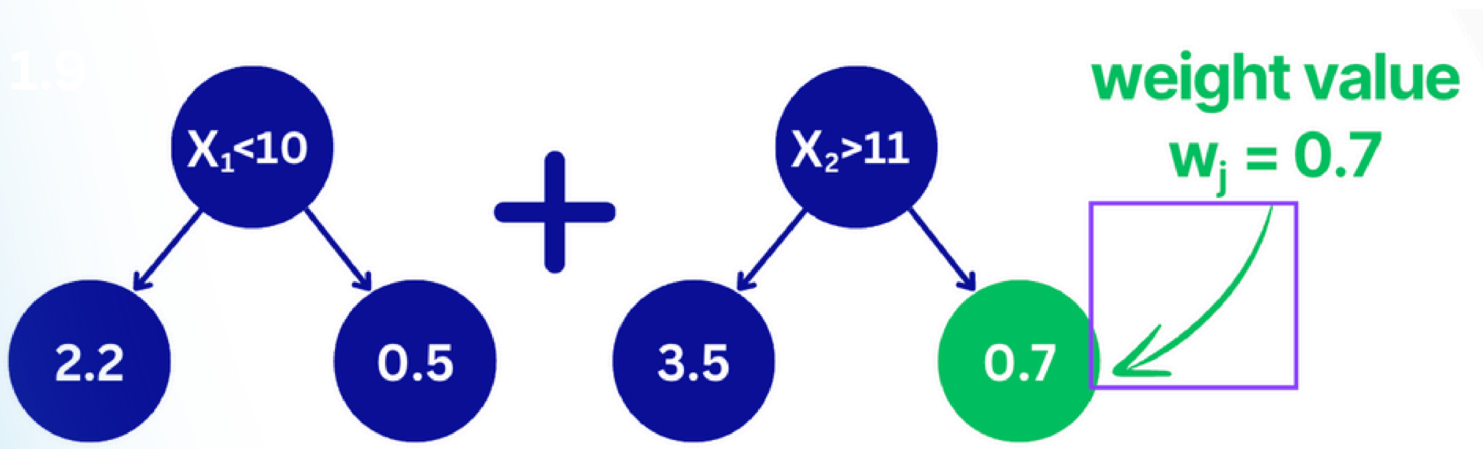
SAVE/LIKE FOR LATER

Step-by-step breakdown of

GRADIENT BOOSTING REGULARIZATION

*(learn **for interview** and **daily use**)*

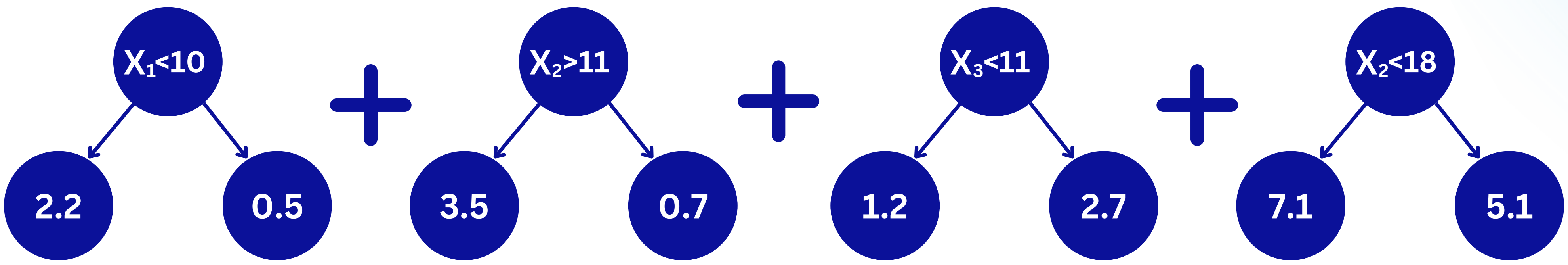


$$L(F) = \sum^{N} \overbrace{l\left(y_i, F(x_i)\right)}^{\text{loss (e.g. MSE)}} + \overbrace{\Omega(F)}^{\text{Penalty}}$$

TIMUR BIKMUKHAMETOV

SAVE/LIKE FOR LATER

Gradient boosting is a powerful algorithm capable to learn complex patterns in tabular data but can overfit quickly.



There are several parameters that must be be controlled to avoid it.

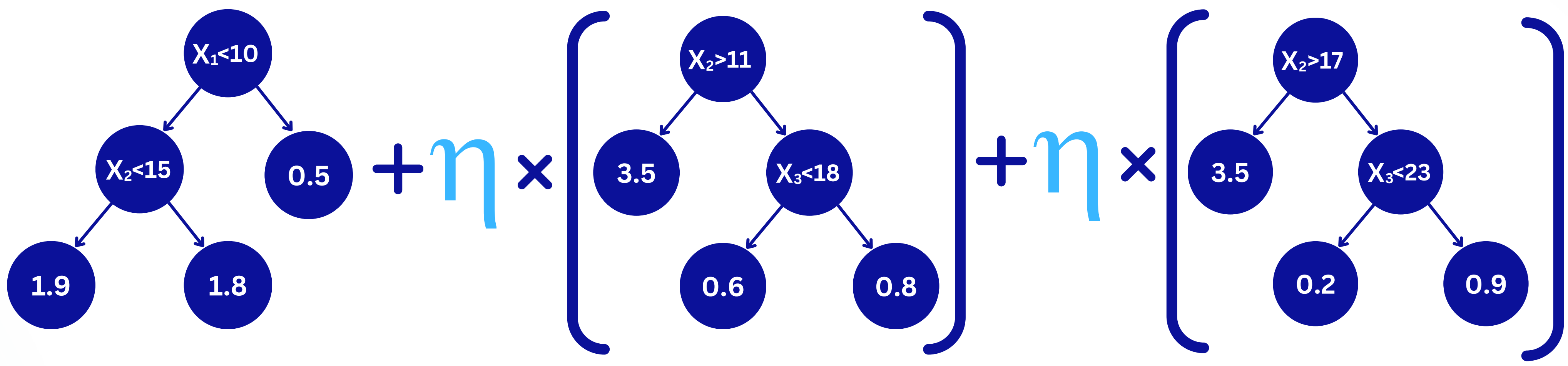
Let’s breakdown them down!

1. Learning rate (aka shrinkage, eta)

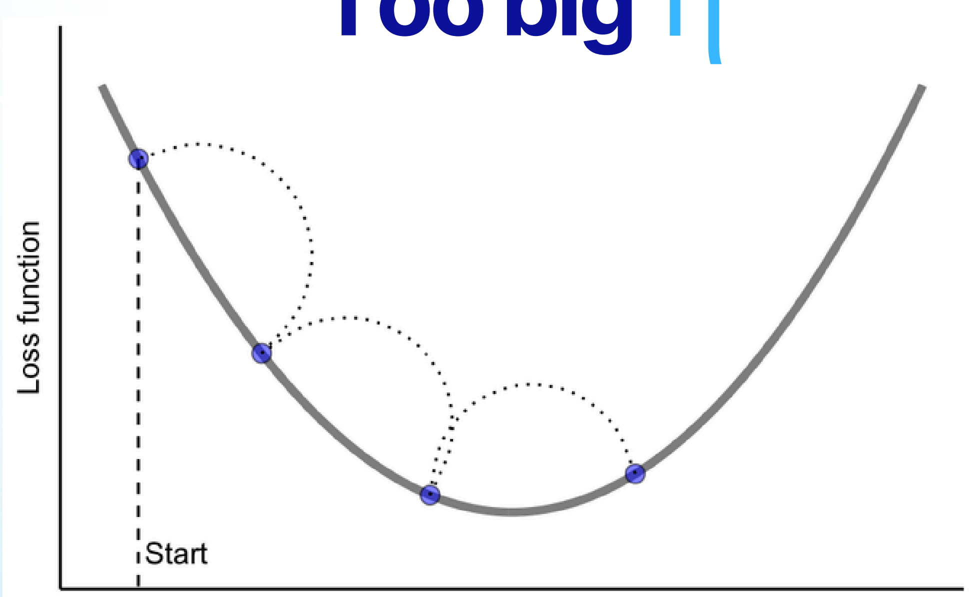
Learning rate (shrinkage) parameter multiplies the output of each weak learner (tree).

A small value of it literally shrinks each tree output weight values, so smaller values are added to the sum of the tree ensemble.

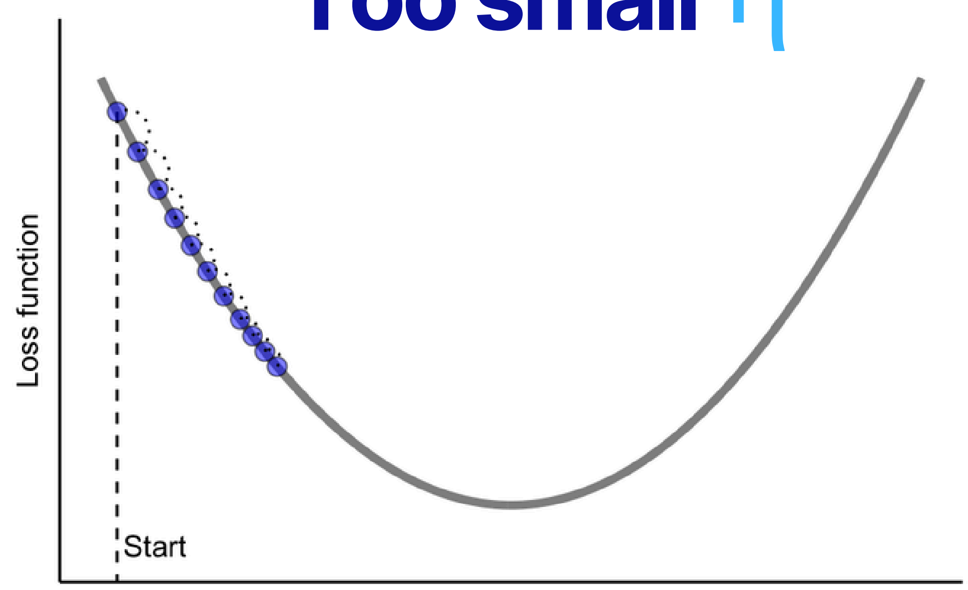
As such, decreasing this parameter makes the boosting process more conservative because smaller boosting steps are taken.



Too big η



Too small η



See next one!

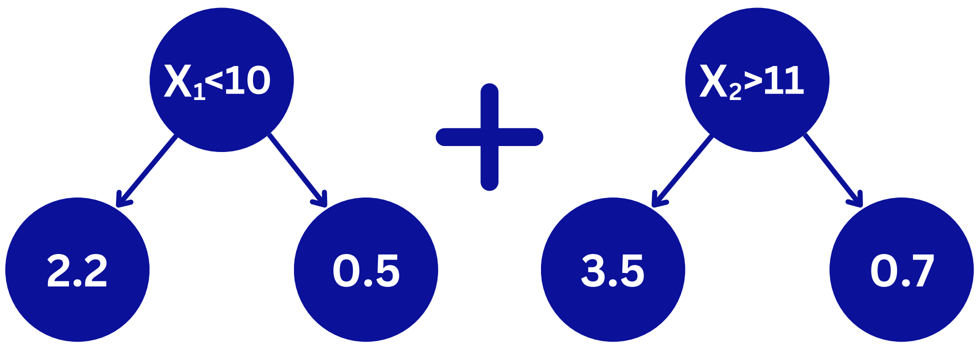
SAVE/LIKE FOR LATER

2. Tree depth

Tree depth defines how deep (complex) the decision trees are.

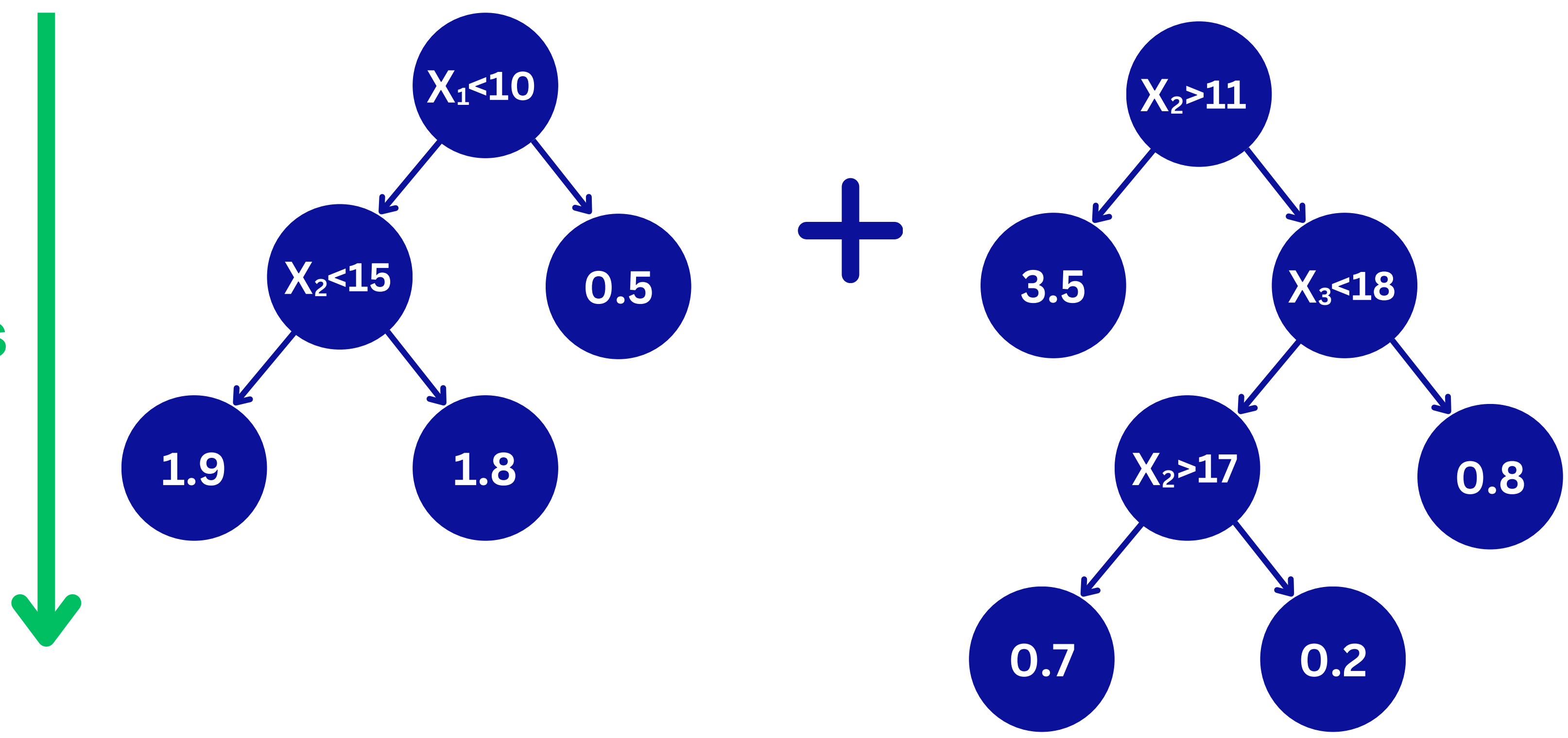
A small tree depth makes the algorithm more conservative and avoid overfitting. However, too small values with combined with a small number of trees can lead to underfitting.

Shallow threes



If the depth increases, it makes the each tree to learn more complex patterns which can result in learning noise. This can quickly lead to overfitting.

Depth increases



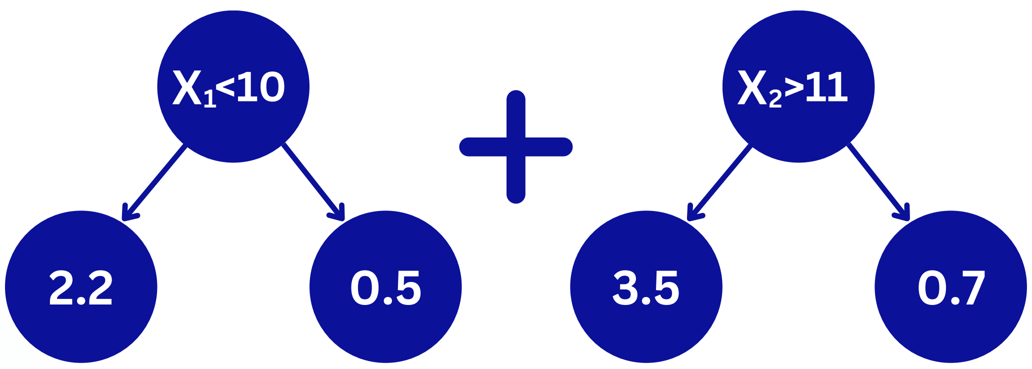
See next one!

3. Number of trees (with early stopping)

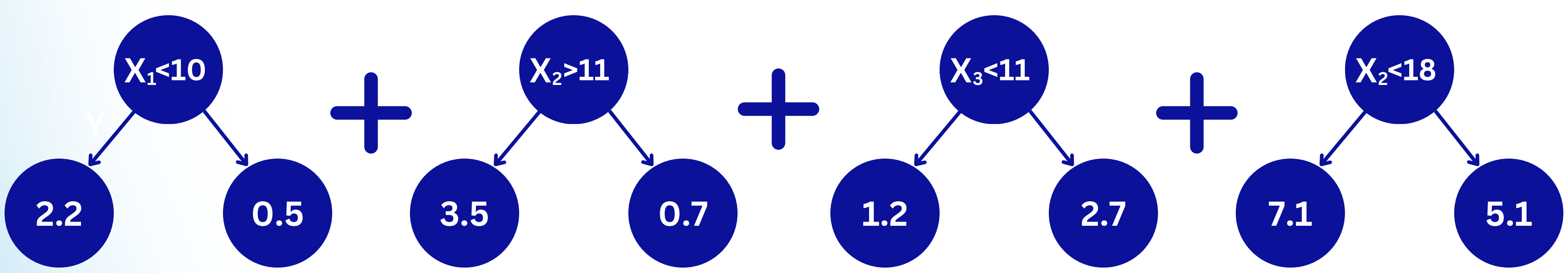
The number of trees will make the algorithm to learn more complex patters (reduce the bias but increase the variance).

A big number of trees will drive the algorithm to overfit.

A common way to select an optimal number of trees is to use early stopping technique.



Increasing number of trees makes to learn complex patterns but can lead to overfitting.



See next one!

4. Subsample ratio

Subsample ratio defines how much data is used to fit a decision tree at each iteration step.

If subsample ratio=0.5, it means 50% of the data is used to fit a decision tree. Subsampling is performed at each iteration.

Reducing the ratio helps to prevent overfitting because each tree can learn only a part of the information in the data.

It naturally makes to avoid learning complex patterns including noise.

Index	X1	X2	X3	Target
1	10	8	16	2.5
2	15	20	18	1.5
3	12	3	11	8.7
4	11	2	11	2.5
5	7	15	28	4.6
6	15	7	6	8.1

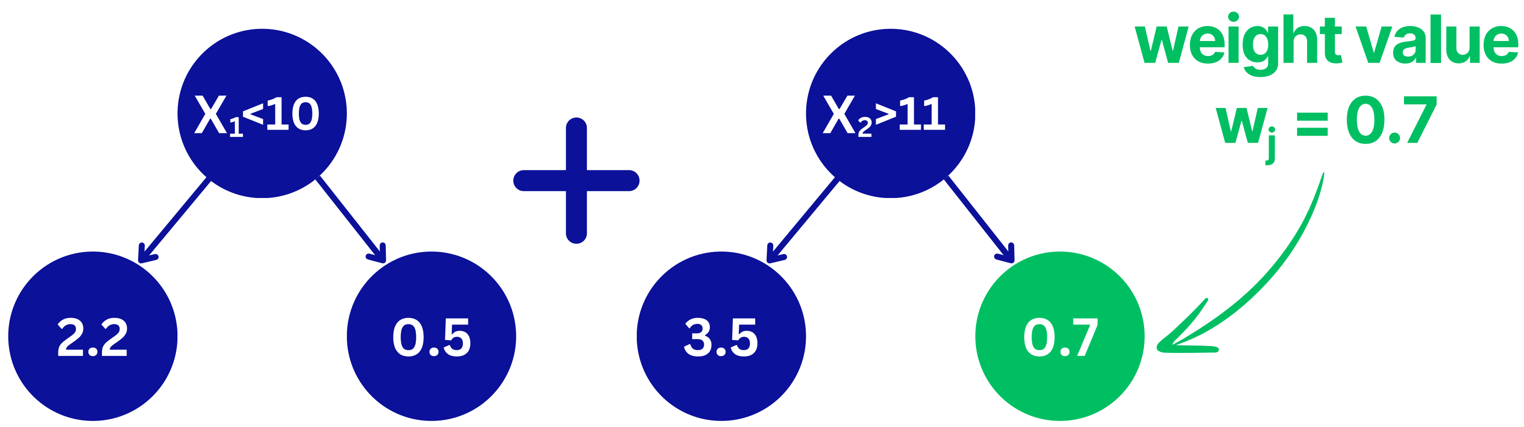
Only subsampled data is used at each iteration

See next one!

5. L1/L2 regularization

In Gradient Boosting, L1/L2 regularization penalizes weight values that are located at the leaf nodes.

Similar to linear regression, in L1, sum of the absolute leaf weight values are taken while for L2 - sum of squared values.



loss (e.g. MSE) — Penalty

$$L(F) = \sum_{i=1}^{N} \underbrace{l\left(y_i, F(x_i)\right)}_{\text{loss (e.g. MSE)}} + \underbrace{\Omega(F)}_{\text{Penalty}}$$

L1

$$\Omega(F) = \boxed{\alpha} \sum_{j=1}^{M} |w_j|$$

L2

$$\Omega(F) = \boxed{\lambda} \sum_{j=1}^{M} w_j^2$$

Tuning parameter

See next one!

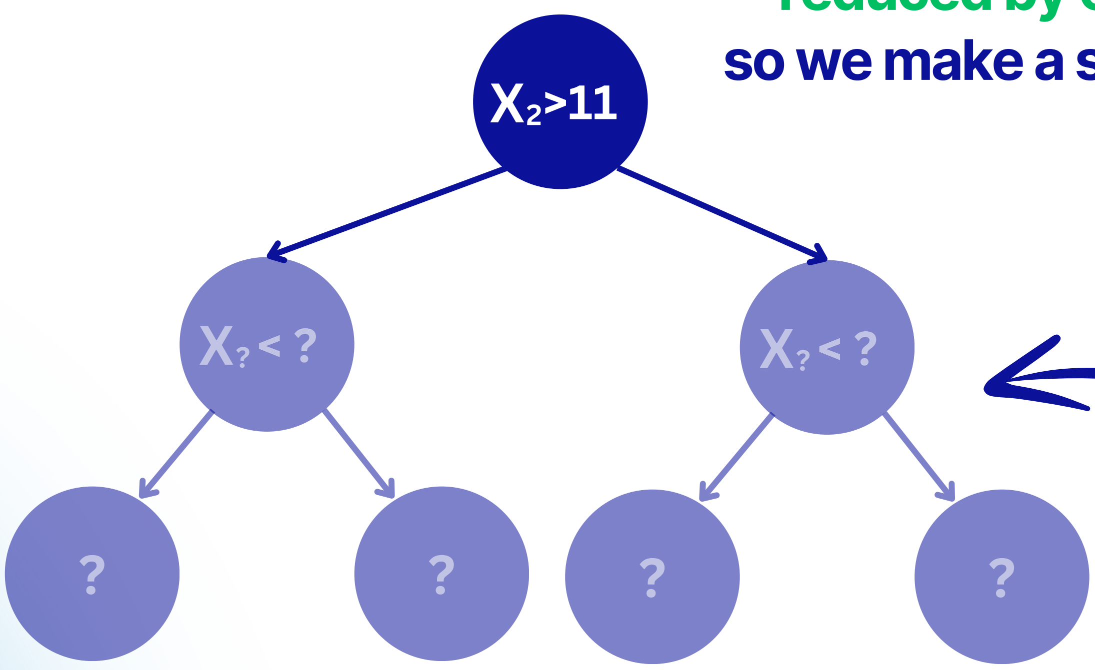
6. Minimum loss reduction

Loss reduction is used to calculate the feature candidates (X1, X2 and X3) and their values to make a split in each node.

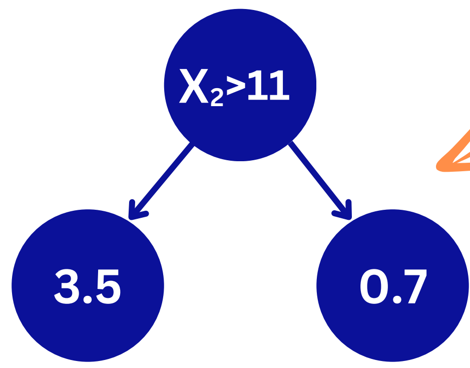
Assume the minimum loss reduction = 5

Assume after splitting based on X2, the loss reduced by 6, so we make a split

$$L(F) = \sum_{i=1}^{N} l\left(y_i, F(x_i)\right) + \Omega(F)$$



After evaluating all possibilities, we found that a maximum possible loss reduction at this depth is 4. So, we stop growing the tree.



If the minimum loss reduction is big, the trees stop growing quickly, so the algorithm will be more conservative.

Let's sum up!

SAVE/LIKE FOR LATER

Summary

1. Gradient boosting is a powerful algorithm capable to learn complex patterns in tabular data but can overfit quickly.

2. There are 6 main parameters to avoid overfitting and they have a different mechanisms to do so.

3. These parameters must be well-understood by a Data Scientist and to be selected by cross-validaiton.

Repost to network, so people know your expertise and interests!

Finally...

Want to see more breakdowns like this?

Like post

Follow me

so I know that the content resonates!

TIMUR BIKMUKHAMETOV