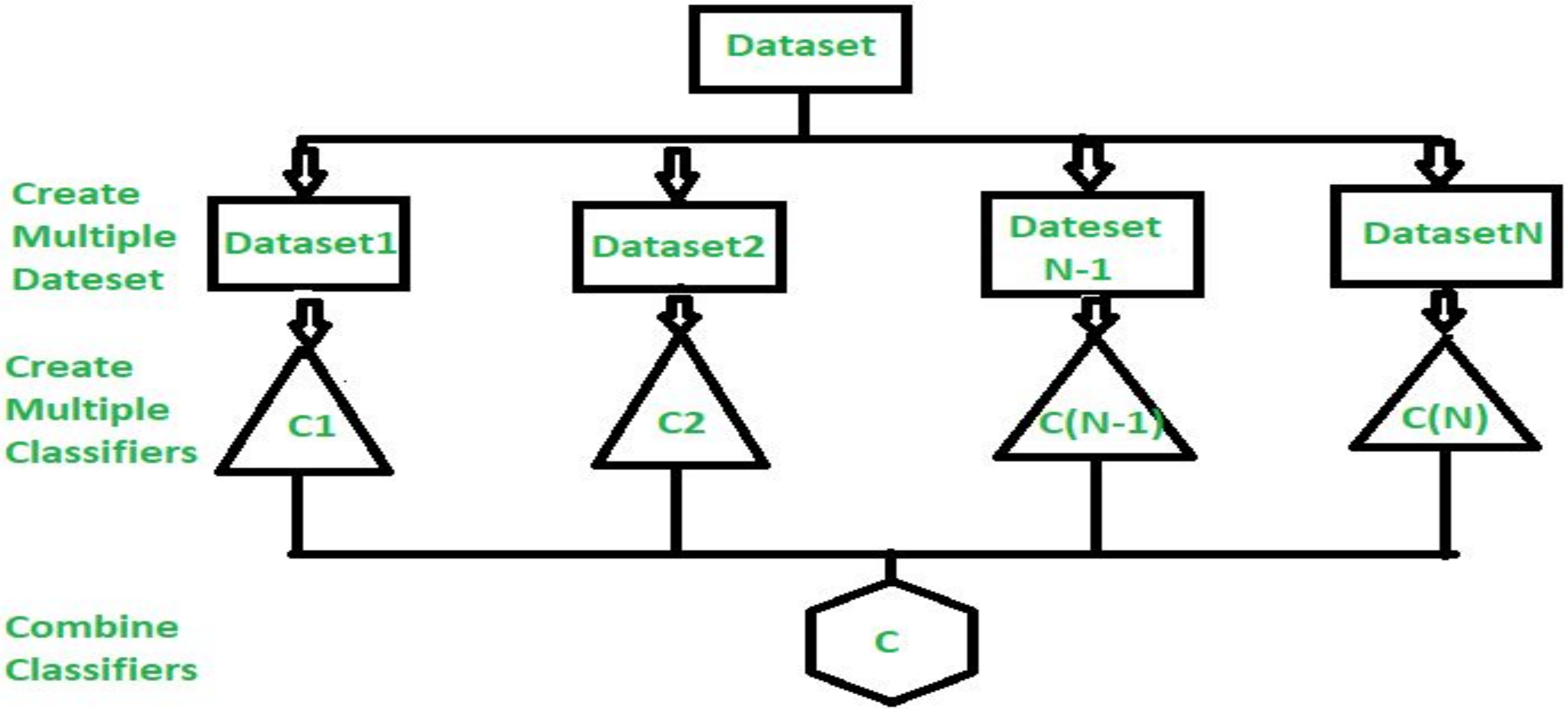
1. **Performance:** An ensemble can make better predictions and achieve better performance than any single contributing model.
2. **Robustness**: An ensemble reduces the spread or dispersion of the predictions and model performance.

Ensembles are used to achieve better predictive performance on a predictive modeling problem than a single predictive model. The way this is achieved can be understood as the model reducing the variance component of the prediction error by adding bias

Different ways to generate the ensemble classifier.

1. Manipulating the training data:We can manipulate the data sampling techniques using bagging and boosting.
2. Manipulate the input features:We select a subset of the feature which are needed for your model. eg:RandomForest
3. Manipulating the class labels: We create an artificial label say A0,A1.Here A0 corresponds to the 0 class and A1 corresponds to 1st class.This is mainly used in ECOC(error correcting output coding).
4. Manipulating the learning algorithm: We take artificial neural network model we can change the weights or we can change the neuron connection and topology to generate a new model.

Logical View of Ensemble classifier



RANDOM FOREST

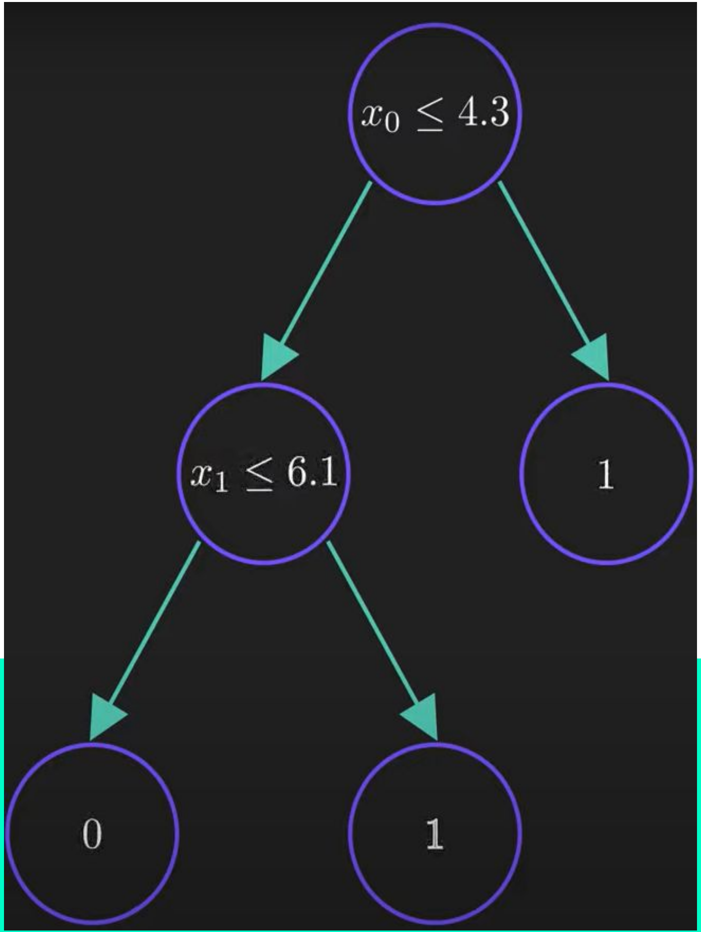
DECISION TREE BASIC

It splits the dataset recursively using the decision nodes unless we are left with pure leaf nodes and finds the best split by maximizing the entropy gain.

If a data sample satisfies the condition at decision node then it moves to the left child else it moves to the right and finally reaches a leaf node where a class label is assigned to it.

In a decision tree, each internal node represents a 'test' on an attribute (e.g., whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). A node that has no children is a leaf."

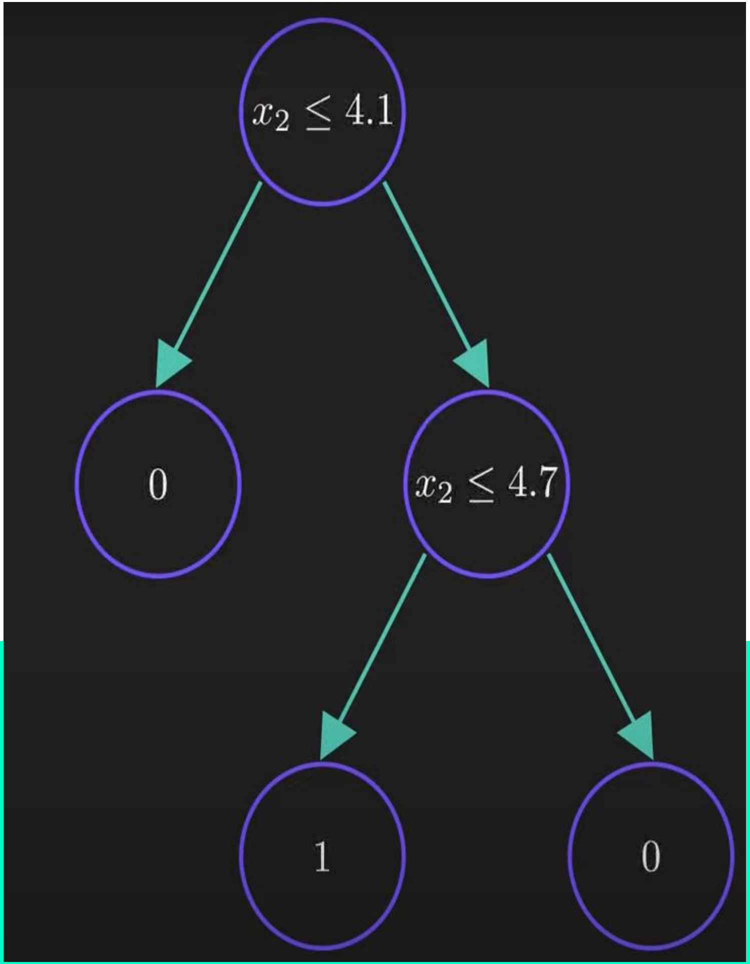
id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1



Why Random Forest?

Decision Trees are highly sensitive to the training data which could result in high variance.

id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	6.5	4.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1



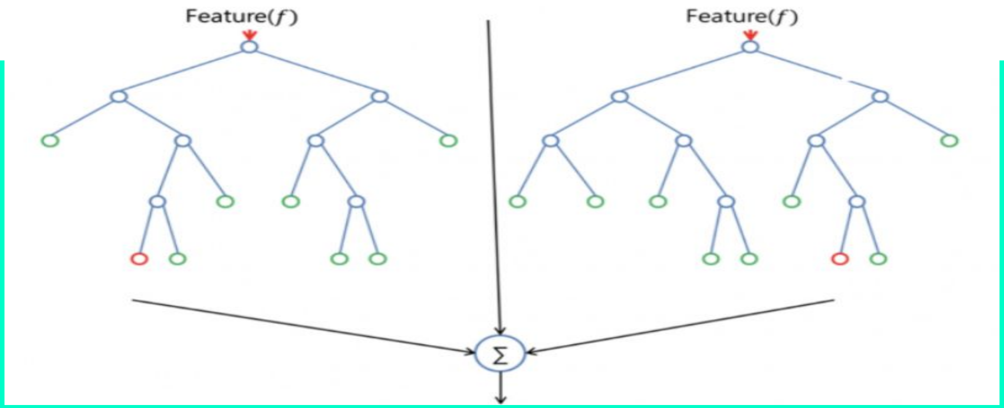
RANDOM FOREST DEFINITION

The random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction **and it's much less sensitive to the training data.**

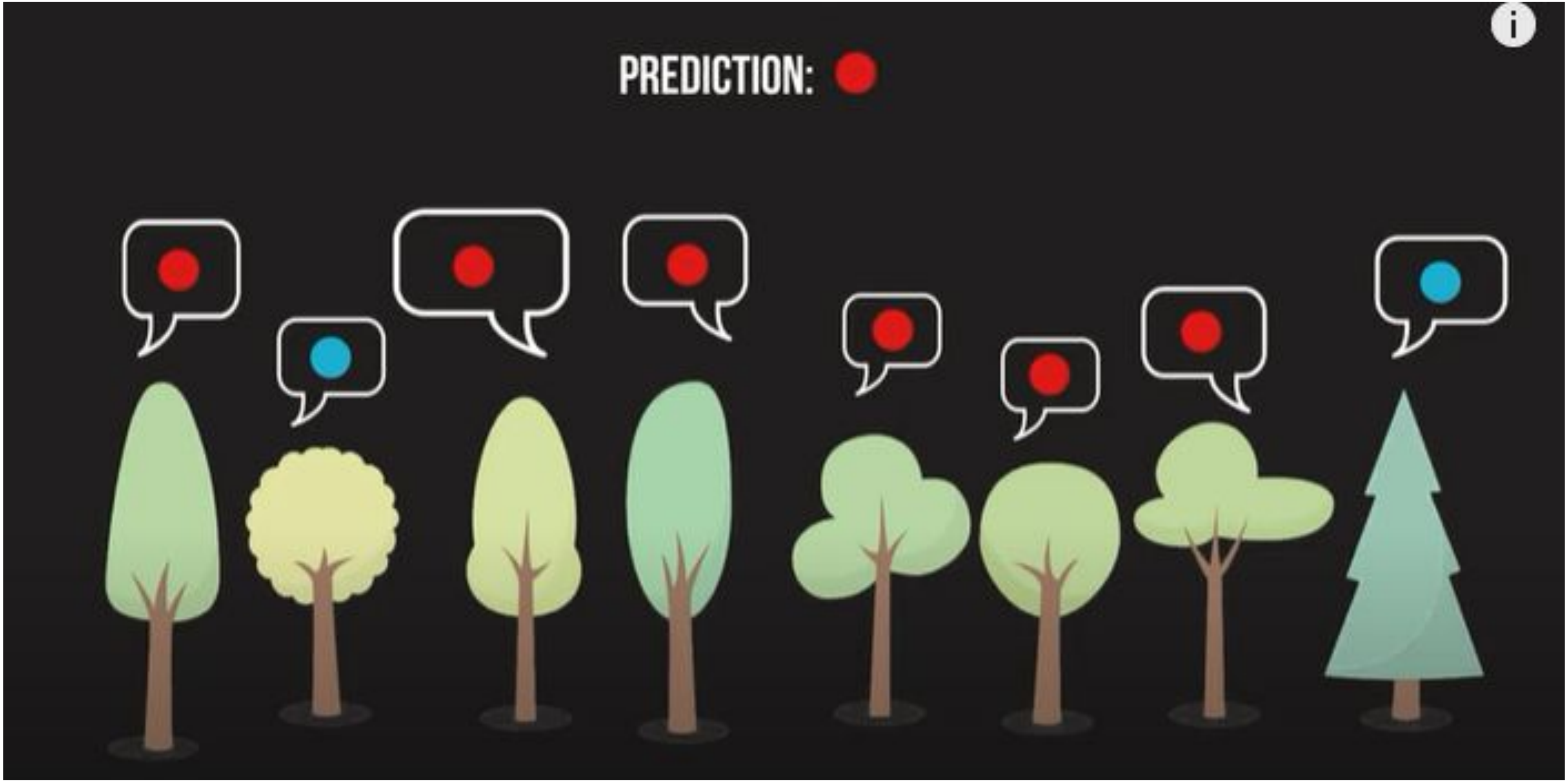
The random forest algorithm is used in a lot of different fields, like banking, the stock market, medicine and e-commerce.

We use multiple trees and hence the name forest.

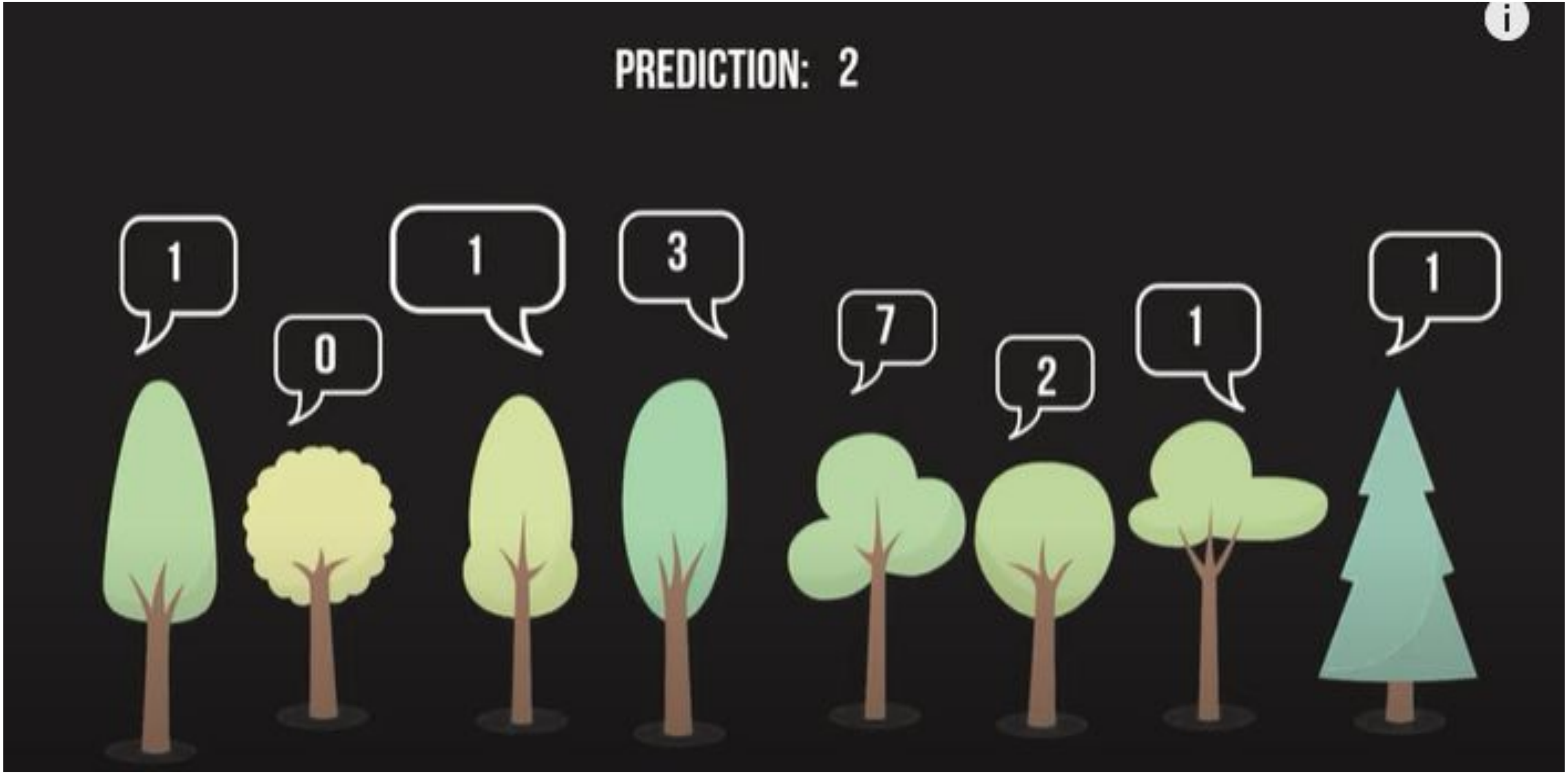
But why 'RANDOM'?

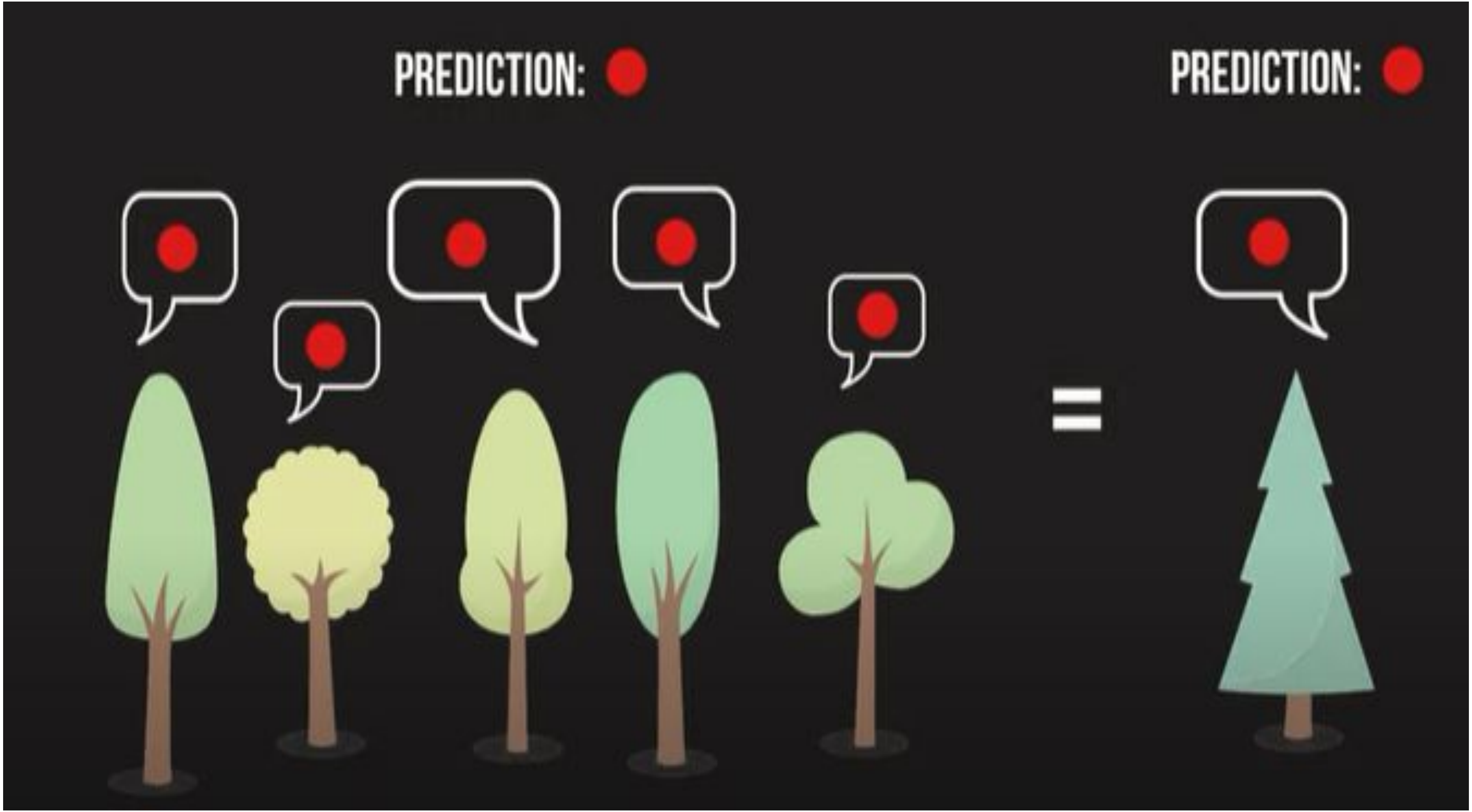


For Classification

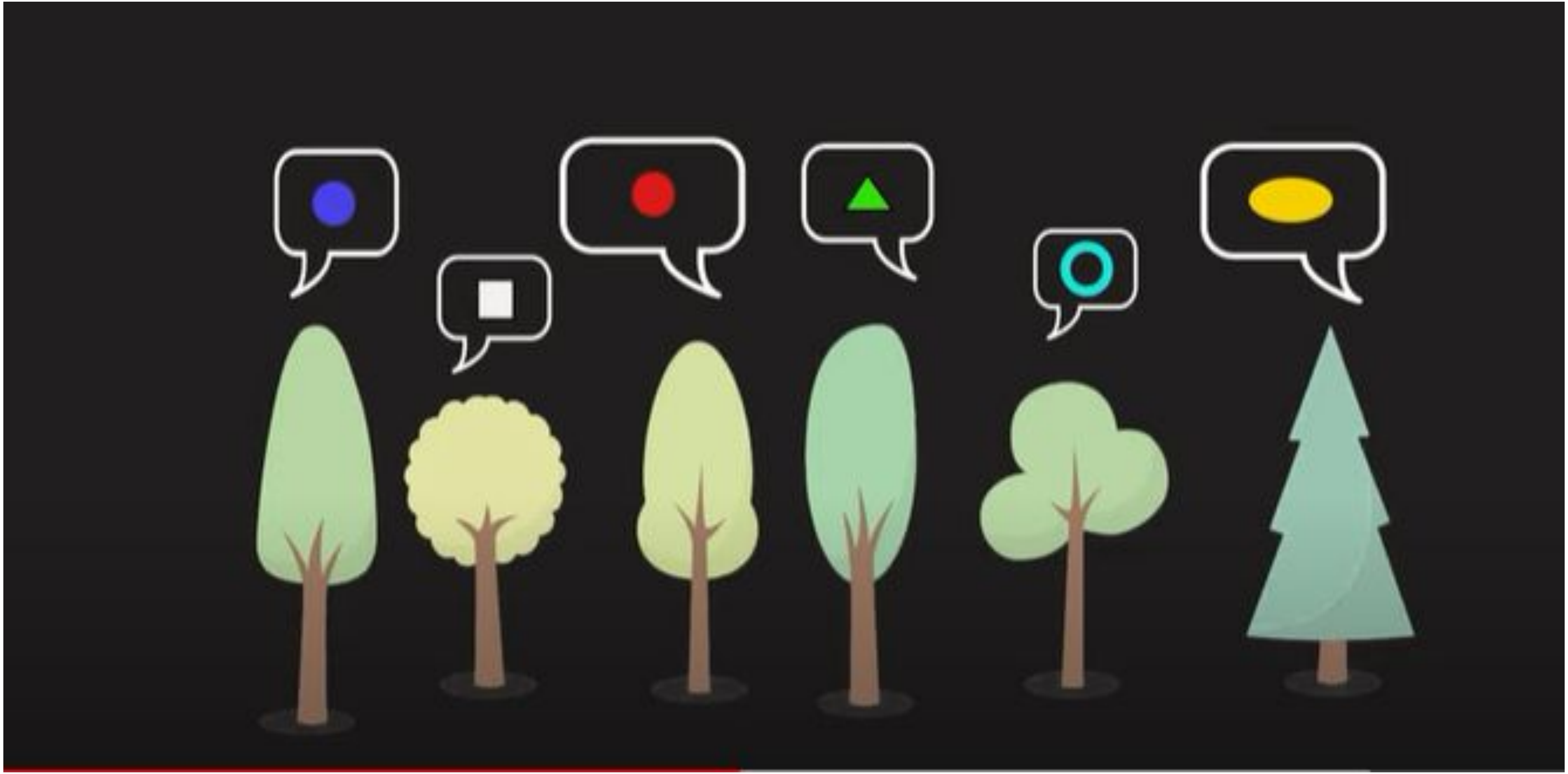


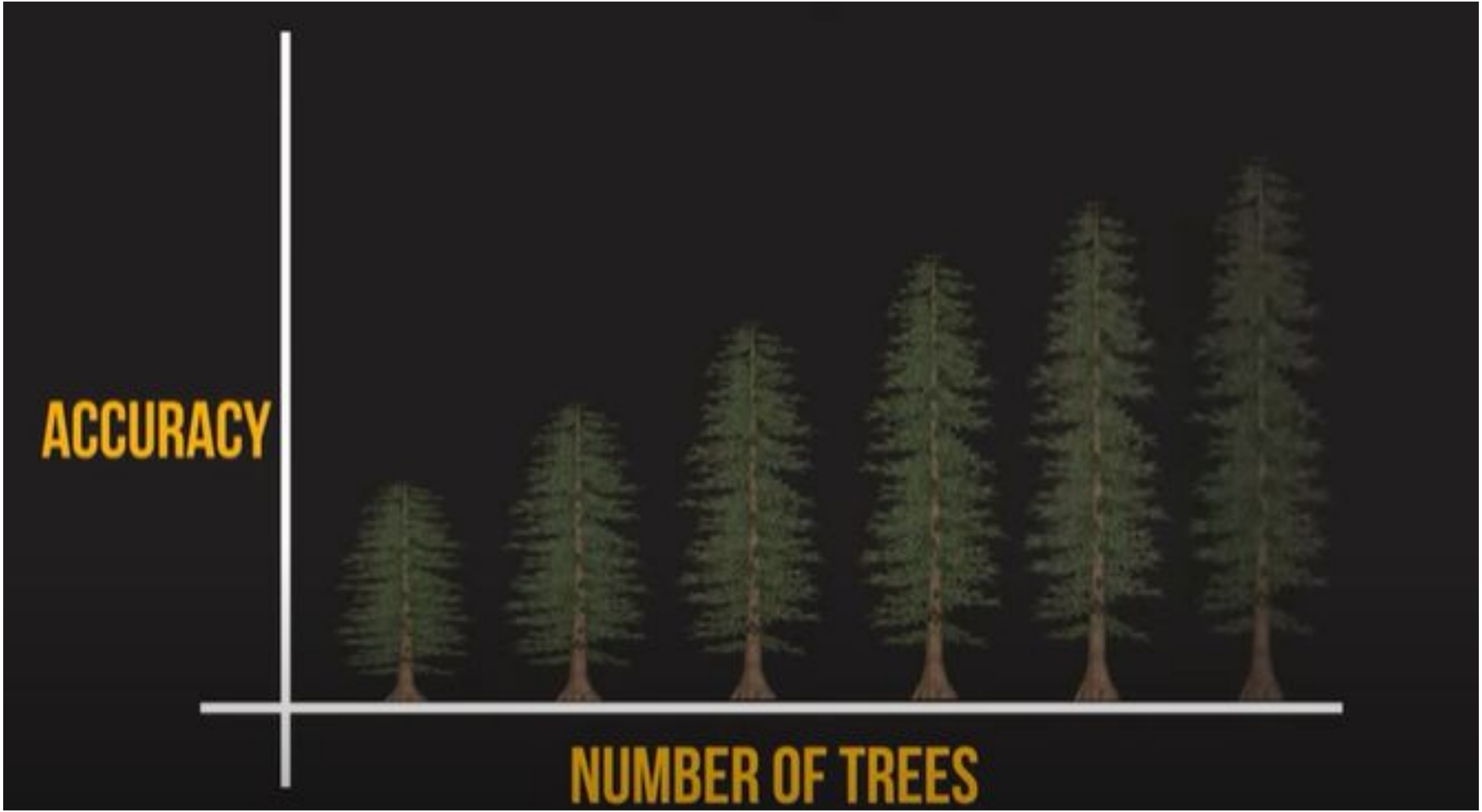
For regression :




PREDICTION:
PREDICTION:

Uncorrelated is important for Random Forest –



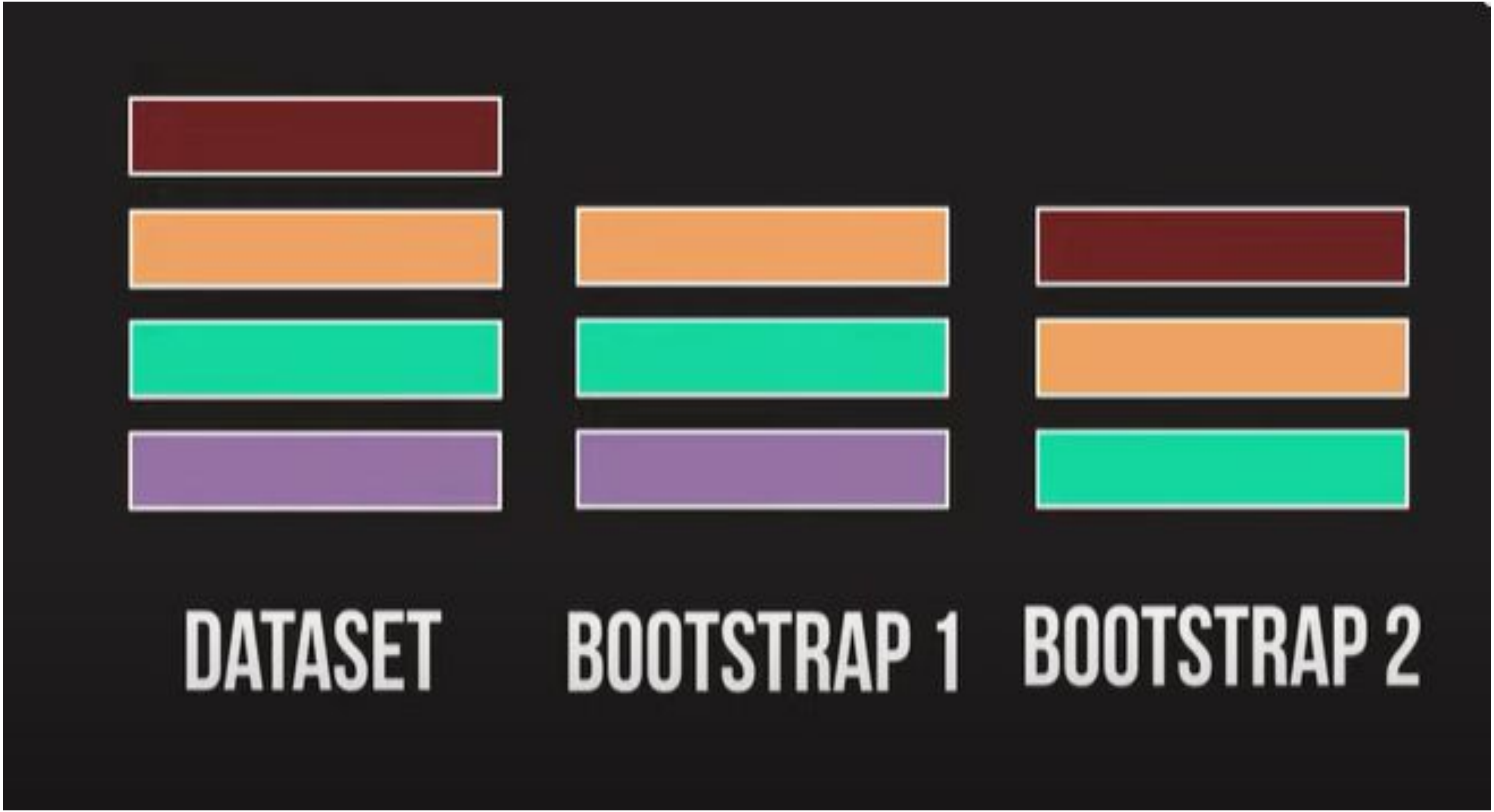

ACCURACY
NUMBER OF TREES

METHODS TO DECORRELATE TREES:

- BOOTSTRAP AGGREGATING / BAGGING

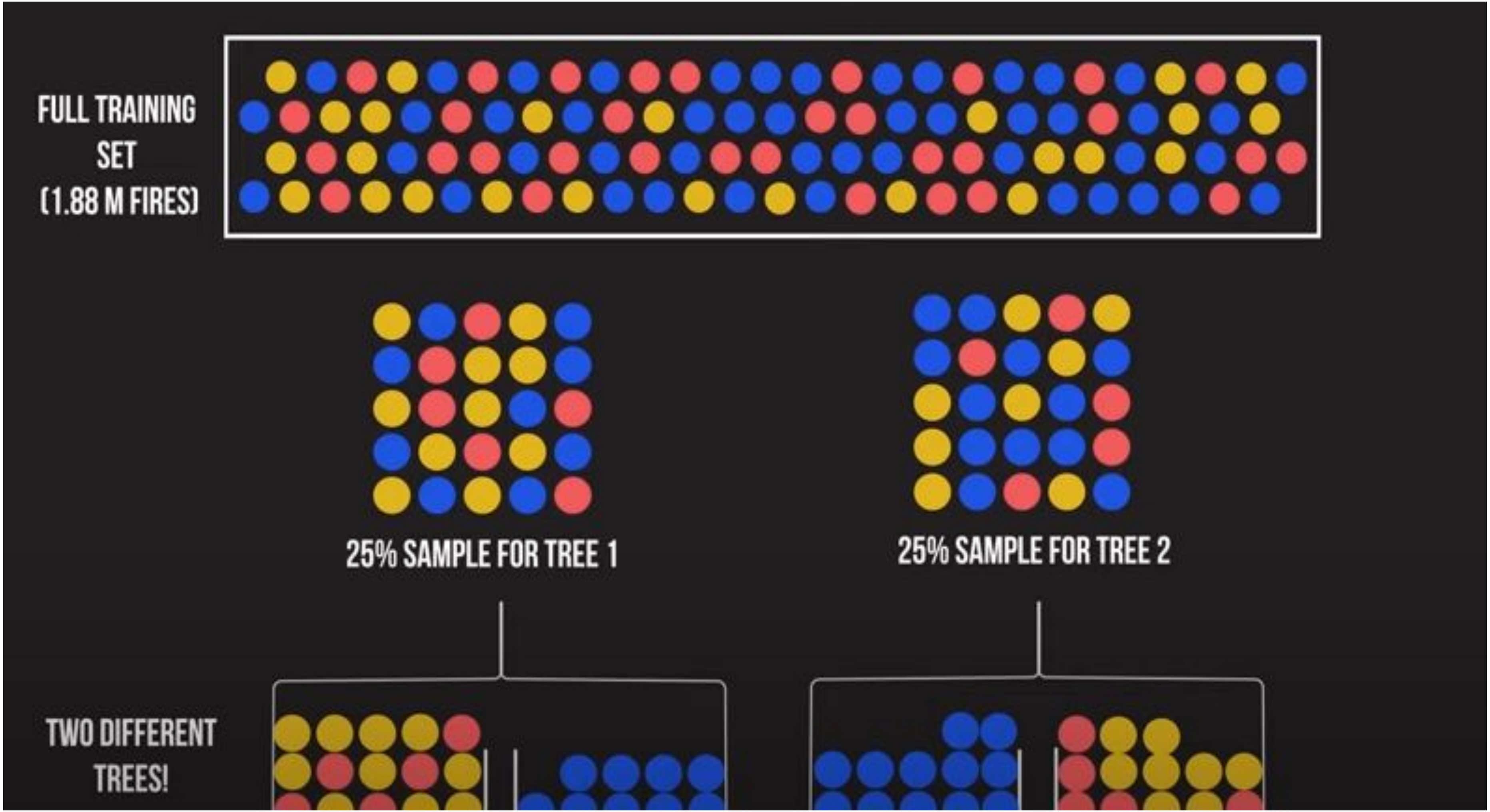
- FEATURE RANDOMNESS

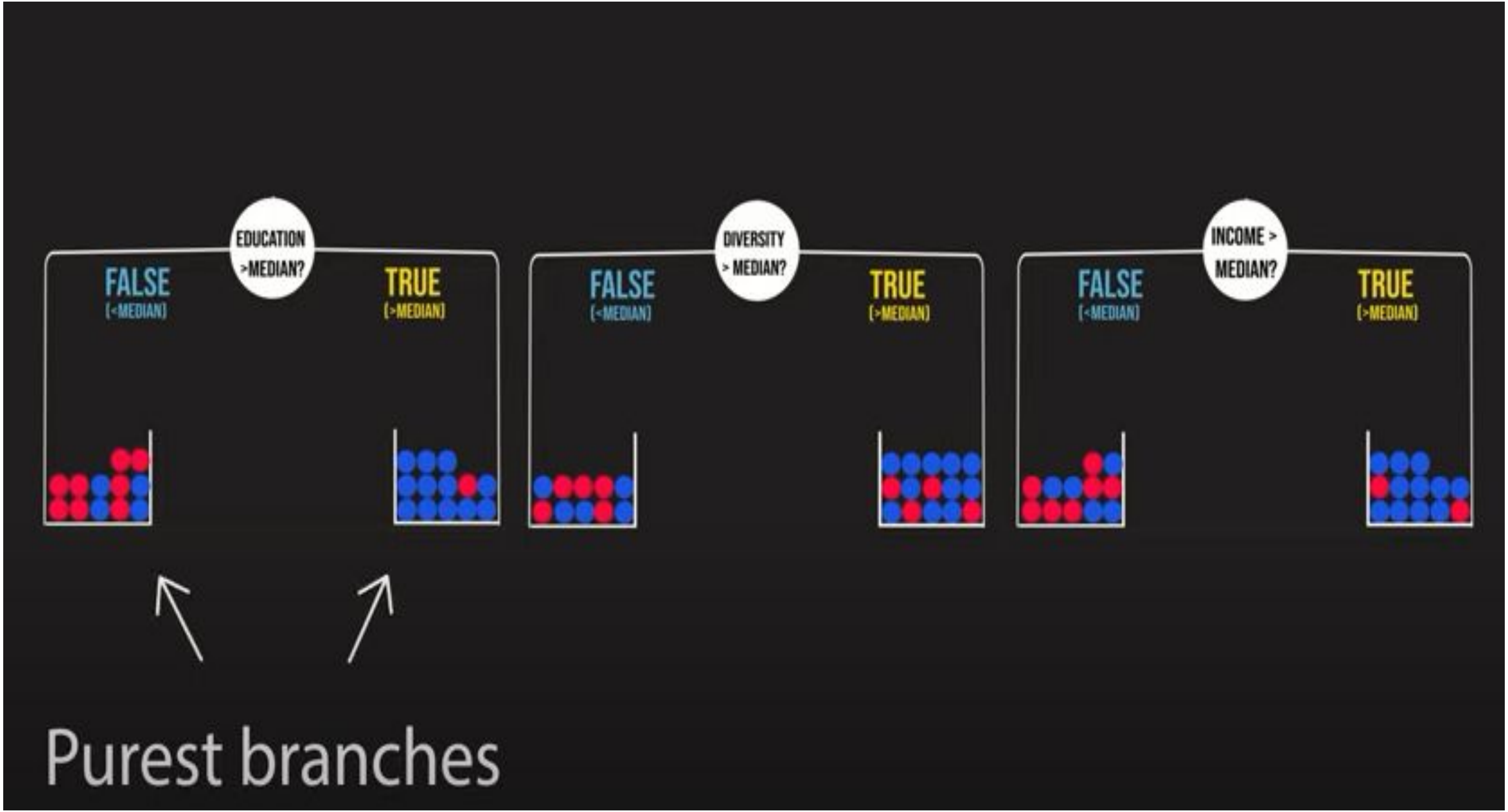
Bootstrapping



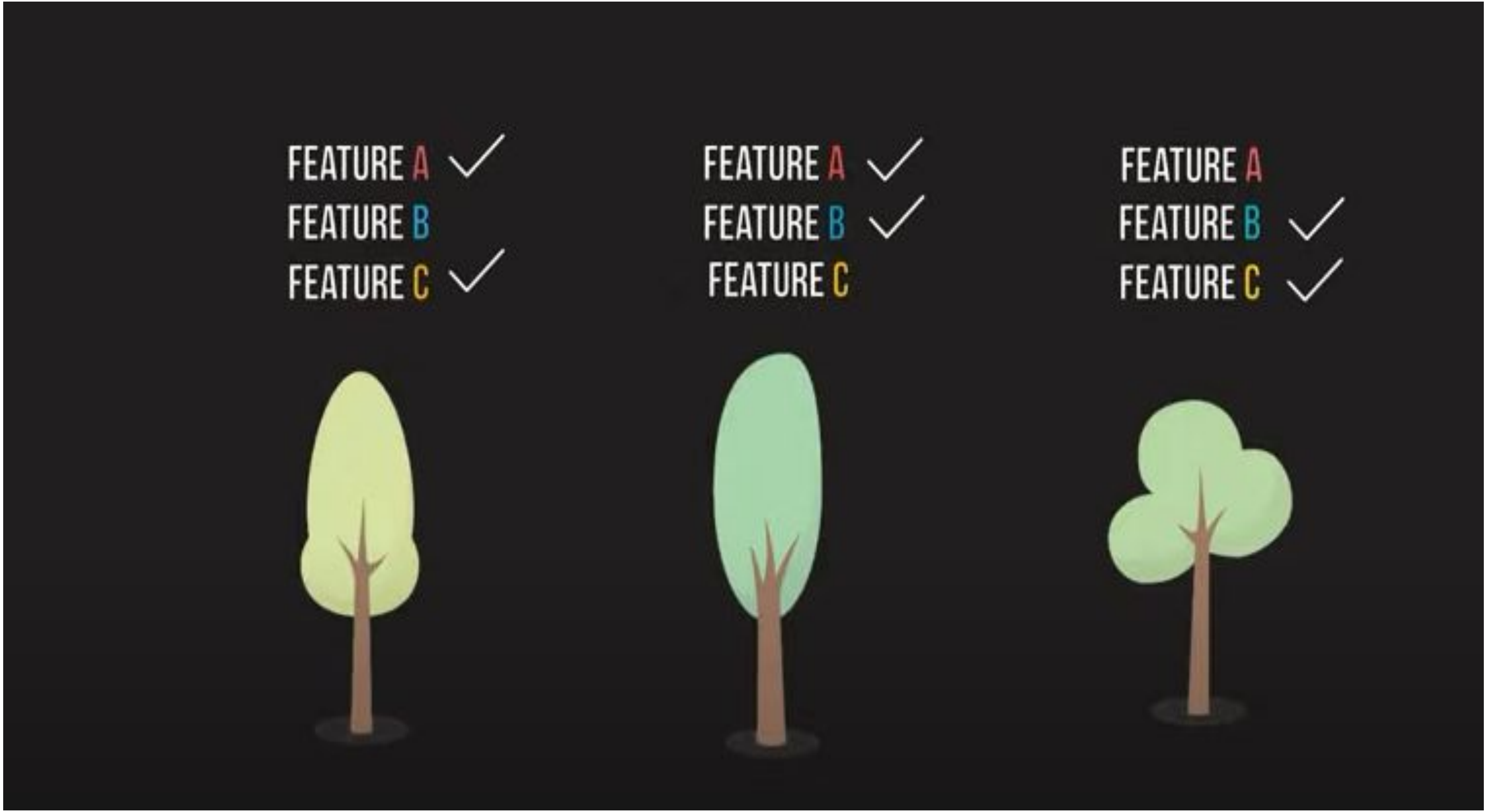
DATASET
BOOTSTRAP 1
BOOTSTRAP 2

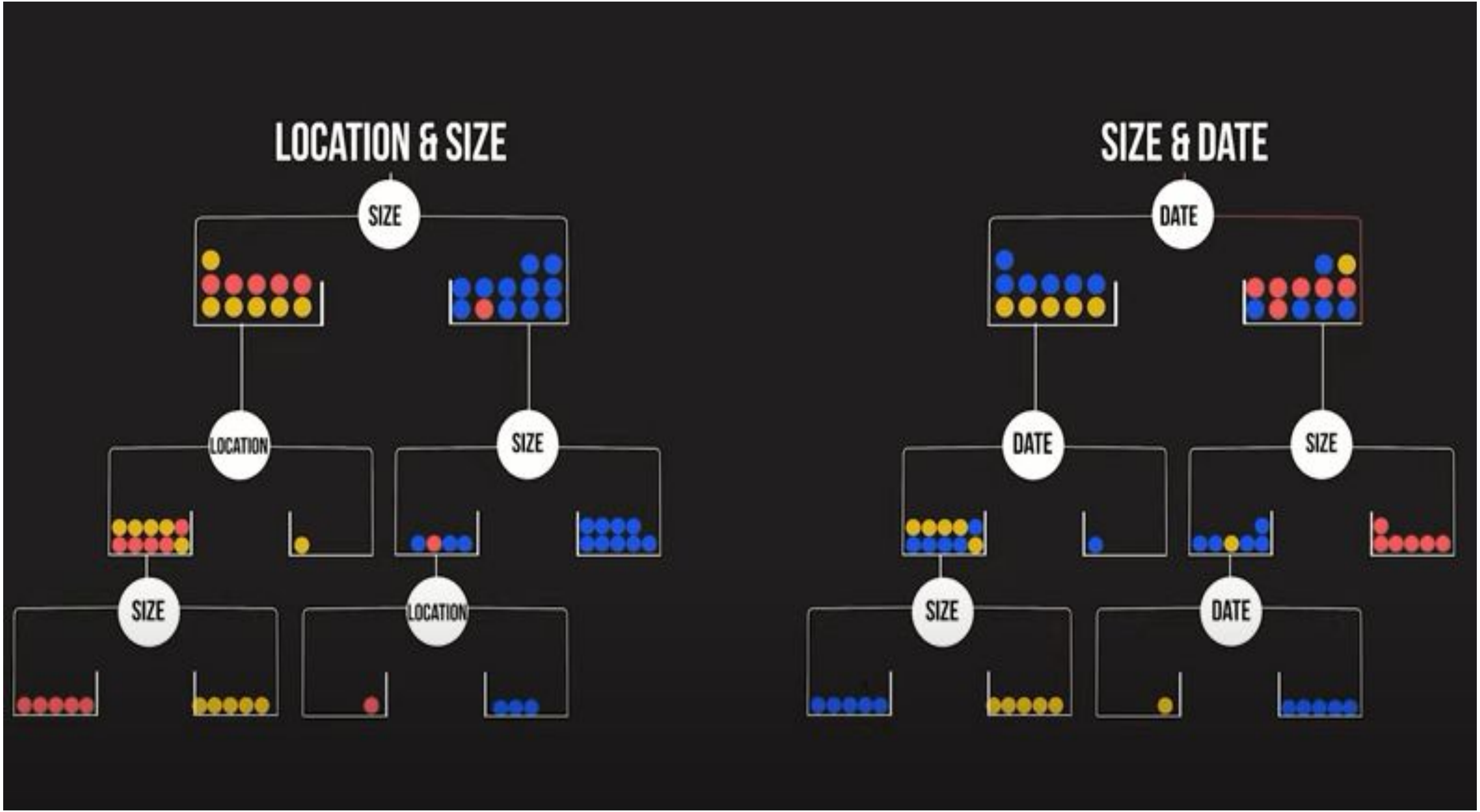

ENTIRE TRAINING DATASET
RANDOMLY SAMPLED TRAINING SETS


FULL TRAINING SET (1.88 M FIRES)
25% SAMPLE FOR TREE 1
25% SAMPLE FOR TREE 2
TWO DIFFERENT TREES!


EDUCATION >MEDIAN?
FALSE (<MEDIAN)
TRUE (>MEDIAN)
DIVERSITY > MEDIAN?
FALSE (<MEDIAN)
TRUE (>MEDIAN)
INCOME > MEDIAN?
FALSE (<MEDIAN)
TRUE (>MEDIAN)
Purest branches

Feature Randomness


FEATURE A
FEATURE B
FEATURE C
FEATURE A
FEATURE B
FEATURE C
FEATURE A
FEATURE B
FEATURE C


LOCATION & SIZE
SIZE
LOCATION
SIZE
SIZE
LOCATION
SIZE & DATE
DATE
DATE
SIZE
SIZE
DATE

Now Let’s put them together…

id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

id
2
0
2
4
5
5

id
2
1
3
1
4
4

id
4
1
3
0
0
2

id
3
3
2
5
1
2

id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

id
2
0
2
4
5
5

x_0, x_1

id
2
1
3
1
4
4

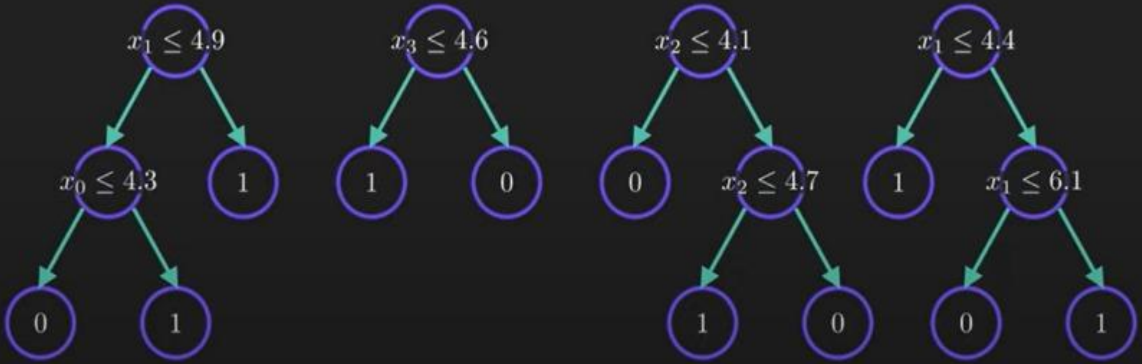
x_2, x_3

id
4
1
3
0
0
2

x_2, x_4

id
3
3
2
5
1
2

x_1, x_3

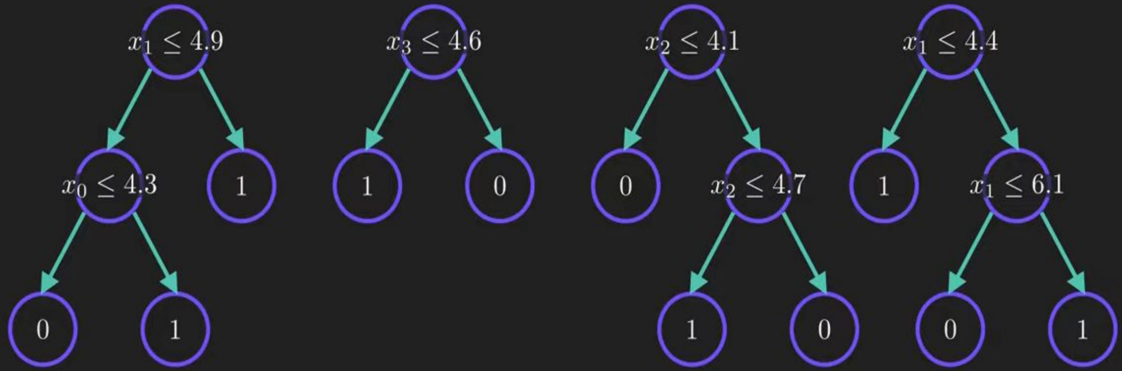


id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

id	id	id	id
2	2	4	3
0	1	1	3
2	3	3	2
4	1	0	5
5	4	0	1
5	4	2	2

x_0, x_1 $\quad$ x_2, x_3 $\quad$ x_2, x_4 $\quad$ x_1, x_3

2.8	6.2	4.3	5.3	5.5



1 $\quad$ 0 $\quad$ 1 $\quad$ 1

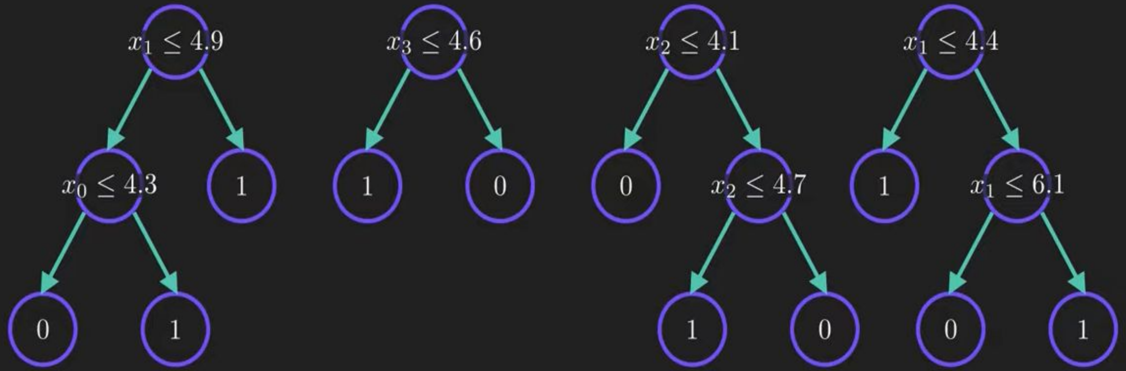
id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

2.8	6.2	4.3	5.3	5.5

Why random?

id	id	id	id
2	2	4	3
0	1	1	3
2	3	3	2
4	1	0	5
5	4	0	1
5	4	2	2

x_0, x_1 x_2, x_3 x_2, x_4 x_1, x_3

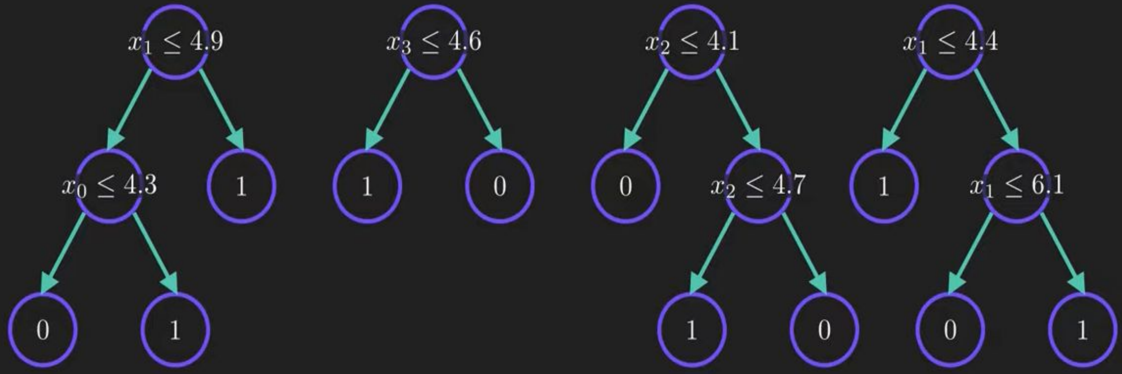


1 0 1 1

id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

id	id	id	id
2	2	4	3
0	1	1	3
2	3	3	2
4	1	0	5
5	4	0	1
5	4	2	2
x_0, x_1	x_2, x_3	x_2, x_4	x_1, x_3

2.8	6.2	4.3	5.3	5.5



1 0 1 1

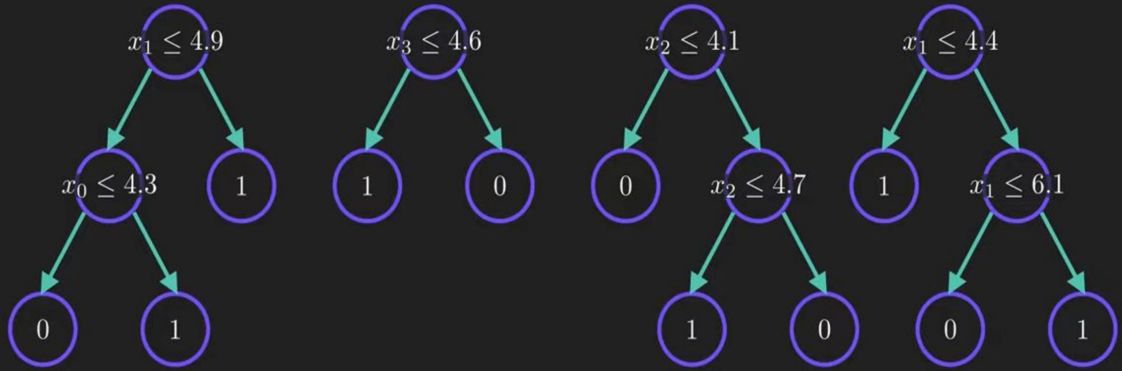
Why bootstrapping and Feature Selection?

id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

id	id	id	id
2	2	4	3
0	1	1	3
2	3	3	2
4	1	0	5
5	4	0	1
5	4	2	2

x_0, x_1	x_2, x_3	x_2, x_4	x_1, x_3

2.8	6.2	4.3	5.3	5.5



1	0	1	1

Bagging = Bootstrapping + Agreegating

Thank You