Random Forests

Yvenn Amara-Ouali

2024

General principle

Setting

- Random forests are a class of algorithms used for regression and classification problems
- They are often used in applied fields since they handle high-dimensional settings
- They have good predictive power and can outperform state-of-the-art methods
- But mathematical properties of random forests remain a bit obscure

Nomenclature (regression example)

- We are given a training set $Dn = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ where the pairs $(X_i, Y_i) \in [0,1]^d \times \mathbb{R}$ are i.i.d. distributed as (X, Y)
- We assume that

$$Y = m(X) + \varepsilon$$

- We want to build an estimate of the regression function m using random forest algorithm

Session outline

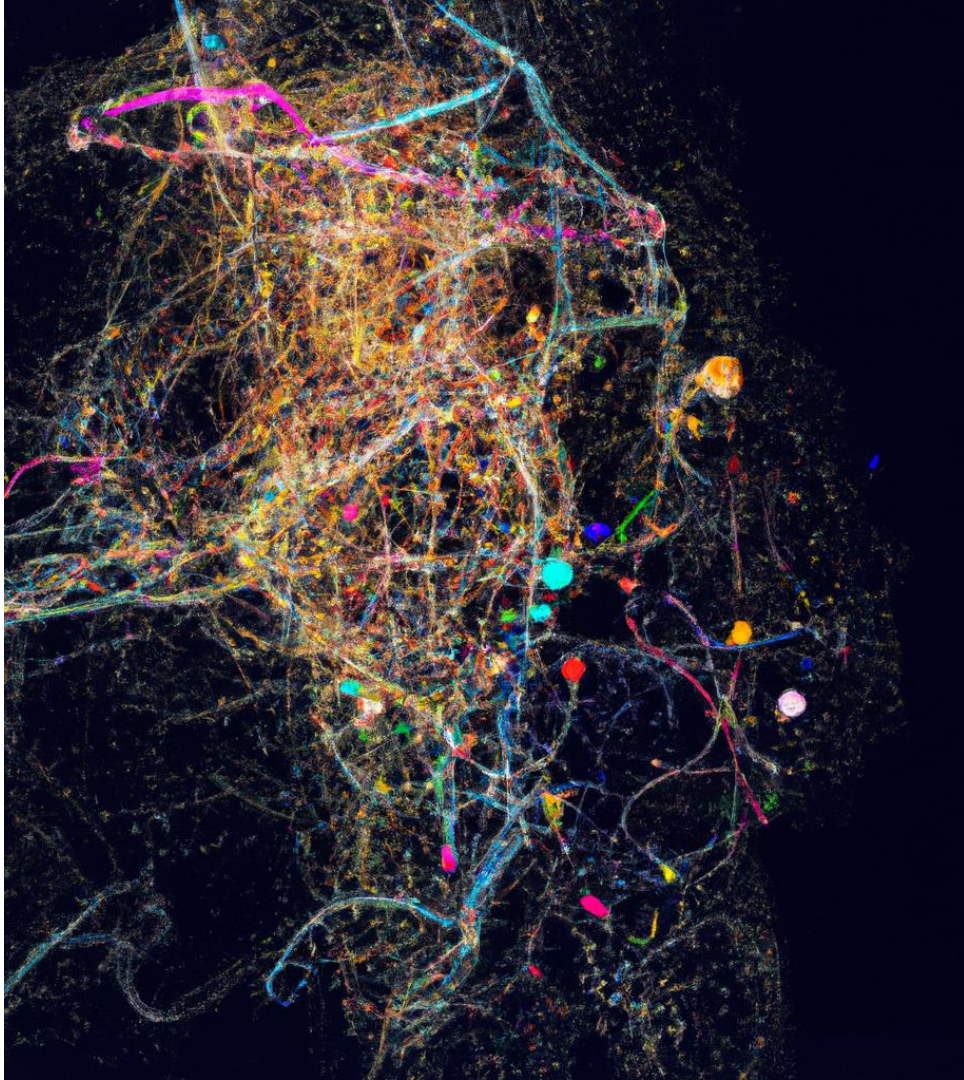
Decision Trees
- CART

Bagging
- Bootstrap Aggregation

Random Forests
- The wisdom of the forest

Decision Trees

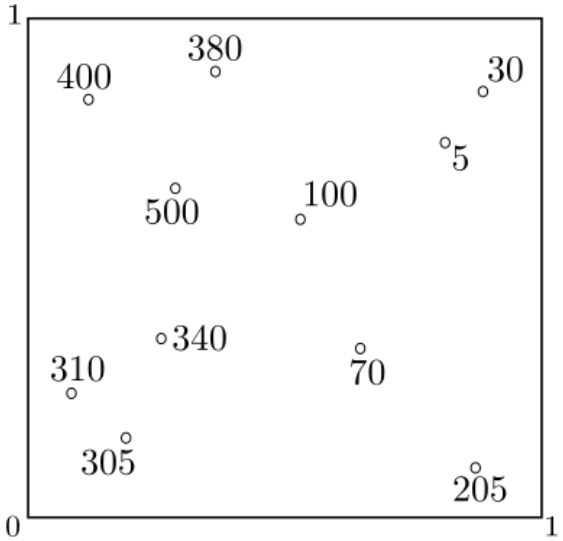
CART



Building a decision tree

Procedure

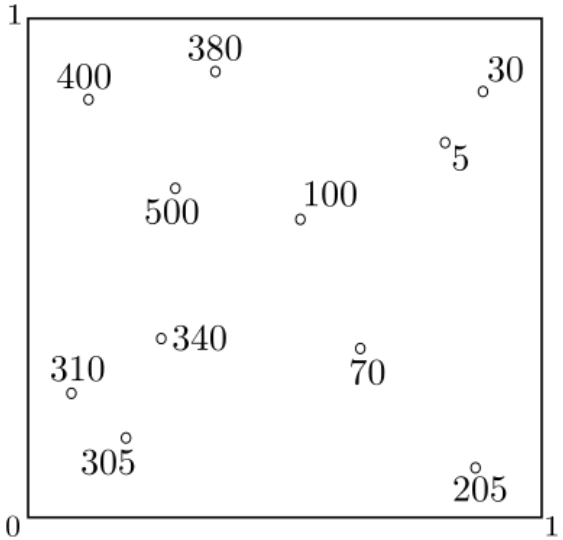
- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied



Building a decision tree

Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied

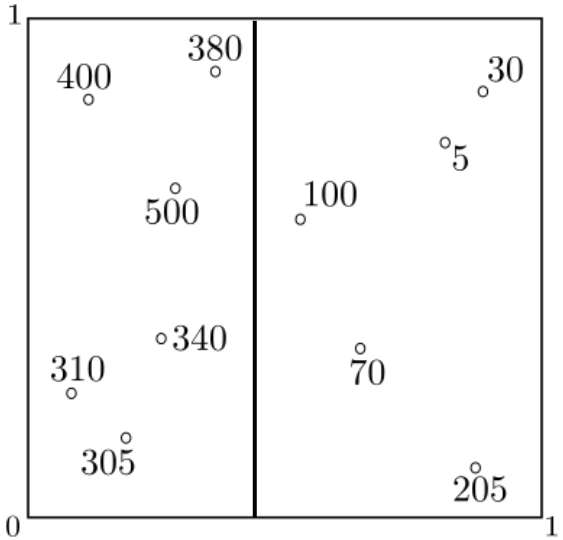


$k = 0$

Building a decision tree

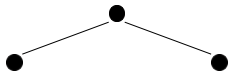
Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied



$k = 0$

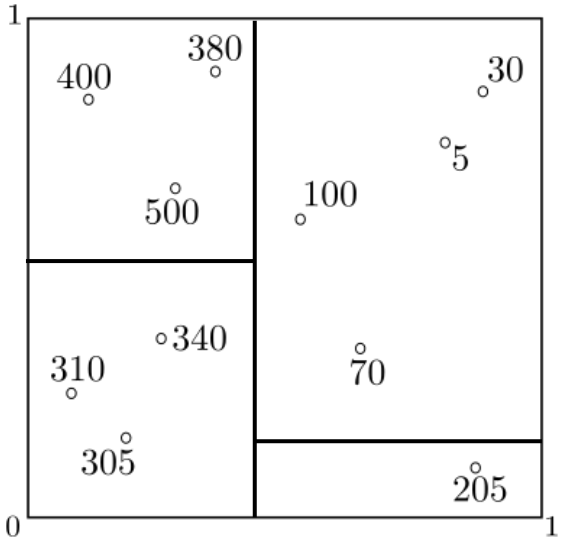
$k = 1$



Building a decision tree

Procedure

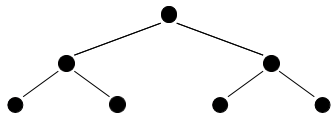
- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied



$k = 0$

$k = 1$

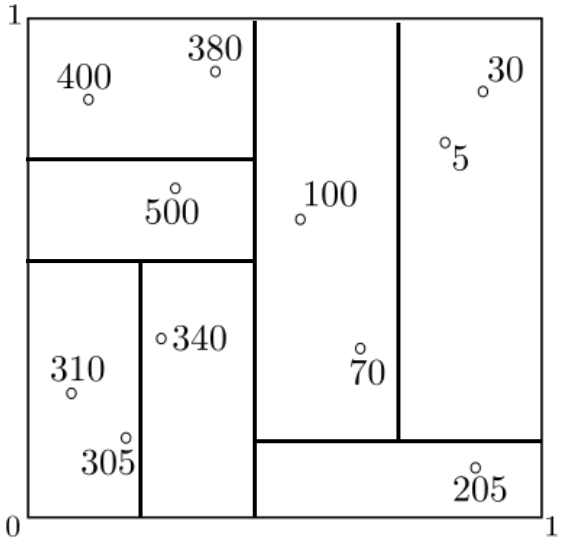
$k = 2$



Building a decision tree

Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied

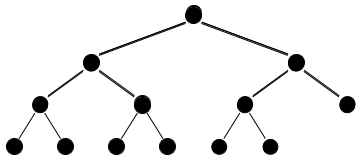


$k = 0$

$k = 1$

$k = 2$

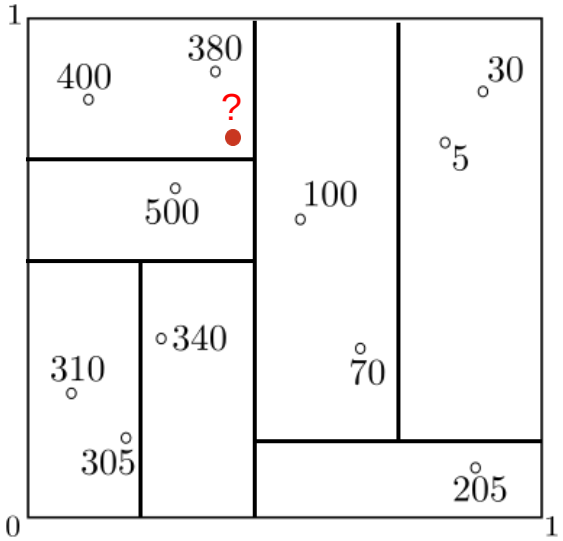
$k = 3$



Building a decision tree

Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied

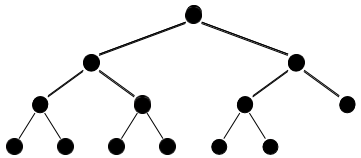


$k = 0$

$k = 1$

$k = 2$

$k = 3$

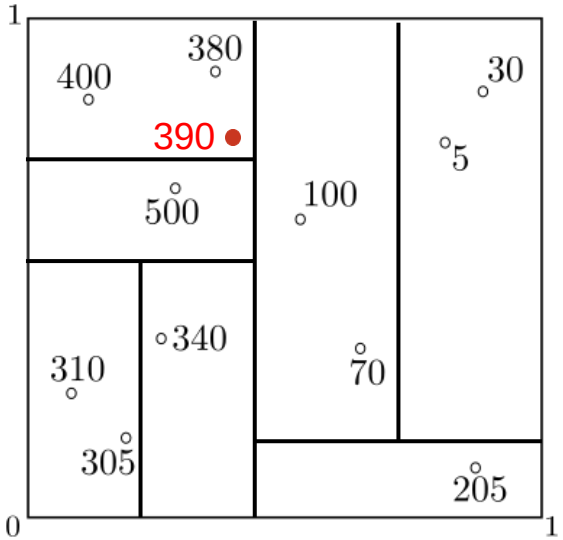


How to predict a new entry $Y|X$?

Building a decision tree

Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied

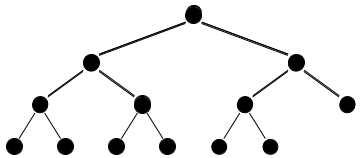


$k = 0$

$k = 1$

$k = 2$

$k = 3$

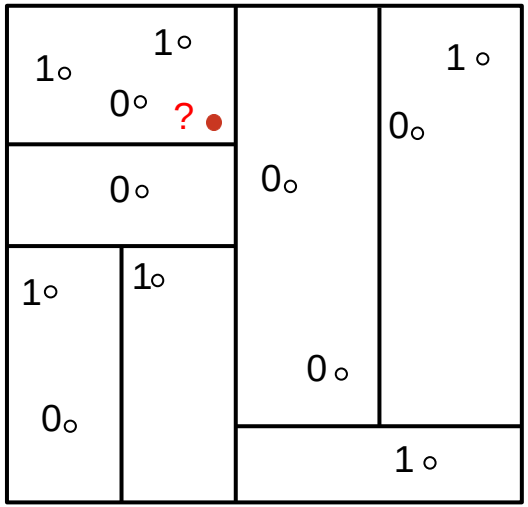


By averaging the training set values gathered in the tree leaf

Building a decision tree

Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied

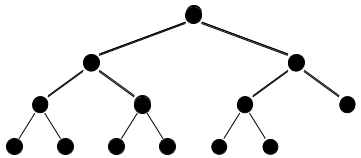


$k = 0$

$k = 1$

$k = 2$

$k = 3$

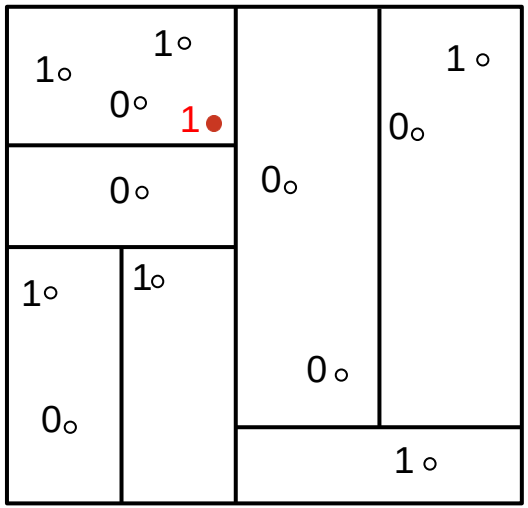


How about classification ?

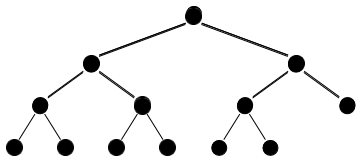
Building a decision tree

Procedure

- Trees are built recursively by splitting the current cell into two children until some stopping criterion is satisfied



$k = 0$
$k = 1$
$k = 2$
$k = 3$



Hard voting system

(can also use soft voting)

Building a decision tree

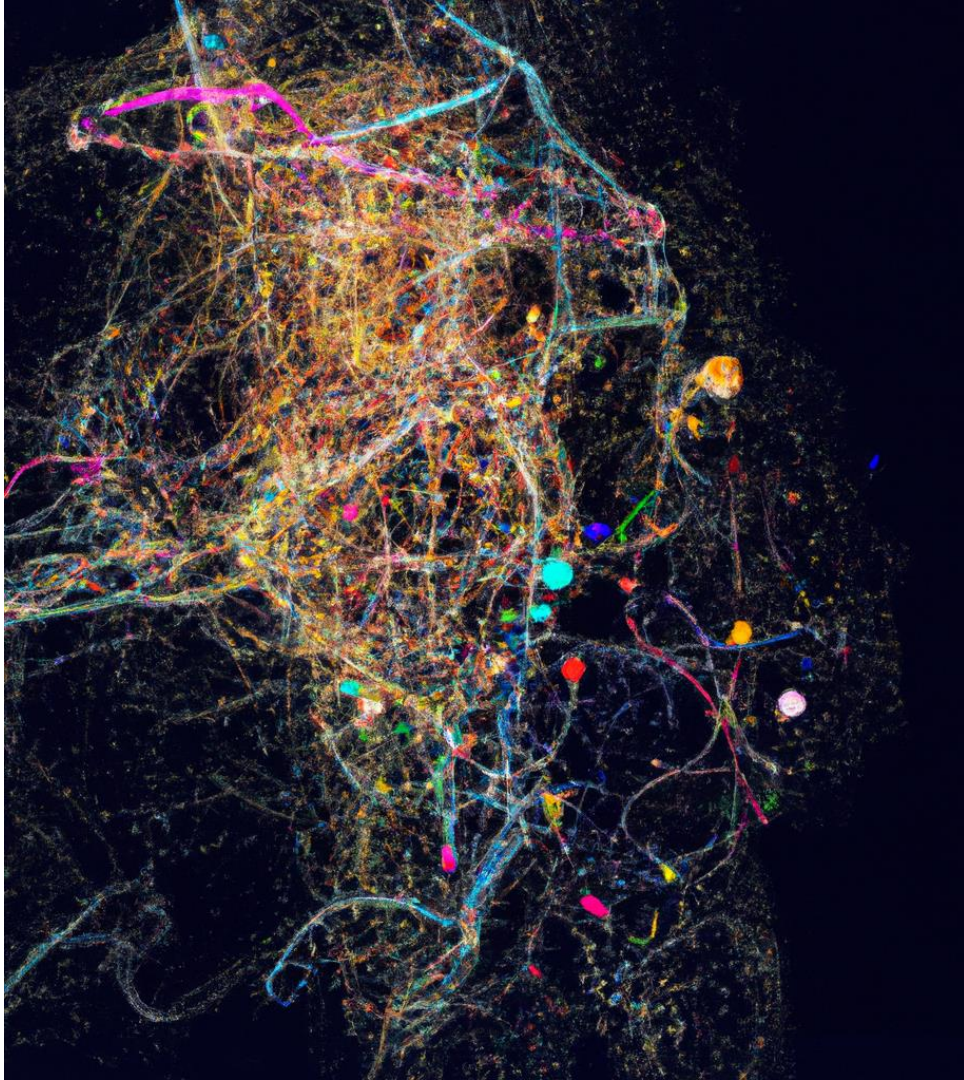
Criteria
- Gini or Entropy – classification
- Variance reduction – regression

Advantages
- Easily interpretable
- Can handle numerical and categorical data
- Does not require scaling of features
- Can approximate any Boolean function (including XOR)

Drawbacks
- Overfitting
- Learning the optimal decision tree is NP-complete

Bagging

Bootstrap Aggregation



From a tree to a bag of trees

Bagging

- Decision trees models are highly interpretable and fast to train
- However, in order to capture a complex decision boundary (or to approximate a complex function), we need to use a deep tree (since each time we can only make axis aligned splits)
- Large trees have high variance and are prone to overfitting.
- For these reasons, in practice, decision tree models often underperforms when compared with other classification or regression methods.

From a tree to a bag of trees

Bagging

- One way to adjust for the high variance of the output of an experiment is to perform the experiment multiple times and then average the results
- The same idea can be applied to high variance models:

1. (Bootstrap) we generate multiple samples of training data, via bootstrapping. We train a full decision tree on each data sample
2. (Aggregate) for a given input, we output the averaged outputs of all the models for that input

- For classification, we return the class that is outputted by the plurality of the models.
- This method is called Bagging (Breiman, 1996), short for, **B**ootstrap **Agg**regat**ing**

Bootstrap Aggregation

Bagging

- One way to adjust for the high variance of the output of an experiment is to perform the experiment multiple times and then average the results
- The same idea can be applied to high variance models:

1. (Bootstrap) we generate multiple samples of training data, via bootstrapping. We train a full decision tree on each data sample
2. (Aggregate) for a given input, we output the averaged outputs of all the models for that input

- For classification, we return the class that is outputted by the plurality of the models.
- This method is called Bagging (Breiman, 1996), short for, **B**ootstrap **Agg**regat**ing**

Bootstrap Aggregation

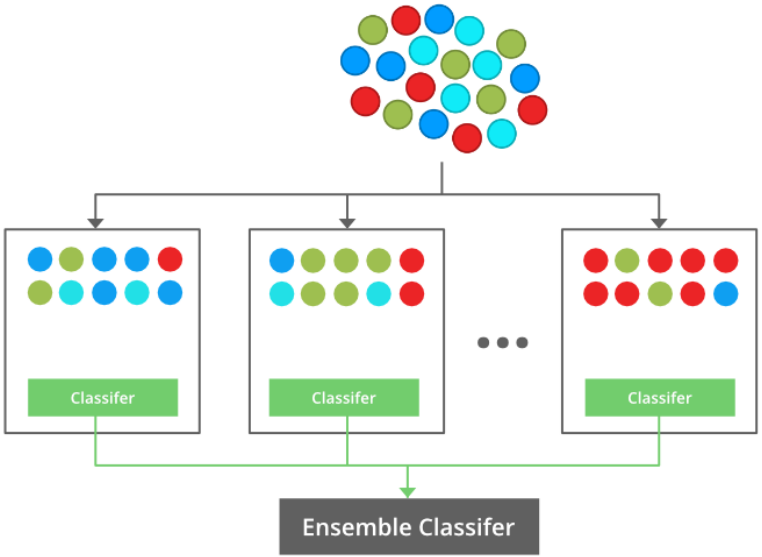
Bagging

Note that bagging enjoys the benefits of

1. High expressiveness - by using full trees each model is able to approximate complex functions and decision boundaries.
2. Low variance - averaging the prediction of all the models reduces the variance in the final prediction, assuming that we choose a sufficiently large number of trees

Bootstrap Aggregation

Bagging



Original Data

Bootstrapping

Aggregating

Bagging

Interpretability issues

However, the major drawback of bagging
(and other ensemble methods that we will study) is that the averaged model is no longer easily interpretable - i.e. one can no longer trace the 'logic' of an output through a series of decisions based on predictor values!

Interpretability of Bagging

The major drawback of bagging

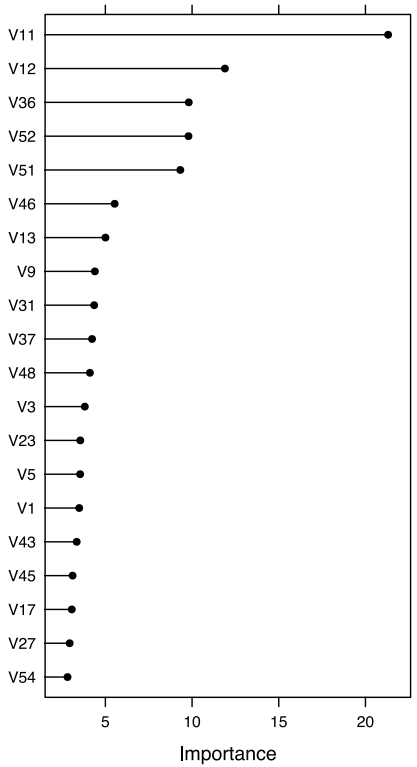
The averaged model is no longer easily interpretable - i.e. one can no longer trace the 'logic' of an output through a series of decisions based on predictor values

Variable importance

- Calculate the total amount that the RSS (for regression) or Gini index (for classification) is decreased due to splits over a given predictor, averaged over all trees.
- A large value indicates an important predictor.

Variable importance

Bagging



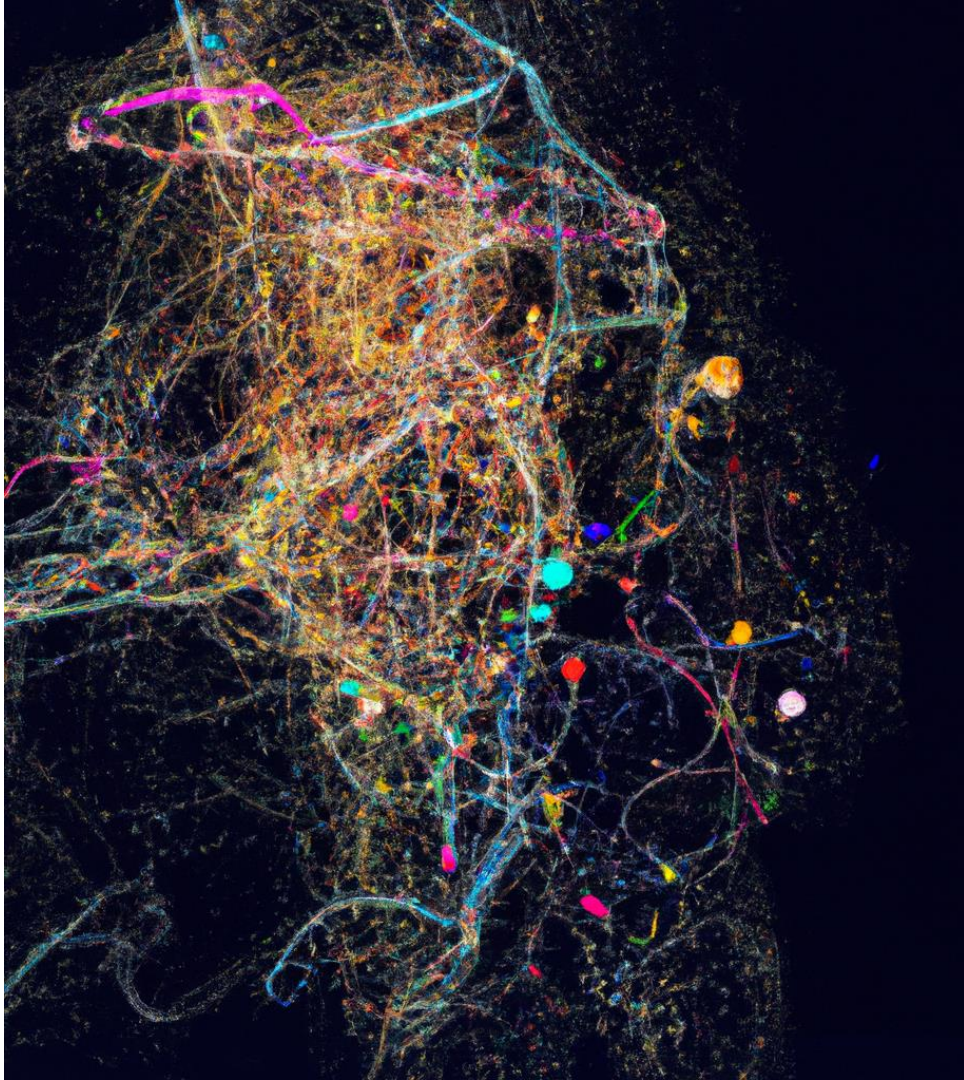
Out-of-Bag Error

- Bagging is an ensemble method ⇒ a single model is produced by training and aggregating multiple models
- With ensemble methods, we can use a new metric for assessing the predictive performance of the model⇒ the out-of-bag error
- Given a training set and an ensemble of models each trained on a bootstrap sample, we compute the out-of-bag error of the averaged model by the following procedure

1. For each point in the training set, we average the predicted output for this point over the models whose bootstrap training set excludes this point. We compute the squared error of this averaged prediction. Call this the point-wise out-of-bag error.
2. We average the point-wise out-of-bag error over the full training set.

Random Forests

The wisdom of the crowd



Random Forest Algorithm

Creation

- Random forests were created by Leo Breiman [2001]

Theoretical results

- Many theoretical results focus on simplified version on random forests, whose construction is independent of the dataset [Biau et al., 2008, Biau, 2012, Genuer, 2012, Zhu et al., 2012, Arlot and Genuer, 2014]

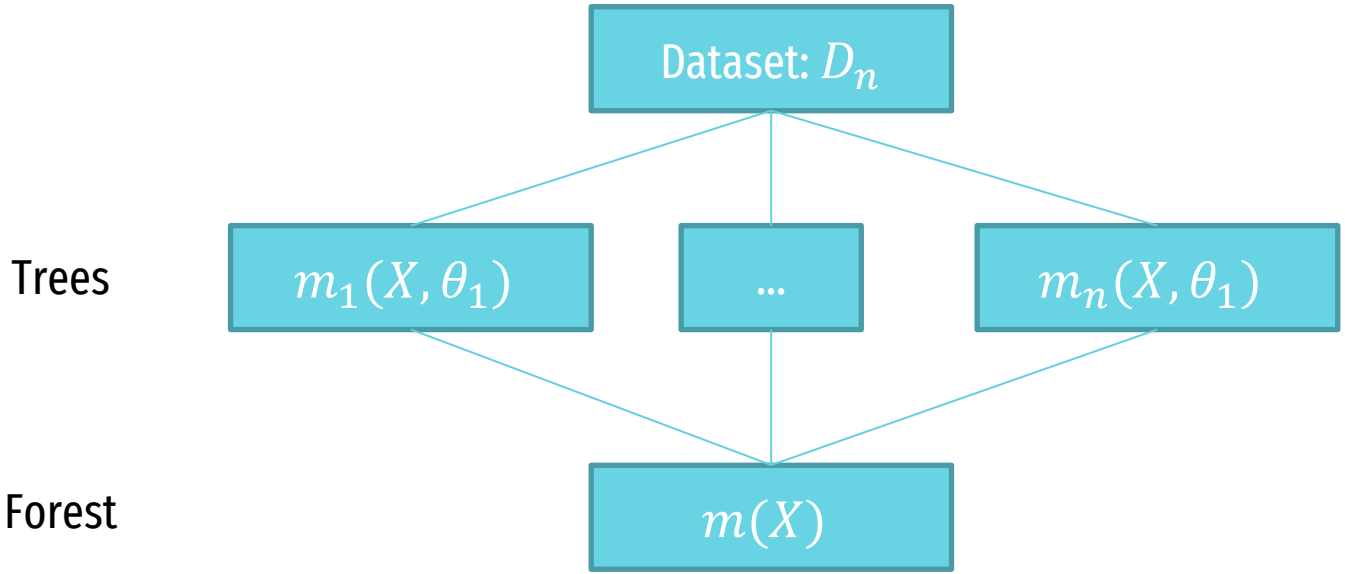
Literature review

- Methodological review [Criminisi et al., 2011, Boulesteix et al., 2012]
- Theoretical review [Biau and Scornet, 2016]

What is a Forest ?

Ensemble learning

- Using multiple trees trained in parallel



Improving on Bagging

Correlated predictors

- In practice, the ensembles of trees in Bagging tend to be highly correlated
- Suppose we have an extremely strong predictor, x_j , in the training set amongst moderate predictors
- The greedy learning algorithm ensures that most of the
- models in the ensemble will choose to split on x_j in early iterations
- That is, each tree in the ensemble is identically distributed, with the expected output of the averaged model the same as the expected output of any one of the trees.

Improving on Bagging

The wisdom of the forest

- Random Forest is a modified form of bagging that creates ensembles of “independent” decision trees.
- To de-correlate the trees, we:

1. train each tree on a separate bootstrap sample of the full training set (same as in bagging)
2. for each tree, at each split, we randomly select a set of J' predictors from the full set of predictors.

- From the J′ predictors, we select the optimal predictor and the optimal corresponding threshold for the split.

Tuning Random Forests

Different Hyperparameters

- Random forest models have multiple hyperparameters to tune:

1. the number of predictors to randomly select at each split
2. the total number of trees in the ensemble
3. the minimum leaf node size

- In theory, each tree in the random forest is full, but in practice this can be computationally expensive (and added redundancies in the model), thus, imposing a minimum node size is not unusual.

Tuning Random Forests

Validation

- There are standard (default) values for each of random forest hyper-parameters recommended by long time practitioners, but generally these parameters should be tuned through cross validation (making them data and problem dependent).
- Using out-of-bag errors, training and cross validation can be done in a single sequence - we cease training once the out-of-bag error stabilizes

Variable importance

Using OOB error

- Record the prediction accuracy on the oob samples for each tree
- Randomly permute the data for column j in the oob samples then record the accuracy again
- The decrease in accuracy as a result of this permuting is averaged over all trees, and is used as a measure of the importance of variable j in the random forest

Final Comments

Irrelevant predictors

- When the number of predictors is large, but the number of relevant predictors is small, random forests can perform poorly
- In each split, the chances of selecting a relevant predictor will be low and hence most trees in the ensemble will be weak models

Number of trees

- Increasing the number of trees in the ensemble generally does not increase the risk of overfitting.
- Again, by decomposing the generalization error in terms of bias and variance, we see that increasing the number of trees produces a model that is at least as robust as a single tree.
- However, if the number of trees is too large, then the trees in the ensemble may become more correlated, increase the variance.

References

- Erwan Scornet – A walk in Random Forest
- Pavlos Protopapas, Kevin Rader, Rahul Dave Margo Levine - Regression Trees & Random Forests