cct | College Dublin
Computing • IT • Business

Machine Learning for Data Analysis
MSc in Data Analytics
CCT College Dublin

Classification using Decision Trees and RF
Week 2

Lecturer: Dr. Muhammad Iqbal*

Email: miqbal@cct.ie

Agenda

cct College Dublin
Computing • IT • Business

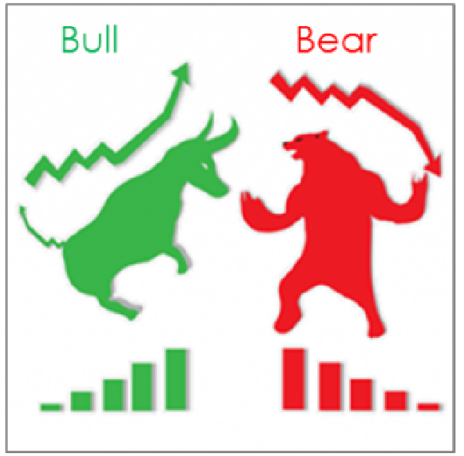
- Classification and Regression
- Classification Methods and Examples
- Structure of Hunt's Algorithm
- Measures of Node Impurity
- Calculation of Gini Index and Error
- Comparison of Impurity Measures
- Misclassification Error vs Gini Index
- Advantages and Disadvantages of Decision Tree Based Classification
- Ensemble Classifiers
- Random Forest and Algorithm

Classification and Regression

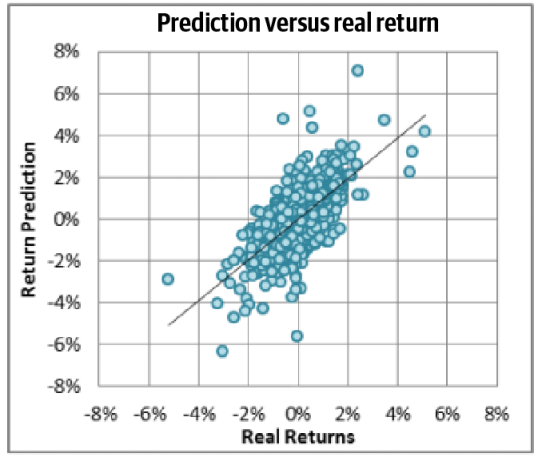
cct College Dublin
Computing • IT • Business

- **Classification**
- Classification is a subcategory of supervised learning in which the goal is to predict the categorical class labels of new instances based on past observations.
- **Regression**
- Regression is another subcategory of supervised learning used in the prediction of continuous outcomes. In regression, we are given a number of predictor (explanatory) variables and a continuous response variable (outcome or target), and we try to find a relationship between those variables that allows us to predict an outcome.

Classification



Regression



Classification Methods

cct College Dublin
Computing • IT • Business

- **Given a collection of records (training set)**
  - Each record is characterized by a tuple (***x***, *y*), where ***x*** is the attribute set and *y* is the class label
    - ***x***: attribute, predictor, independent variable, input
    - *y*: class, response, dependent variable, output
  - **Base Classifiers**
    - **Decision Tree based Methods**
    - Rule-based Methods
    - Nearest-neighbor
    - Neural Networks
    - Naïve Bayes and Bayesian Belief Networks
    - Support Vector Machines
  - **Ensemble Classifiers**
    - Boosting, Bagging, Random Forests

Task:

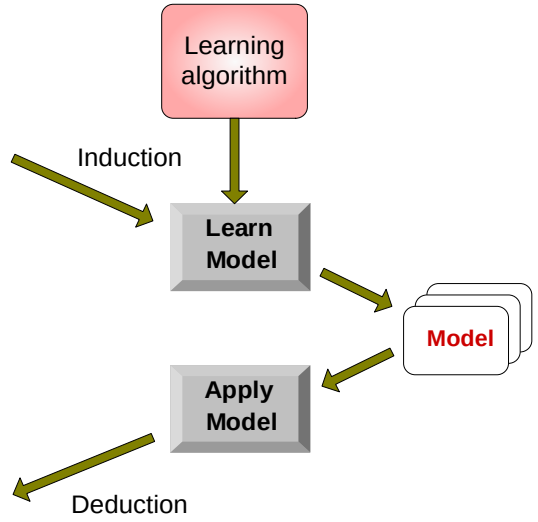
- Learn a model that maps each attribute set ***x*** into one of the predefined class labels *y.*

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



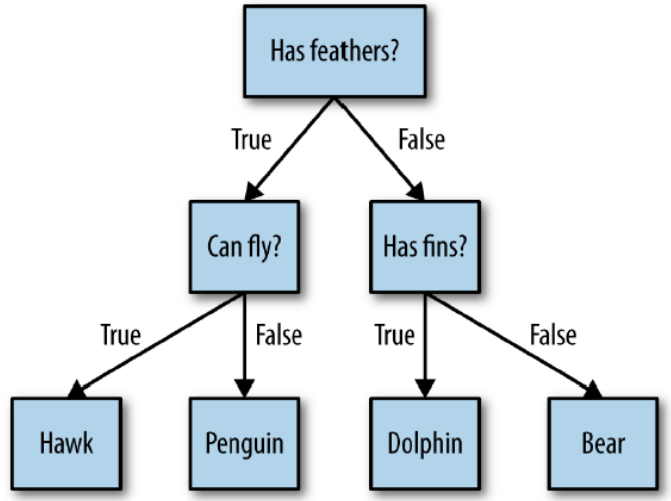
Classification

Example




Decision trees are widely used models for classification and regression tasks. They learn a hierarchy of if/else questions, leading to a decision.

- These questions are similar to the questions that you might ask in a game of 20 Questions.
- Suppose you want to distinguish between the following four animals: **bears, hawks, penguins, and dolphins.**
- Your goal is to get to the right answer by asking as few if/else questions as possible. You might start off by asking whether the animal has feathers, a question that narrows down your possible animals to just two.
- If the answer is "**yes**," you can ask another question that could help you to distinguish between hawks and penguins.
  - For example, you could ask whether the animal can fly. If the animal doesn't have feathers, your possible animal choices are dolphins and bears, and you will need to ask a question to distinguish between these two animals. For example, asking whether the animal has fins.



A decision tree to distinguish among several animals

- In this illustration, each node in the tree either represents a question or a terminal node (also called a leaf) that contains the answer.
- The edges connect the answers to a question with the next question you would ask.

Decision Trees

Examples

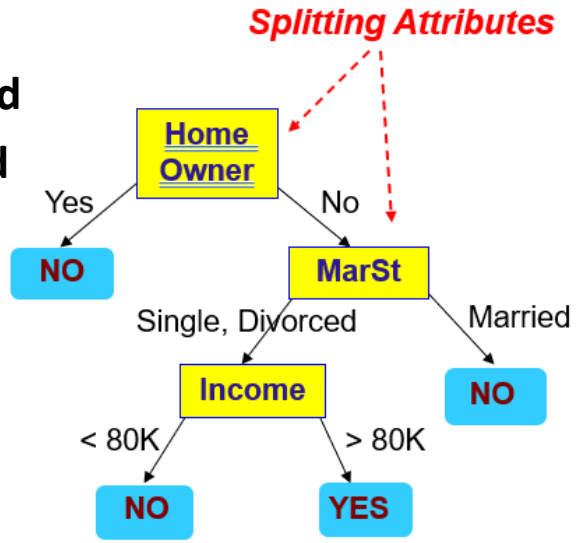
Model: Decision Tree

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
		categorical	categorical	continuous	class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

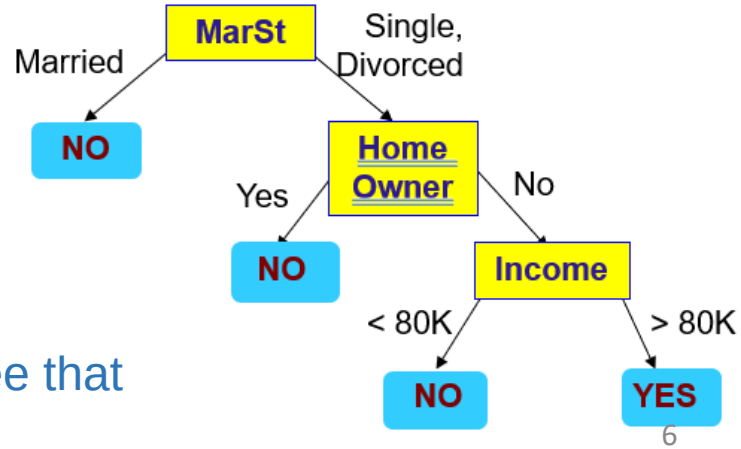
Training Data

- **Can you determine the chances of new customer's default or not based on Home Owner, Martial Status and Annual Income features?**

First possible Decision Tree



2nd possible Decision Tree

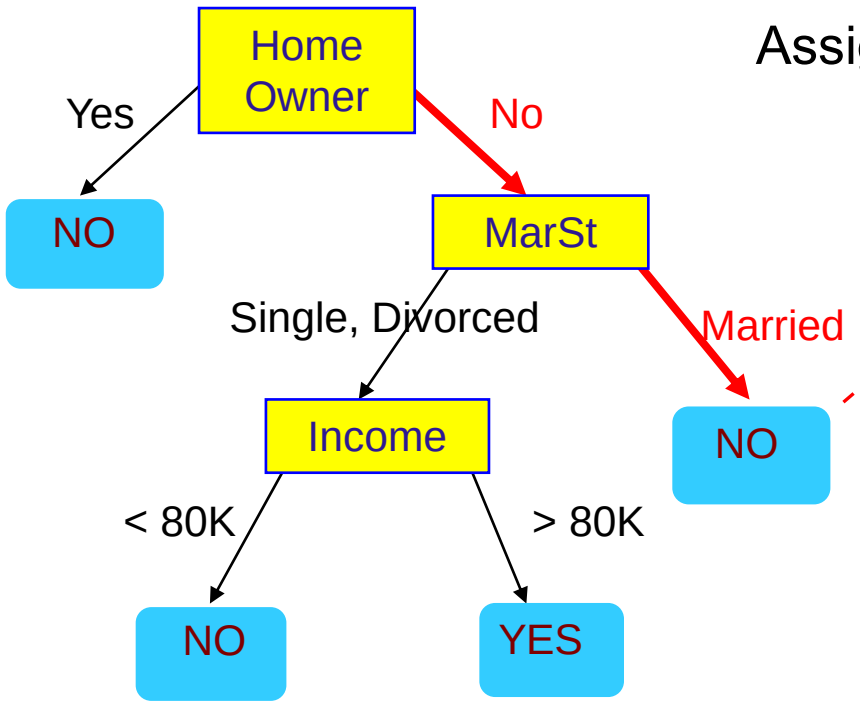


There could be more than one tree that fits the same data!

Apply Model to Test Data

Test Data

Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?



Assign Defaulted to "No"

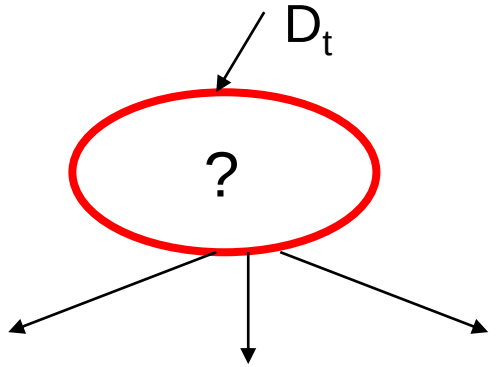
- **Many Algorithms:**
  - Hunt's Algorithm (one of the earliest)
  - CART (Classification and Regression Trees) is similar to C4.5.
  - ID3, C4.5
  - SLIQ, SPRINT

Structure of Hunt's Algorithm

cct | College Dublin
Computing • IT • Business

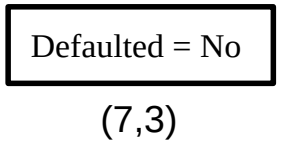
- We can develop decision trees in a recursive fashion using a Hunt's algorithm.
- The training dataset is successively partitioned until form the purer subsets. Let $\mathbf{D_t}$ be the set of training records that reach a node **t**.
- The general recursive procedure is defined as below
- **Methodology:**
  - If $\mathbf{D_t}$ contains records that belong to the same class $\mathbf{y_t}$, then **t** is a leaf node labeled as $\mathbf{y_t}$
  - If $\mathbf{D_t}$ is an empty set, then **t** is a leaf node labeled by the default class as $\mathbf{y_d}$
  - If $\mathbf{D_t}$ contains records that belong to more than one class, use an attribute test to split the data into smaller subsets. Recursively apply the procedure to each subset.

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

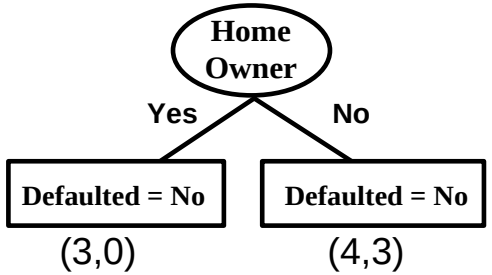


Hunt's Algorithm

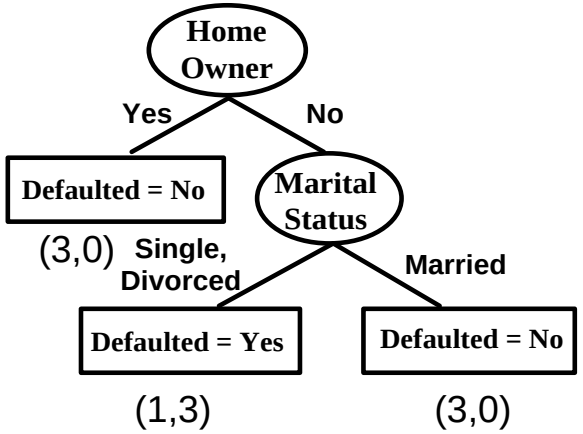
ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



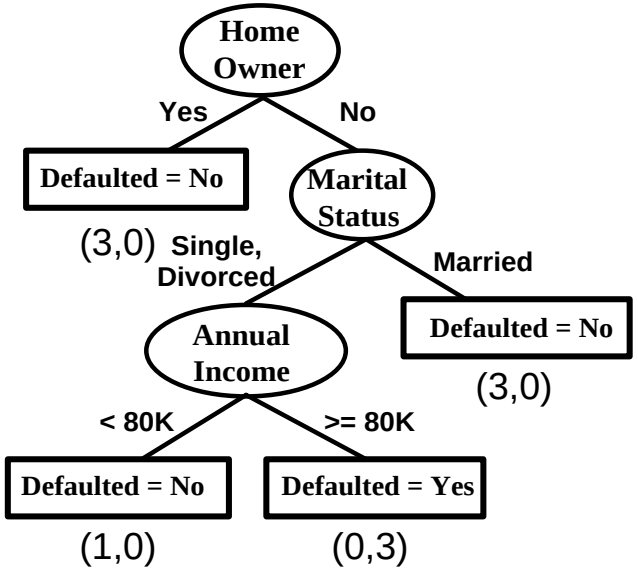
(a)



(b)



(c)



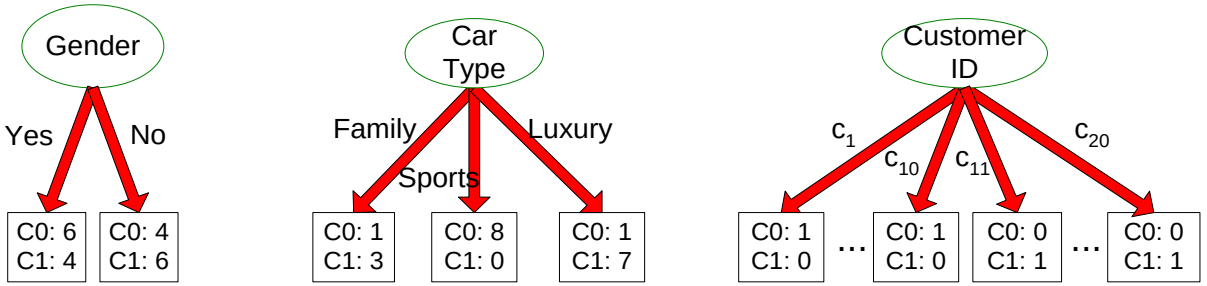
(d)

- **Stop the Split Procedure**
- A stopping condition is needed to terminate the tree-growing process.
- One approach is to keep growing a node until all of the records are either members of the same class or have the same attribute values.
- Despite the fact that there are enough circumstances for the decision tree induction technique to stop, some algorithms also use additional criteria to end the tree-growing process earlier.

Best Split in Decision Trees

College Dublin
Computing • IT • Business

Before Splitting: 10 records of class 0,
10 records of class 1



Which test condition is the best?

Customer Id	Gender	Car Type	Shirt Size	Class
1	M	Family	Small	C0
2	M	Sports	Medium	C0
3	M	Sports	Medium	C0
4	M	Sports	Large	C0
5	M	Sports	Extra Large	C0
6	M	Sports	Extra Large	C0
7	F	Sports	Small	C0
8	F	Sports	Small	C0
9	F	Sports	Medium	C0
10	F	Luxury	Large	C0
11	M	Family	Large	C1
12	M	Family	Extra Large	C1
13	M	Family	Medium	C1
14	M	Luxury	Extra Large	C1
15	F	Luxury	Small	C1
16	F	Luxury	Small	C1
17	F	Luxury	Medium	C1
18	F	Luxury	Medium	C1
19	F	Luxury	Medium	C1
20	F	Luxury	Large	C1

- **Node splitting, or simply splitting, is the process of dividing a node into multiple sub-nodes to create relatively pure nodes.**
- There are multiple ways of doing this, which can be broadly divided into two categories based on the type of target variable.
- **Continuous Target Variable**
  - Reduction in Variance
- **Categorical Target Variable**
  - Gini Impurity
  - Information Gain
  - Chi-Square

Measures of Node Impurity

cct | College Dublin
Computing • IT • Business

- The node impurity is a measure of the homogeneity of the labels at the node. **Gini impurity** and **Entropy** are used for classification.

- **Gini Index**

$$GINI\ (t) = 1 - \sum_j [\, p(\, j \mid t)]^2$$

(NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).

- **Entropy**

$$Entropy\ (t) = -\sum_j p(\, j \mid t) \log\ p(\, j \mid t)$$

- **Misclassification error**

$$Error\ (t) = 1 - \max_i\ P(i \mid t)$$

Finding the Best Split

1. Compute impurity measure (**P**) before splitting
2. Compute impurity measure (**M**) after splitting
   - Compute impurity measure of each child node
   - **M** is the weighted impurity of children
3. Choose the attribute test condition that produces the highest gain

Gain = P – M

or equivalently, lowest impurity measure after splitting **(M)**

Measure of Impurity

GINI

- **Gini Index** for a given node **t**

$$GINI\ (t) = 1 - \sum_j [\, p(\, j \mid t)]^2$$

(NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).

- Maximum $(1 - 1/n_c)$ when records are equally distributed among all classes
- Minimum (0.0) when all records belong to one class
- **For 2-class or binary problem (p, 1 – p)**
  - GINI = $1 - p^2 - (1 - p)^2 = 2p\,(1 - p)$

C1	0
C2	6
Gini=0.000	

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Gini = $1 - P(C1)^2 - P(C2)^2 = 1 - 0 - 1 = 0$

C1	1
C2	5
Gini=0.278	

P(C1) = 1/6 P(C2) = 5/6

Gini = $1 - (1/6)^2 - (5/6)^2 = 0.278$

C1	2
C2	4
Gini=0.444	

P(C1) = 2/6 P(C2) = 4/6

Gini = $1 - (2/6)^2 - (4/6)^2 = 0.444$

C1	3
C2	3
Gini=0.500	

P(C1) = 3/6 P(C2) = 3/6

Gini = $1 - (3/6)^2 - (3/6)^2 = 0.5$

Calculation of Gini Index

cct | College Dublin
Computing • IT • Business

- When a node **p** is split into **k** partitions (children)

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

where, $\mathbf{n_i}$ = number of records at child **i**

n = number of records at parent node **p**

- Choose the attribute that minimizes weighted average **Gini index** of the children
- The above-mentioned generalized formula can be used for two or more splits.
- Gini index is used in decision tree algorithms, such as **CART, SLIQ and SPRINT.**

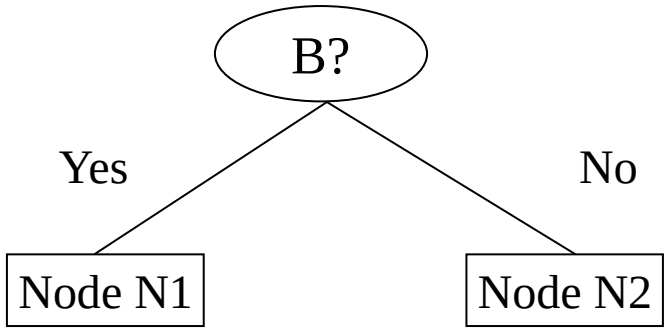
Calculation of Gini Index

Binary Attributes

- Splits into two partitions
- Effect of Weighing partitions:
  - Larger and Purer Partitions are sought for

P(C1) = 7/12 P(C2) = 5/12

Gini = $1 - (7/12)^2 - (5/12)^2 = 0.486$



	Parent
C1	7
C2	5
Gini = 0.486	

Gini(N1)
= $1 - (5/6)^2 - (1/6)^2$
= 0.278

Gini(N2)
= $1 - (2/6)^2 - (4/6)^2$
= 0.444

	N1	N2
C1	5	2
C2	1	4
Gini=0.361		

Weighted Gini of N1 N2
= 6/12 * 0.278 +
  6/12 * 0.444
= 0.361

Gain = 0.486 – 0.361 = 0.125

Calculation of Error

Single Node

cct College Dublin
Computing • IT • Business

Classification error at a node **t**

$$Error\ (t) = 1 - \max_i P(i \mid t)$$

- Maximum (**1 - 1/n_c**) when records are equally distributed among all classes, implying least interesting information
- Minimum (**0**) when all records belong to one class, implying most interesting information

Node N_1	Count
Class=0	0
Class=1	6

$\text{Gini} = 1 - (0/6)^2 - (6/6)^2 = 0$
$\text{Entropy} = -(0/6)\log_2(0/6) - (6/6)\log_2(6/6) = 0$
$\text{Error} = 1 - \max[0/6, 6/6] = 0$

Node N_2	Count
Class=0	1
Class=1	5

$\text{Gini} = 1 - (1/6)^2 - (5/6)^2 = 0.278$
$\text{Entropy} = -(1/6)\log_2(1/6) - (5/6)\log_2(5/6) = 0.650$
$\text{Error} = 1 - \max[1/6, 5/6] = 0.167$

Node N_3	Count
Class=0	3
Class=1	3

$\text{Gini} = 1 - (3/6)^2 - (3/6)^2 = 0.5$
$\text{Entropy} = -(3/6)\log_2(3/6) - (3/6)\log_2(3/6) = 1$
$\text{Error} = 1 - \max[3/6, 3/6] = 0.5$

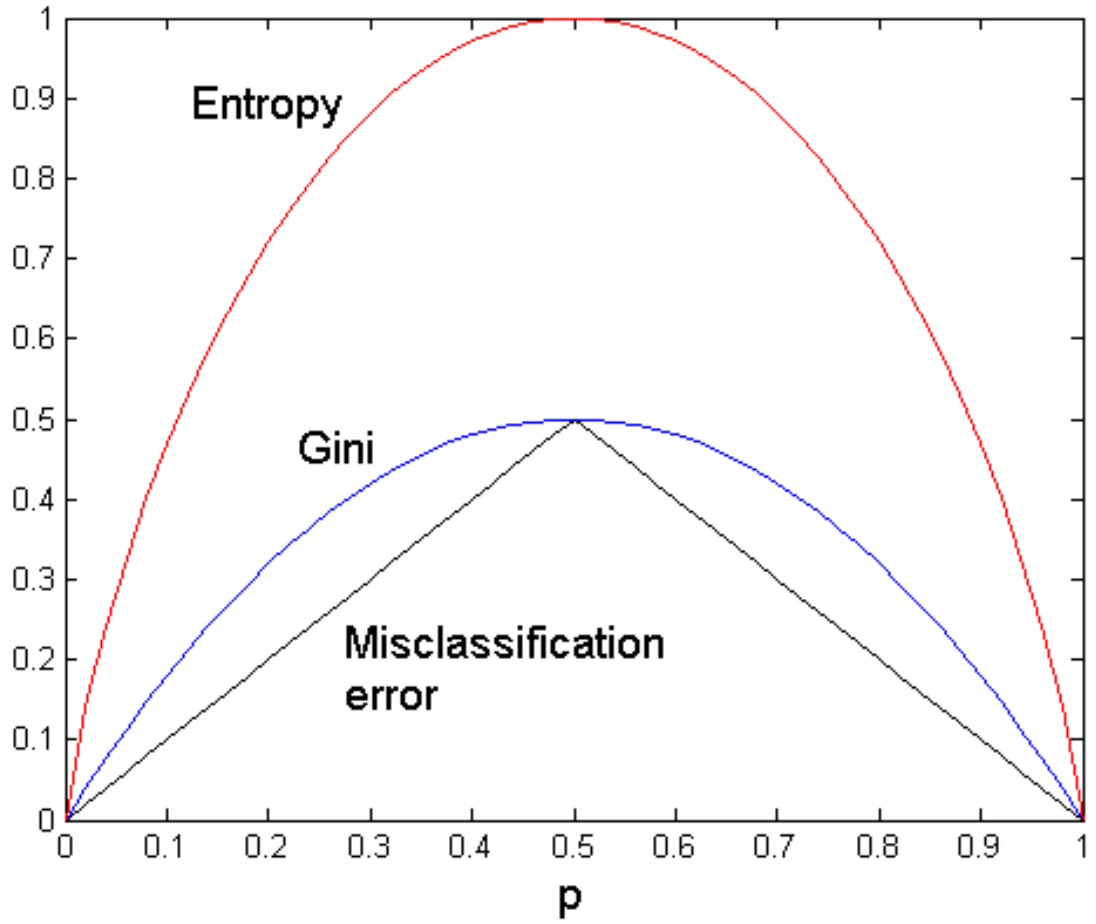
Comparison

Impurity Measures

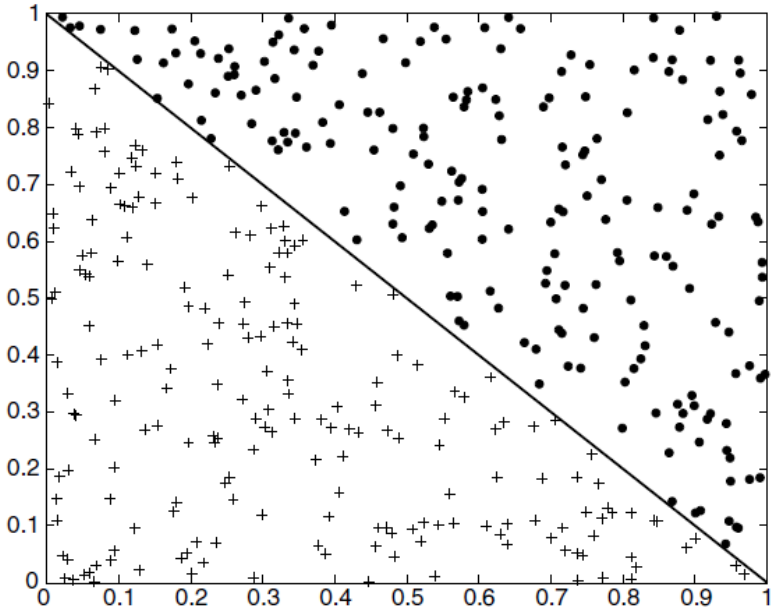
cct College Dublin
Computing • IT • Business

For a 2-class problem:

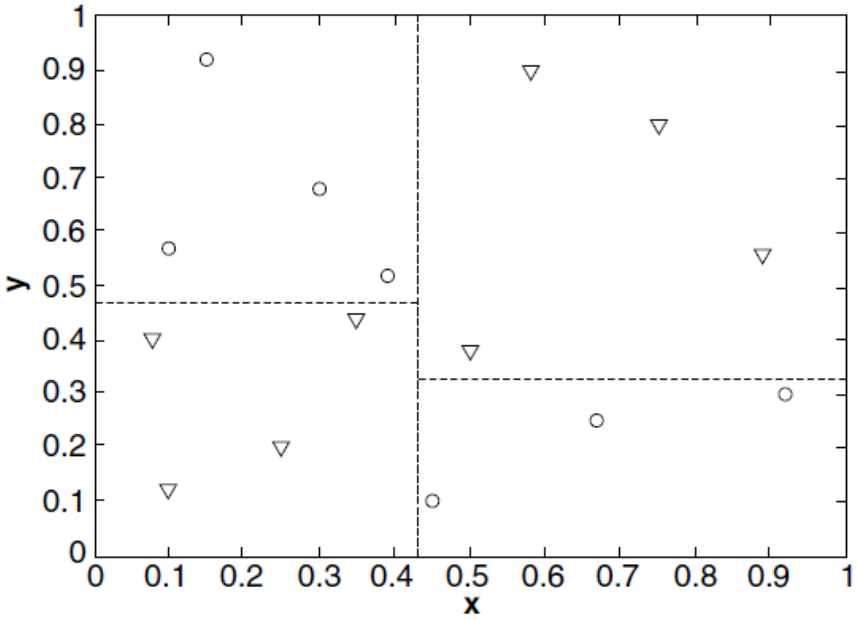
- The **Gini Index** and the **Entropy** have two main differences
- Gini Index has values inside the interval [0, 0.5] whereas the interval of the Entropy is [0, 1].
- Misclassification error has also values in the interval [0, 0.5].

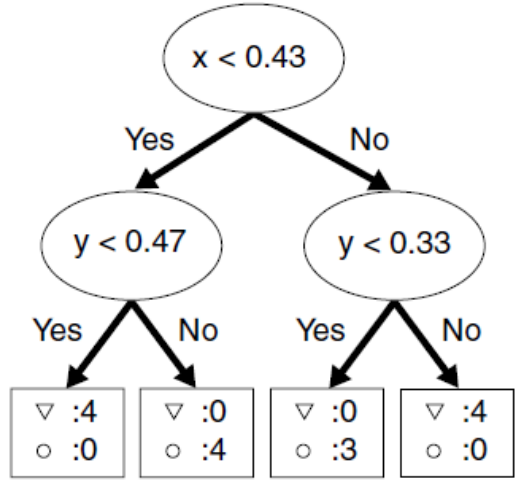


Limitations of
Single Attribute-based Decision Boundaries



- Example of data set that cannot be partitioned optimally using test conditions involving single attributes.





- Example of a decision tree and its decision boundaries for a two-dimensional data set.

Requirements for using
Decision Trees

cct College Dublin
Computing • IT • Business

- **Decision Tree is supervised classification method**
  - The target variable must be categorical
  - Pre-classified target variable must be included in the training set
  - Decision trees learn by example, so training set should contain records with varied attribute values
  - If the training set systematically lacks definable subsets, classification becomes problematic
  - There are different measures for leaf node purity
  - **Classification and Regression Trees (CART)** and **C4.5** are two leading algorithms used in the machine learning

Classification and Regression Trees

Pruning

cct College Dublin
Computing • IT • Business

- In machine learning, **pruning** is a data compression technique that shrinks the size of decision trees by eliminating parts of the tree that are unnecessary and redundant for classifying instances.

- **Pruning** is a method for getting rid of the Decision Tree's components that keep it from developing to its maximum depth. The portions of the tree that lack the ability to classify instances are the portions that are removed. Pruning is crucial because training a decision tree to its maximum depth will certainly result in overfitting the training set.

- To put it simply, **Decision Tree Pruning** aims to build an algorithm that performs worse on training data but performs better on test data in terms of generalization. There are two common strategies to prevent overfitting

    1. Stopping the creation of the tree early (also called pre-pruning)

    2. Building the tree but then removing or collapsing nodes that contain little information (also called post-pruning or just pruning).

- Possible criteria for pre-pruning include limiting the maximum depth of the tree, limiting the maximum number of leaves, or requiring a minimum number of points in a node to keep splitting it.

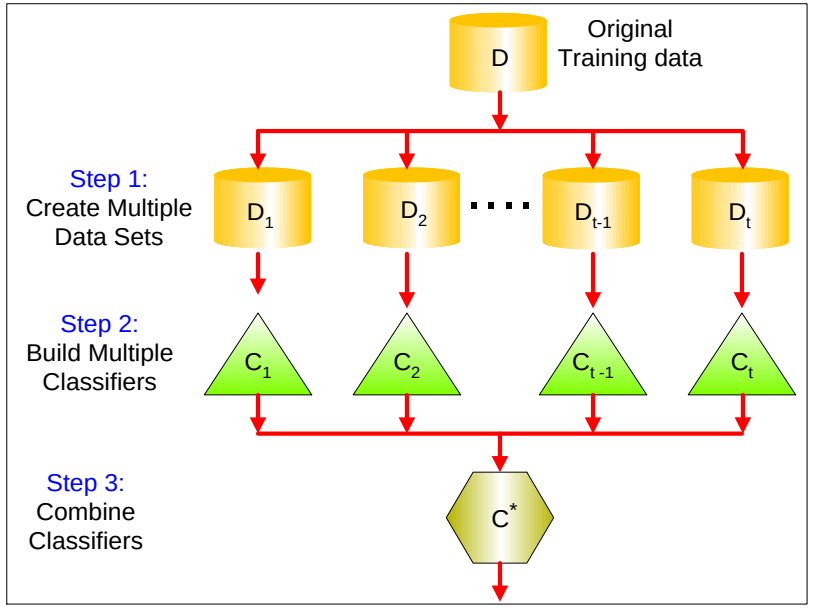
Decision Tree Based Classification

cct | College Dublin
Computing • IT • Business

- **Advantages:**
  - Inexpensive to construct
  - Extremely fast at classifying unknown records
  - Easy to interpret for small-sized trees
  - Robust to noise (especially when methods to avoid overfitting are employed)
  - Can easily handle redundant or irrelevant attributes (unless the attributes are interacting)
- **Disadvantages:**
  - Space of possible decision trees is exponentially large. Greedy approaches are unable to find the best tree
  - Does not take into account interactions between attributes
  - Each decision boundary involves only a single attribute

Ensemble Classifiers

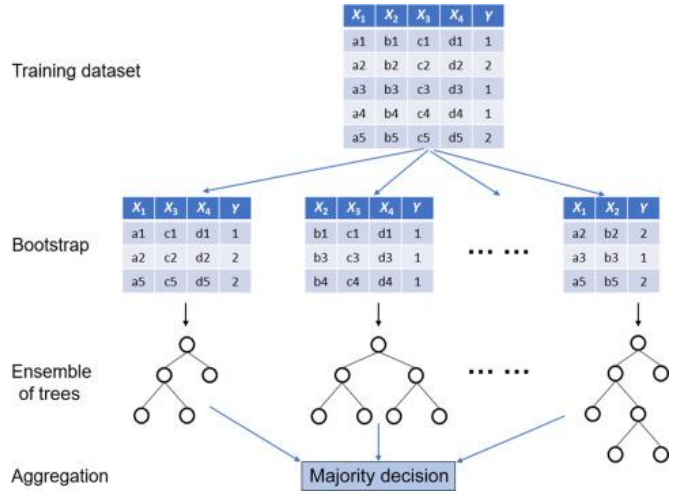
- **Types of Ensemble Methods**
- Manipulate data distribution
  - Example: bagging, boosting
- Manipulate input features
  - Example: **random forests**
- Manipulate class labels
  - Example: error-correcting output coding

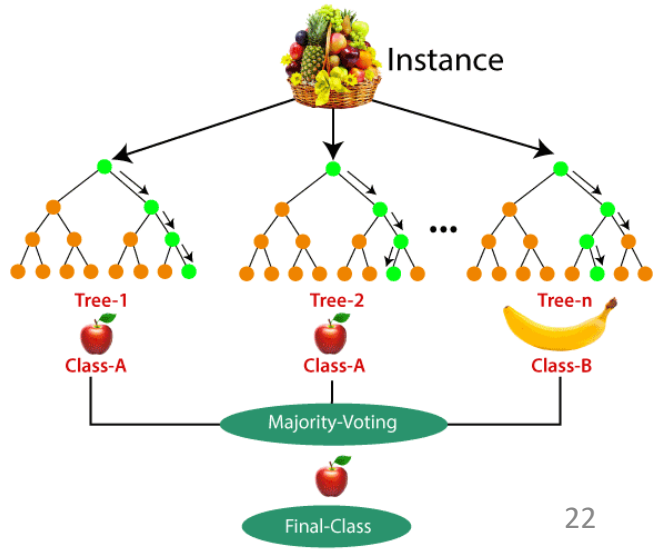


- In ML, multiple instances of the same model are trained on various subsets of the training data using the ensemble technique known as **bagging (Bootstrap Aggregating)**, and then the predictions from all of the instances are combined to get the final prediction. The procedure of **bootstrapping** is used to produce the data subsets from random sampling with replacement.
- In **boosting**, the weak learners are instructed one after the other while concentrating on the errors of the prior learners. In boosting, each repetition gives the misclassified cases more weight.

Random Forest

- **Random Forest** is a supervised learning algorithm. It can be used both for **Classification** and **Regression**.
- A forest is comprised of trees. It has been observed that more trees can form a forest rapidly.
- Random forest creates decision trees on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting.
- It also provides a pretty good indicator of the feature importance.
- Random forests have a variety of applications, such as recommendation engines, image classification and feature selection.
- It can be used to classify loyal loan applicants, identify fraudulent activity and predict diseases.





Random Forest

Example

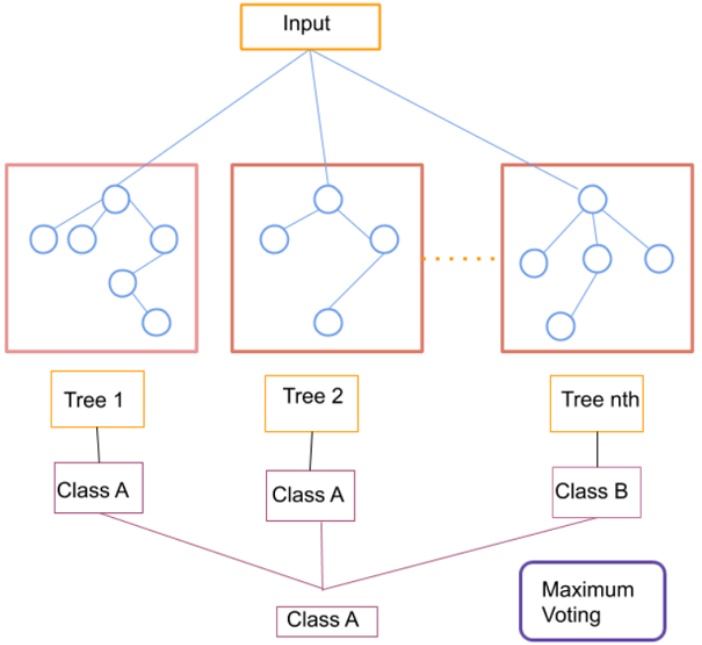
cct College Dublin
Computing • IT • Business

- Suppose you want plana trip and you would like to travel to a specific destination but not sure.
- How to find a place that you will like? You can search online, read reviews on travel blogs and portals, or you can also ask to your friends.
- Let's suppose you have decided to ask your friends and talked with them about their past travel experience to various places. You will get some recommendations from every friend. Now you have to make a list of those recommended places. Then, you ask them to vote (or select one best place for the trip) from the list of recommended places you made. The place with the highest number of votes will be your final choice for the trip.
- In the above decision process, there are two parts. **First**, asking your friends about their individual travel experience and getting one recommendation out of multiple places they have visited.
- This part is like using the decision tree algorithm. In this scenario, each friend makes a selection of the places he or she has visited so far.
- **The second part**, after collecting all the recommendations, is the voting procedure for selecting the best place in the list of recommendations. This whole process of getting recommendations from friends and voting on them to find the best place is known as the random forest algorithm.

Random Forest Algorithm

- **It works in four steps**

1. Select random samples from a given dataset.
2. Construct a decision tree for each sample and get a prediction result from each decision tree.
3. Perform a vote for each predicted result.
4. Select the prediction result with the most votes as the final prediction.



Random Forests vs Decision Trees

- Random forests is a set of multiple decision trees.
- Deep decision trees may suffer from overfitting, but the random forests prevent overfitting by creating trees on random subsets.
- Decision trees are computationally faster.
- Random forests is difficult to interpret, while a decision tree is easily interpretable and can be converted to rules.

Resources/ References

- Introduction to Data Mining, 2nd Edition, Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar, 2019, Pearson.
- Introduction to Machine Learning with Python, Andreas C. Müller and Sarah Guido, O'Reilly Media, Inc. October 2016.
- Data Mining And Machine Learning, Fundamental Concepts And Algorithms, MOHAMMED J. Zaki, Wagner Meira, Jr., Cambridge CB2 8BS, United Kingdom, 2020.
- Discovering Knowledge In Data: An Introduction To Data Exploration, Second Edition, By Daniel Larose And Chantal Larose, John Wiley And Sons, Inc., 2014.
- UCI Repository: http://www.ics.uci.edu/~mlearn/MLRepository.html
- Statlib: http://lib.stat.cmu.edu/
- Some images are used from Google search repository (https://www.google.ie/search) to enhance the level of learning.