LECTURER: TAI LE QUY

MACHINE LEARNING – SUPERVISED LEARNNG

UNIT 5.2

ENSEMBLE METHODS

STUDY GOALS

- explain the power of the ensemble methods widely used in practice.
- define bagging and boosting.
- apply two very popular ensemble models on your own with the use of Python.

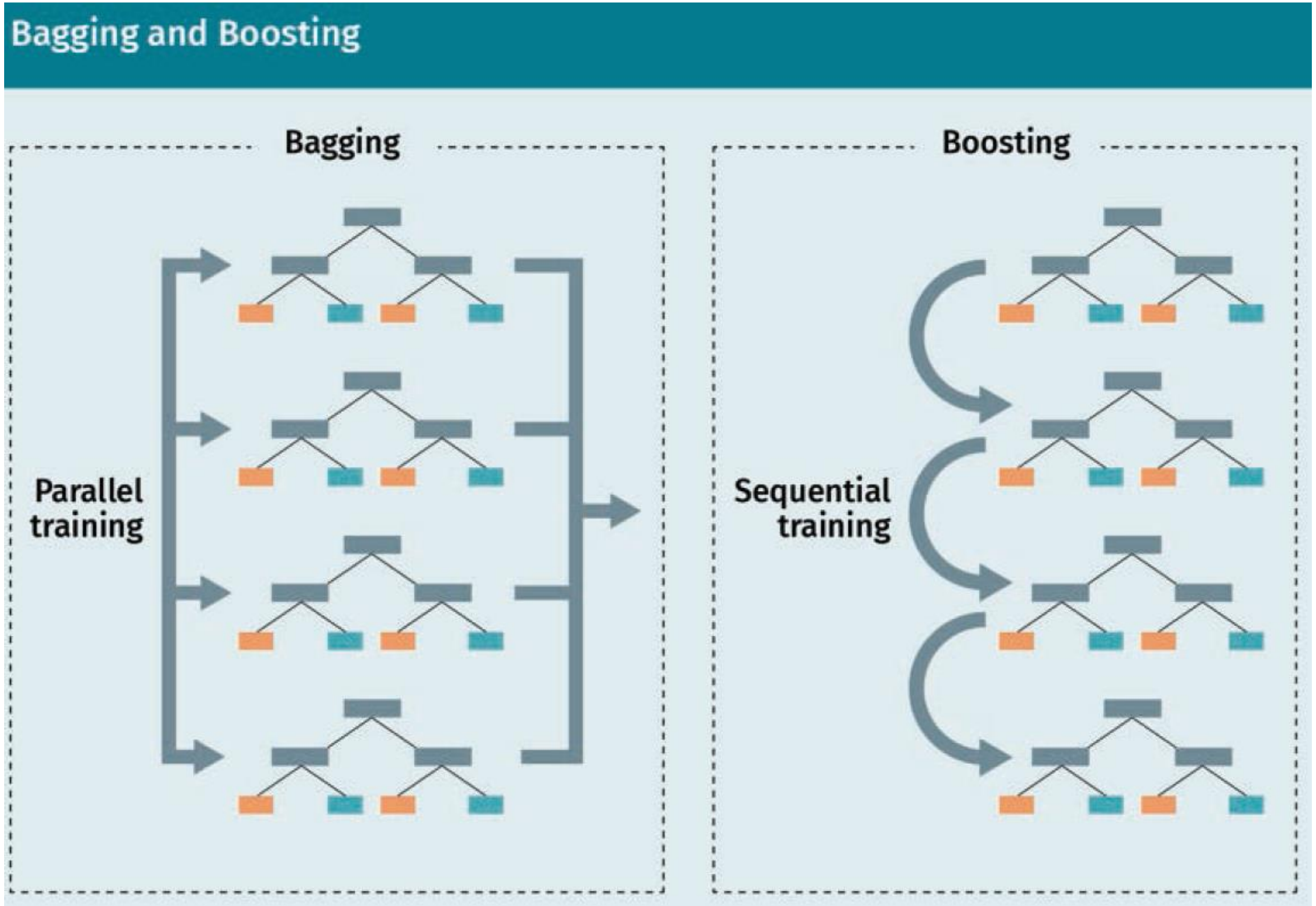
EXPLAIN SIMPLY

1. How do ensemble methods work and how do they combine large numbers of trees to make one strong estimator?
2. How can we apply random forests and gradient boosting, two very well-known ensemble methods, using Python?

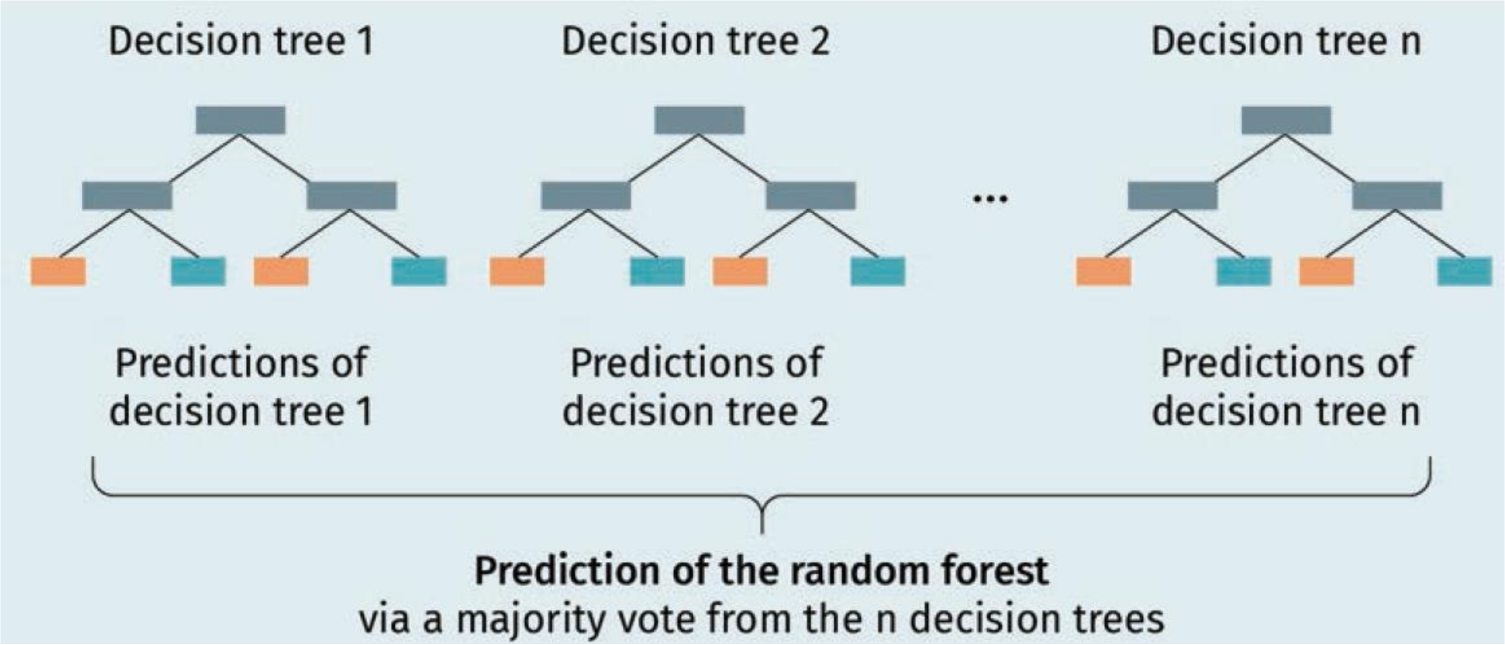
ENSEMBLE METHODS

- Several trees to be bundled together to form one strong estimator
- Ensemble methods can be divided into two categories:
  - Bagging algorithms: individual decision trees are independently trained in parallel
  - Boosting algorithms: decision trees are trained sequentially, and one tree takes the errors of
  - the previously constructed tree into consideration
- The most well-known representatives of bagging and boosting are:
  - Random forest
  - Gradient boosting

ENSEMBLE METHODS

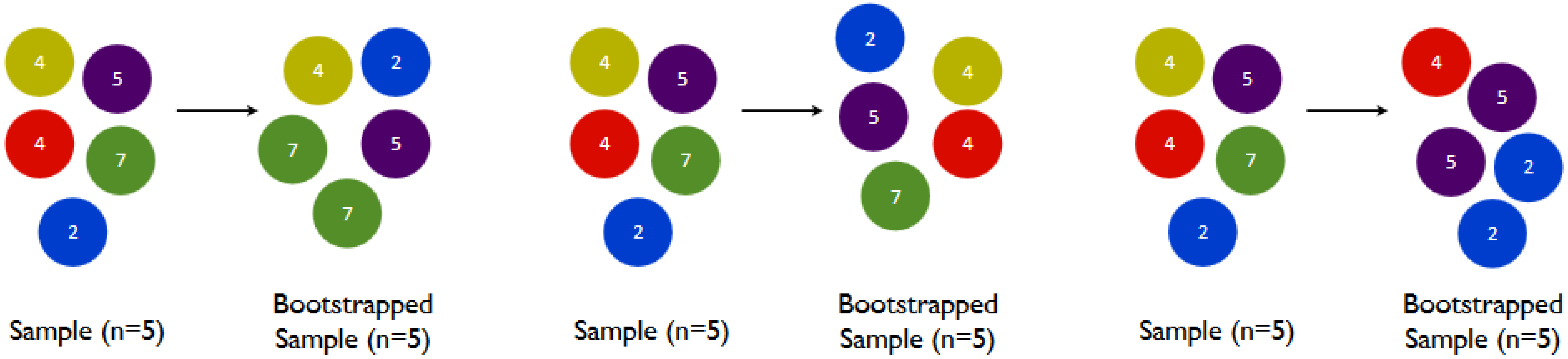


- The random forest (RF) is a high performing algorithm that bundles single decision trees into one strong estimator through bagging
- The final prediction of a RF is then determined by aggregating the predictions of the individual decision trees
  - Combining them via a majority vote



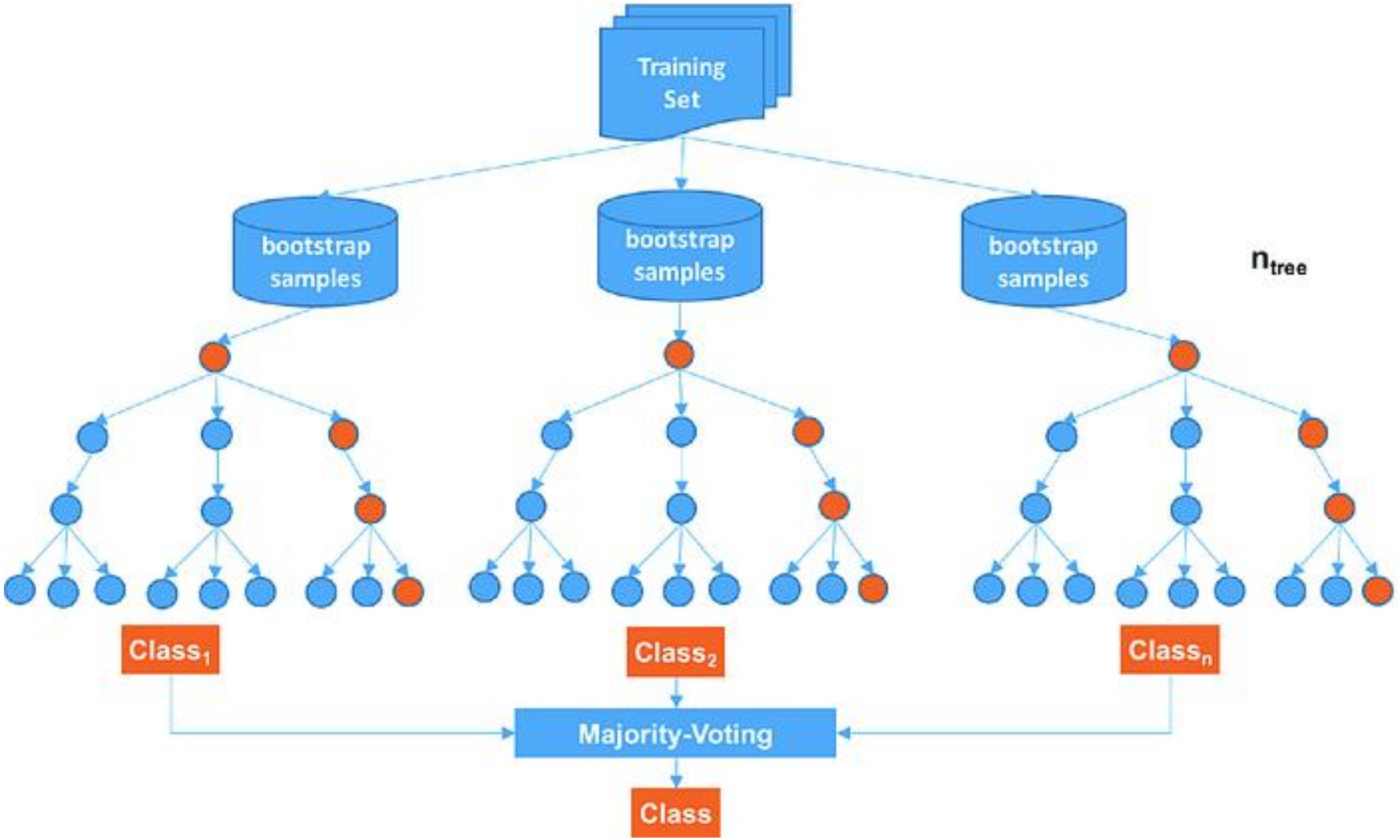
Source of image: Course book

- The individual decision trees are constructed independently of each other with the introduction of a random component
- This is done by building the trees on **bootstrap** copies of the training data
  - This procedure resamples a dataset to create many simulated samples of the same size by drawing randomly with laying back



Source of image: https://stats.stackexchange.com/questions/246726/size-of-bootstrap-samples

RANDOM FOREST

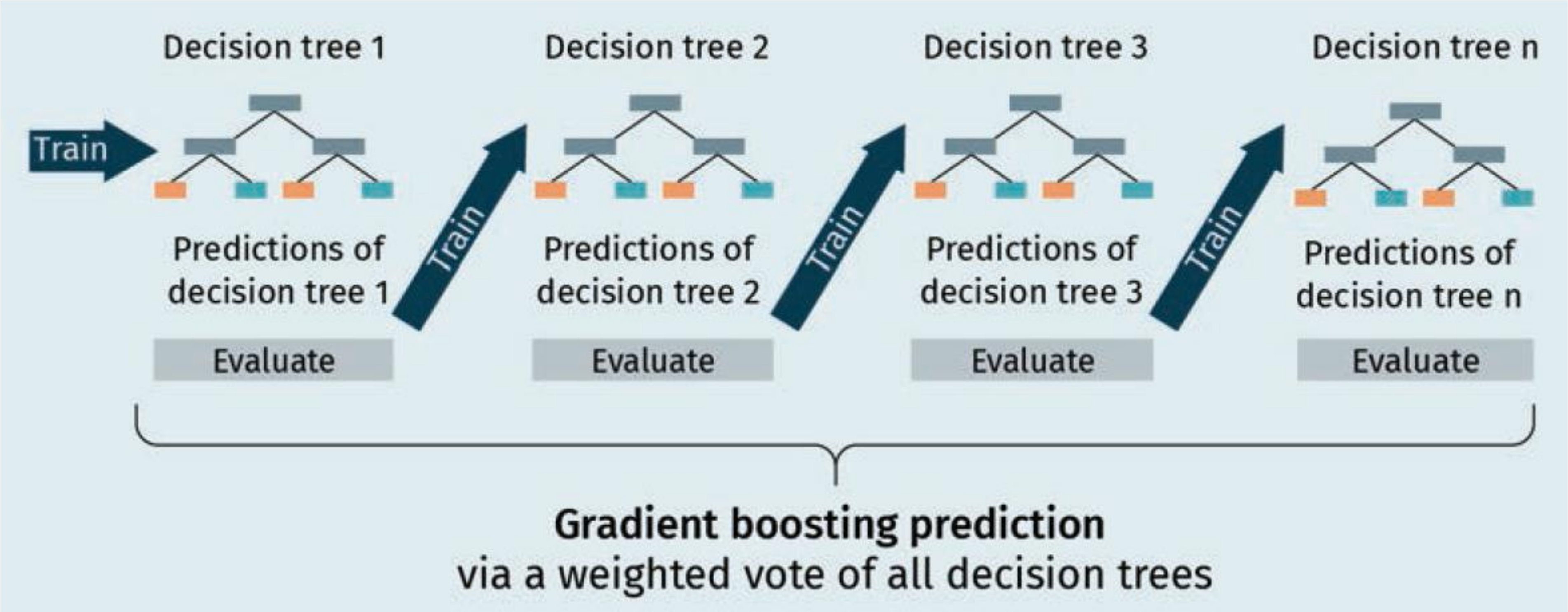


Source of image: https://medium.com/@uma.bollikonda/random-forest-machine-learning-algorithm-401bdcd7a0b8

- Select the number of trees to construct (hyperparameter “**n_trees**”)
- For all number of trees:
  - Generate a bootstrap sample of the original data.
  - Grow a regression/classification tree based on the bootstrapped data.
  - For each node/split, perform the following set of actions:
    - Select a number “**m_try**” of features at random from all p features.
    - Pick the best feature/split-point among the “**m_try**” tested features.
    - Split the node into two child nodes.
  - Use common tree model stop criteria to determine when a tree is completed and unpruned.
  - Output the ensemble of trees.
  - Let each of the trees make a prediction.
  - Use the individual predictions in a voting process in which the final prediction is determined.

- Gradient boosting uses a large number of individual weak estimators (usually decision trees) that are combined into one strong estimator.
- The individual estimators are not trained in parallel and independently of each other
- They are trained sequentially, and one estimator helps optimize the next
- The misclassified samples in one iteration are exaggerated in the following iteration, thereby placing emphasis on the need to classify these samples correctly in the next decision tree to be trained

1. A loss function we want to optimize
2. A set of weak learners generating predictions
3. An additive model for combining the predictions of the weak learners into one strong predictor (thereby minimizing the loss function)



Source of the image: Course book

- Select the number of trees to construct (hyperparameter “**n_trees**”).
- Fit the first weak estimator to the given training data X and generate predictions ŷ, i.e., $F_1(x)=\hat{y}$
- Fit the next weak estimator to the residuals of the previous one, i.e, $h_1(x)=y-F_1(x)$
- Add this weak estimator to the model, i.e., $F_2(x)=F_1(x)+h_1(x)$
- Fit the next weak estimator to the residuals of F_2: $h_2(x)=y-F_2(x)$
- Add this weak estimator to the model, i.e., $F_3(x)=F_2(x)+h_2(x)$
- Continue this process until the number of prospective trees has been reached.

The resulting strong estimator can be mathematically expressed as the additive combination of the single *i* weak estimators of number *n*:

$$f(x) = \sum_{i=1}^{n} f^i(\mathrm{x})$$

STUDY GOALS REVIEW

- explain the power of the ensemble methods widely used in practice.
- define bagging and boosting.
- apply two very popular ensemble models on your own with the use of Python.

UNIT 5.2

TRANSFER TASK

Credit Score Classification: Case Study

– The **credit score** of a person determines the creditworthiness of the person. It helps financial companies determine if you can repay the loan or credit you are applying for.

– Explain and describe how ensemble methods might be applied.

TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your results.
The results will be discussed in plenary.

How did you like the course?