

SELECTED TOPICS IN ENGINEERING

INTR. TO PROG. FOR DATA SCIENCE
ENGR 350

Tuesday–Thursday 10:00–12:45
ENG B05

Dr. Banu Yobaş

Decision Trees (DTs)

- ▶ are a non-parametric supervised learning method used for classification and regression.
- ▶ CART is an acronym introduced by Leo Breiman to refer to Decision Tree algorithms that can be used for classification or regression predictive modeling problems: Classification and Regression Trees (CART)
- ▶ The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.



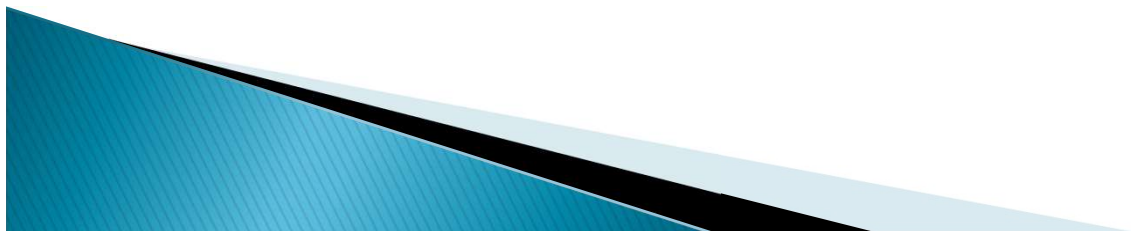
User parameters

- ▶ **Maximum Tree Depth.** This is the maximum number of nodes from the root node of the tree. Once a maximum depth of the tree is met, we must stop splitting adding new nodes. Deeper trees are more complex and are more likely to overfit the training data.
- ▶ **Minimum Node Records.** This is the minimum number of training patterns that a given node is responsible for. Once at or below this minimum, we must stop splitting and adding new nodes. Nodes that account for too few training patterns are expected to be too specific and are likely to overfit the training data



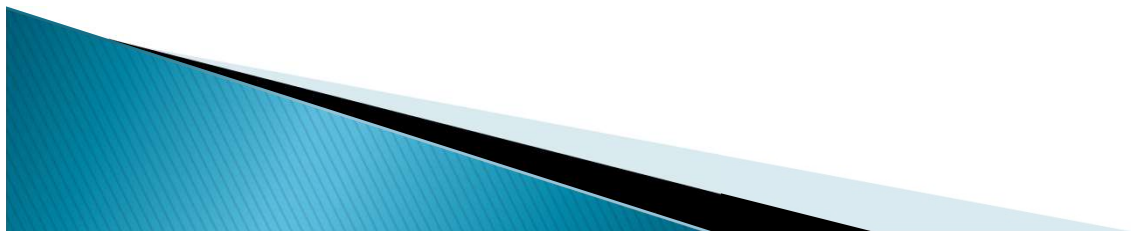
When to stop growing the tree?

- ▶ There is one more condition. It is possible to choose a split in which all rows belong to one group.
- ▶ In this case, we will be unable to continue splitting and adding child nodes as we will have no records to split on one side or another.
- ▶ When we do stop growing at a given point, that node is called a terminal node and is used to make a final prediction.



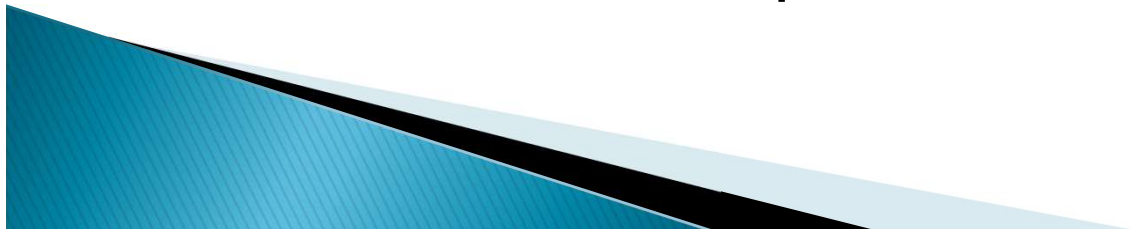
CART

- ▶ Creating a binary decision tree is actually a process of dividing up the input space.
- ▶ A greedy approach is used to divide the space called recursive binary splitting.
- ▶ This is a numerical procedure where all the values are lined up and different split points are tried and tested using a cost function.



Constructing a tree

- ▶ **Regression:** The cost function that is minimized to choose split points is the sum squared error across all training samples that fall within the rectangle.
- ▶ **Classification:** The Gini cost function is used which provides an indication of how pure the nodes are, where node purity refers to how mixed the training data assigned to each node is.
- ▶ Splitting continues until nodes contain a minimum number of training examples or a maximum tree depth is reached.



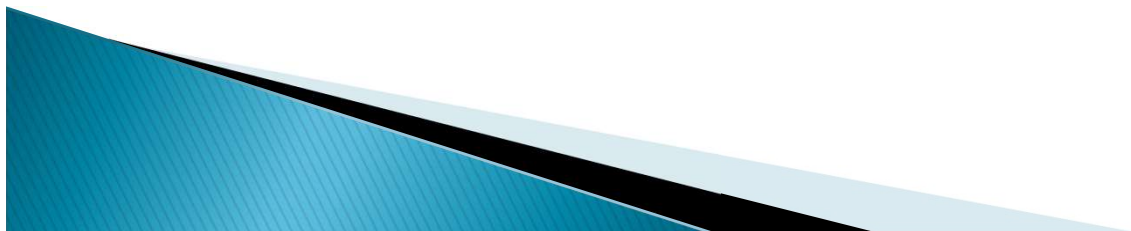
Gini index

- ▶ is the name of the cost function used to evaluate splits in the dataset.
- ▶ A Gini score gives an idea of how good a split is by how mixed the classes are in the two groups created by the split.
- ▶ A perfect separation results in a Gini score of 0,
- ▶ the worst case split that results in 50/50 classes in each group result in a Gini score of 0.5 (for a 2 class problem).



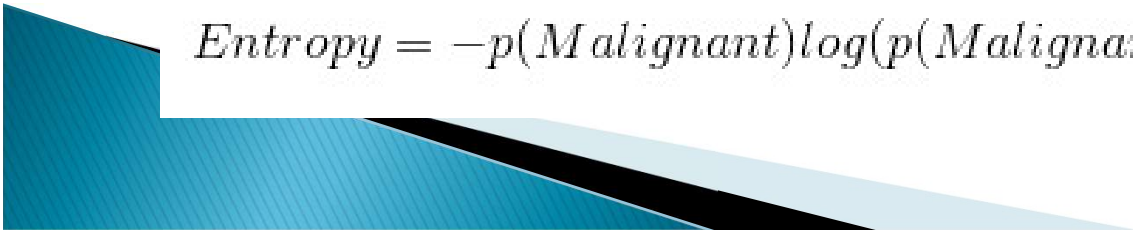
Decision Tree

- ▶ splits the data variable by variable starting with the variable with the highest predictive power, less impurity and lower entropy
- ▶ The main aim of this method is to minimize impurity in each node.
- ▶ Impurity of nodes is calculated by the entropy of data in the node.



Entropy

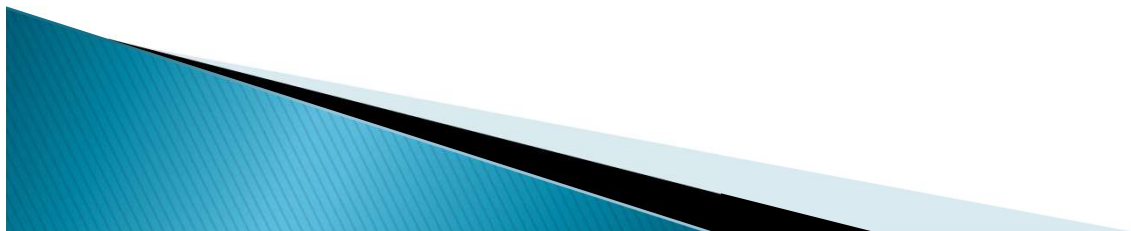
- ▶ is the amount of information disorder or
- ▶ the amount of randomness in the data or
- ▶ uncertainty.
- ▶ the lower the entropy the less uniform the distribution and the purer the node.
- ▶ If a sample is completely homogenous then the entropy is completely zero and
- ▶ if a sample is equally divided it has an entropy of 1.


$$Entropy = -p(Malignant)\log(p(Malignant)) - p(Benigh)\log(p(Benigh))$$

Information gain

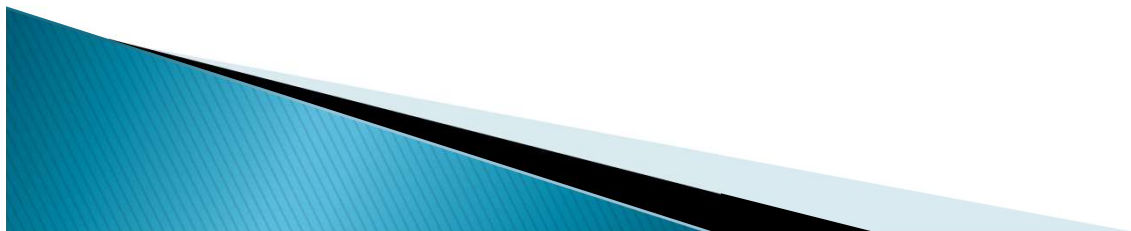
- ▶ the information that can increase the level of certainty after splitting
- ▶ uses information-theory-based entropy to indicate how much extra information is gained by the decision node

IG = Entropy of the tree before the split – weighted entropy after the split.



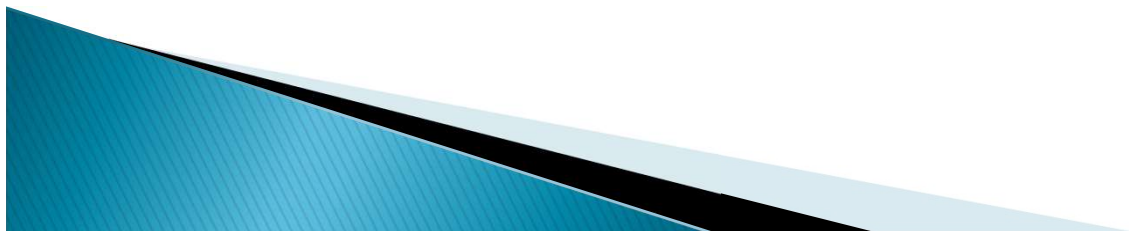
Some advantages of decision trees

- ▶ Simple to understand and to interpret. Trees can be visualised.
- ▶ Requires little data preparation. Other techniques often require data normalisation, dummy variables need to be created and blank values to be removed. Note however that this module does not support missing values.
- ▶ The cost of using the tree (i.e., predicting data) is logarithmic in the number of data points used to train the tree.
- ▶ Able to handle both numerical and categorical data. Other techniques are usually specialised in analysing datasets that have only one type of variable.



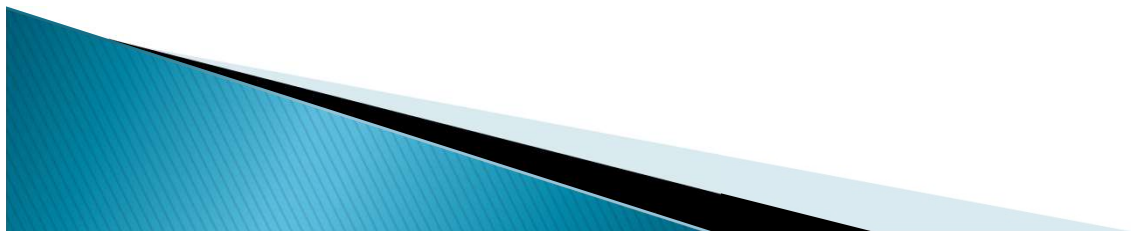
Some advantages of decision trees

- ▶ Able to handle multi-output problems.
- ▶ Uses a white box model. If a given situation is observable in a model, the explanation for the condition is easily explained by boolean logic. By contrast, in a black box model (e.g., in an artificial neural network), results may be more difficult to interpret.
- ▶ Possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model.
- ▶ Performs well even if its assumptions are somewhat violated by the true model from which the data were generated.



The disadvantages of DT

- ▶ Decision-tree learners can create over-complex trees that do not generalise the data well. This is called **overfitting**. Mechanisms such as pruning (not currently supported), setting the minimum number of samples required at a leaf node or setting the maximum depth of the tree are necessary to avoid this problem.
- ▶ Decision trees can be unstable because small variations in the data might result in a completely different tree being generated. This problem is mitigated by using decision trees within an ensemble.



The disadvantages of DT

- ▶ The problem of learning an optimal decision tree is known to be NP-complete under several aspects of optimality and even for simple concepts. Consequently, practical decision-tree learning algorithms are based on heuristic algorithms such as the greedy algorithm where locally optimal decisions are made at each node. Such algorithms cannot guarantee to return the globally optimal decision tree. This can be mitigated by training multiple trees in an ensemble learner, where the features and samples are randomly sampled with replacement.
- ▶ There are concepts that are hard to learn because decision trees do not express them easily, such as XOR, parity or multiplexer problems.
- ▶ Decision tree learners create *biased trees* if some classes dominate. It is therefore recommended to balance the dataset prior to fitting with the decision tree.



Decision_Tree_Classifier

```
from sklearn.tree import DecisionTreeClassifier

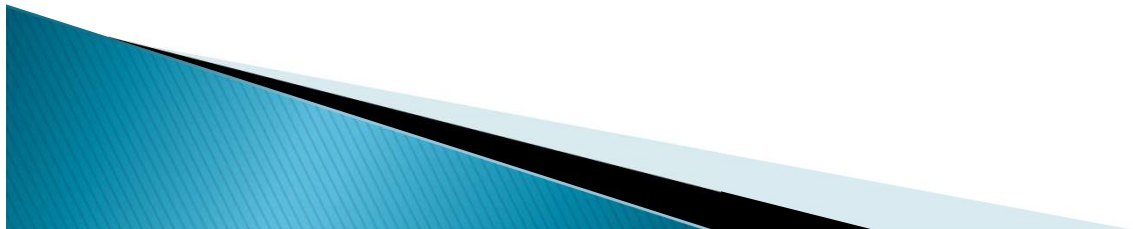
classifierx= DecisionTreeClassifier(...parameters )
#see documentation in sklearn

type(classifierx)

.classifierx

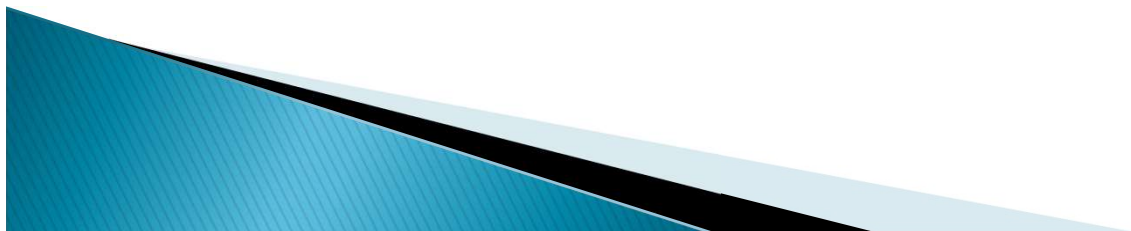
classifierx.fit(X_trainset,y_trainset)

classifierx.predict(X_test)
```



Decision_Tree_Classifier

- ▶ The decision estimator has an attribute called `tree_` which stores the entire tree structure and allows access to low level attributes.
- ▶ The binary tree `tree_` is represented as a number of parallel arrays.
- ▶ The i -th element of each array holds information about the node ``i``.
- ▶ Node 0 is the tree's root.



Decision Trees

- ▶ Regarding iris dataset a preprocessed data set is also available
- ▶ The classes of flowers are encoded as 0,1,2
- ▶ You can alternatively import the dataset with the following commands using sklearn module:

```
>>> from sklearn import datasets  
>>> iris = datasets.load_iris()
```

