CLASSIFICATION II -BIS
BEYOND DECISION TREE

Dr. Lablanche Pierre-Yves
African Institute for Mathematical Sciences

REMINDER

Building a full (until complete purity) single decision tree has several drawbacks. Constraining the tree construction (complexity) allows to :

- limit the resources used
- Reduce the overfitting issue (does not suppress it)

But does not solve the local optimal vs global optimal problem!

Solution : **Ensemble Method**

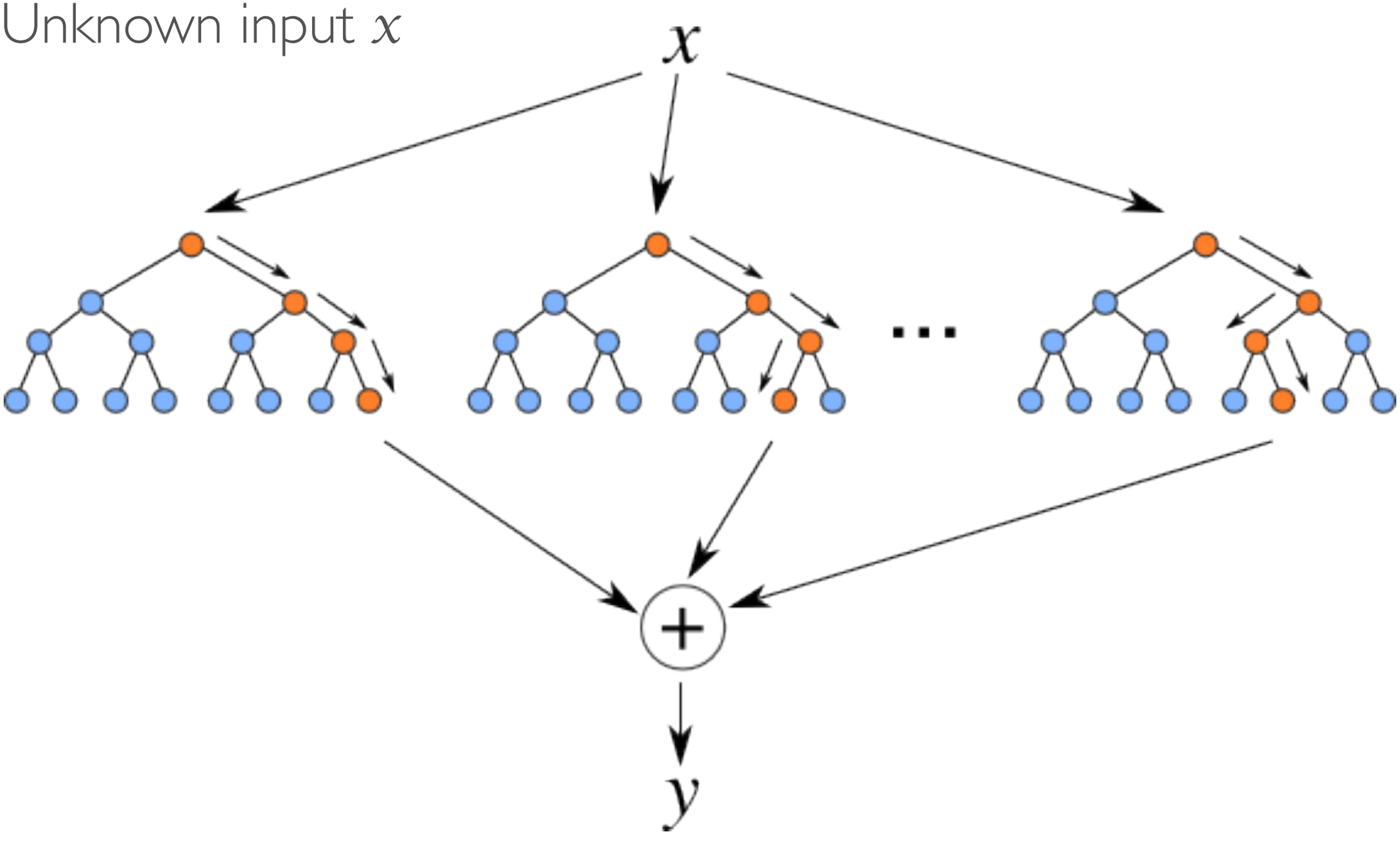
BAGGING

Bagging (also called **B**ootstrap **Agg**regat**ing**) in three simple steps : from data set $\mathcal{D}$ size n

- Chose m new sub-training set of size n' from the original training set. (using sampling with replacement)
- Build a decision tree for each sub-set (m trees total)
- Combine each tree prediction by voting (averaging for regression)

BAGGING

Unknown input x



FROM TREE TO FOREST

Ensemble = more than one

More than one tree = Forest

Thus :

Bootstrap Aggregation + Decision Tree
= Decision Forest

Where does the "Random" come from ?

CLEVER BAGGING

Bagging does not prevent decision trees from being correlated.

If decision trees are correlated, bagging is just a waste of resource.

Very complicated to predict how similar the trees will be.

Solution : Force Randomness at decision node

RANDOM FOREST

RANDOM FOREST

For each tree, limit the number of features to consider for the bet split at a decision node - Random draw of k' features out of original k features.

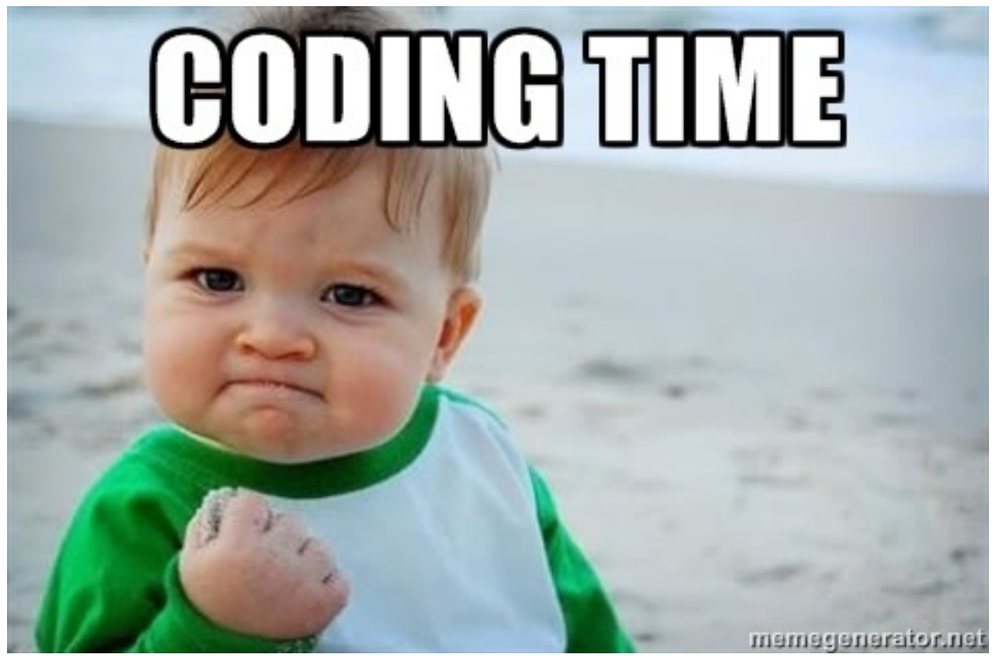
Best split threshold still computed by the machine.

Finally :

Bagging + Random Feature Selection Decision Tree = Random Forest

CODING TIME (1)

- Use **`RandomTreeClassifier`** to classify the iris dataset.
- Play with the different parameters!



MORE RANDOMNESS !!!

Possible to push randomisation a step further.

Randomly choose k' features AND threshold by feature

Keep the best randomly generated feature/threshold pair

Bagging +
Random Feature and Threshold Decision Tree
= Extra (Randomised) Trees

…does not necessarily perform better than Random Forests.

CODING TIME (2)

- Use **`ExtraTreesClassifier`** to classify the iris dataset.
- Recall **`DecisionTreeClassifier()`** and **`RandomTreeClassifier()`** to classify the iris dataset.
- Compare the three classifiers!