Machine Learning

Decision Tree

Random Forest

Ensemble Methods

1. Decision Trees

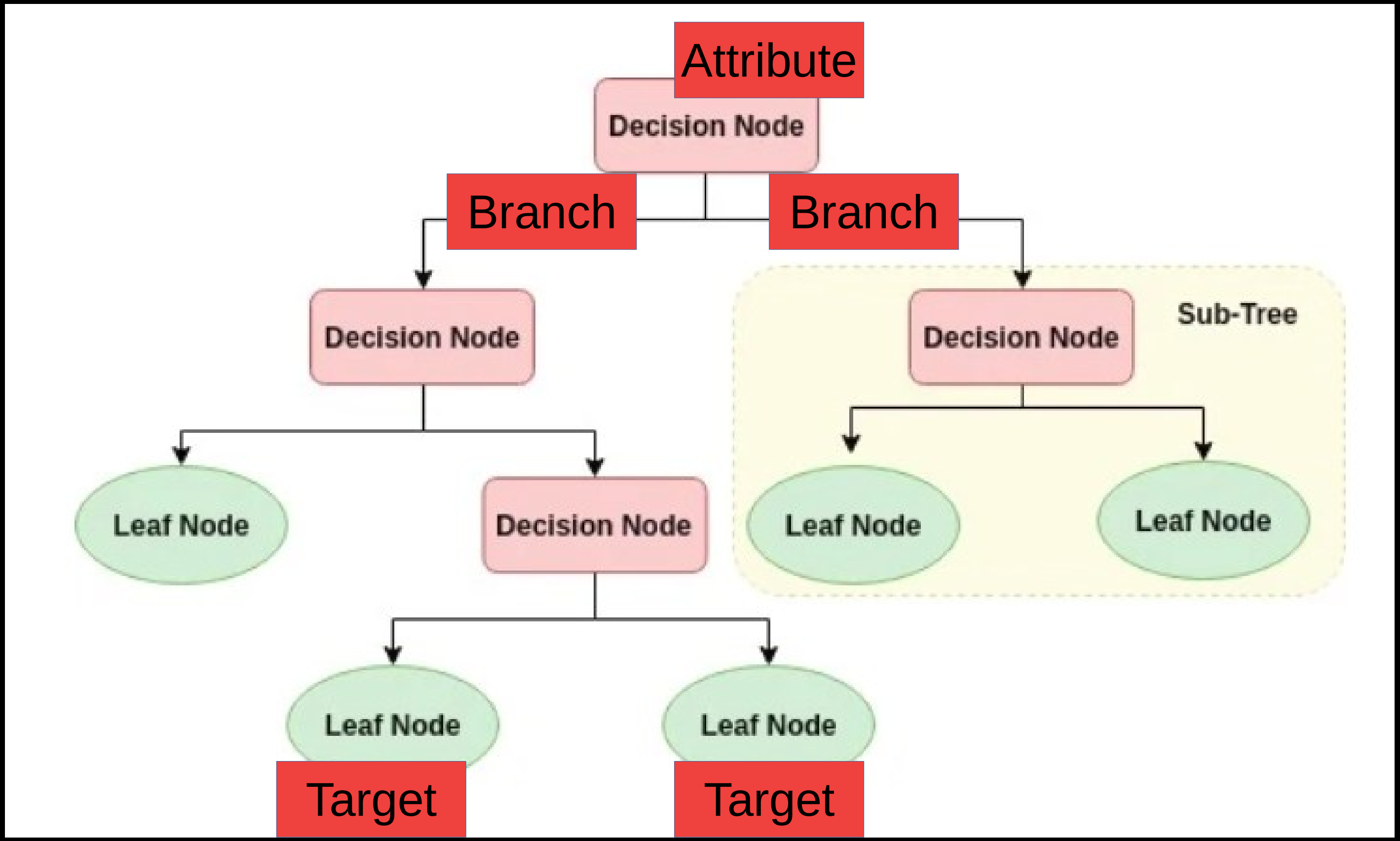
Learning models: (algorithms)

- C5.0
- CART
- C4.5
- ID3

...

Regression

Classification



2. Gain of Information, Entropy, Gini Index & Pruning

Considerations in the construction of decision trees:

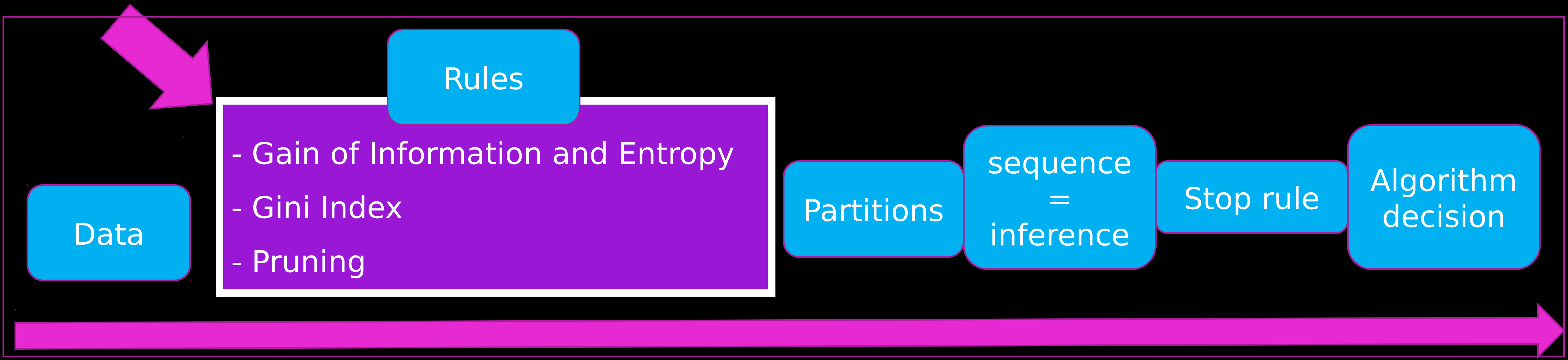
- Which attribute should be used to start the tree?

- What should be the next attribute?

- When to stop building branches on the tree (to avoid overfitting)?

"Making a decision" from ML's point of view is to use the term "making an inference", which is one of the elements supporting decision making.

Decision making is the last stage of a complex process, which involves:
- Diagnosis
- Pattern Recognition
- Causal analysis, etc.



Data

Rules

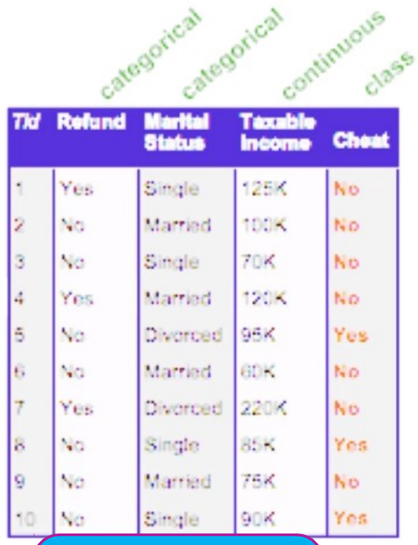
- Gain of Information and Entropy
- Gini Index
- Pruning

Partitions

sequence = inference

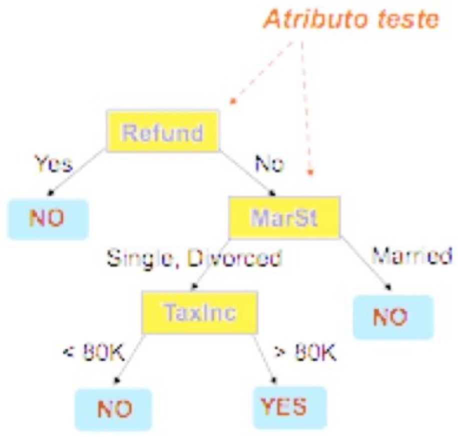
Stop rule

Algorithm decision

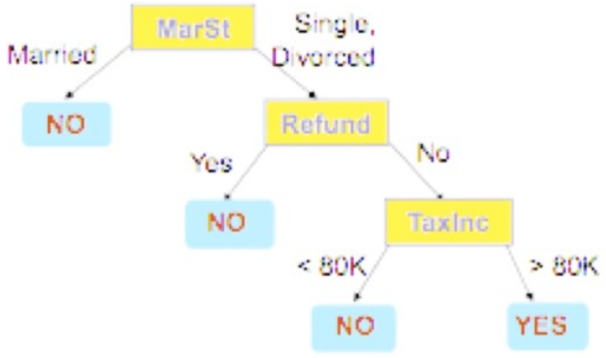

categorical
categorical
continuous
class

Tid	Refund	Marital Status	Taxable Income	Cheat
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



Model: Decision Tree



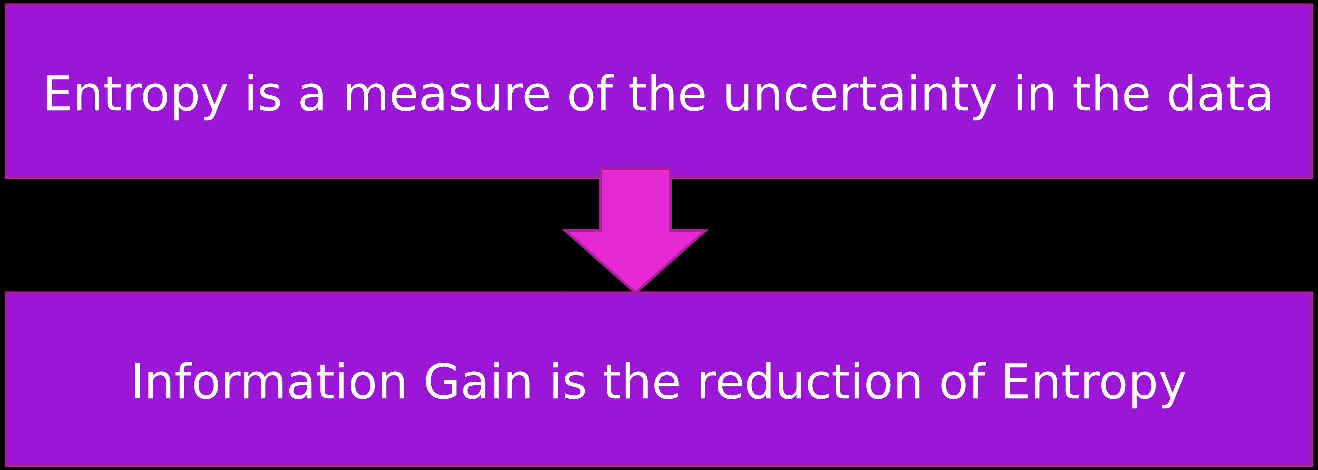
There can be more than one decision tree for the same data set

How to define the **Root Node** and how to D**ivide the Dataset?**

- Greedy Selection
- Division based on nominal attributes
  - Binary Bivision
  - Multiple Division
- Division based on continuous attributes
  - Binary Decision
  - Discretization (Static & Dynamic)

Stop rule

Gain of Information, Entropy



Entropy is a measure of the uncertainty in the data

Information Gain is the reduction of Entropy

Stop rule

Gini Index

The Gini index is used to measure the probability that two random items belong to the same class.

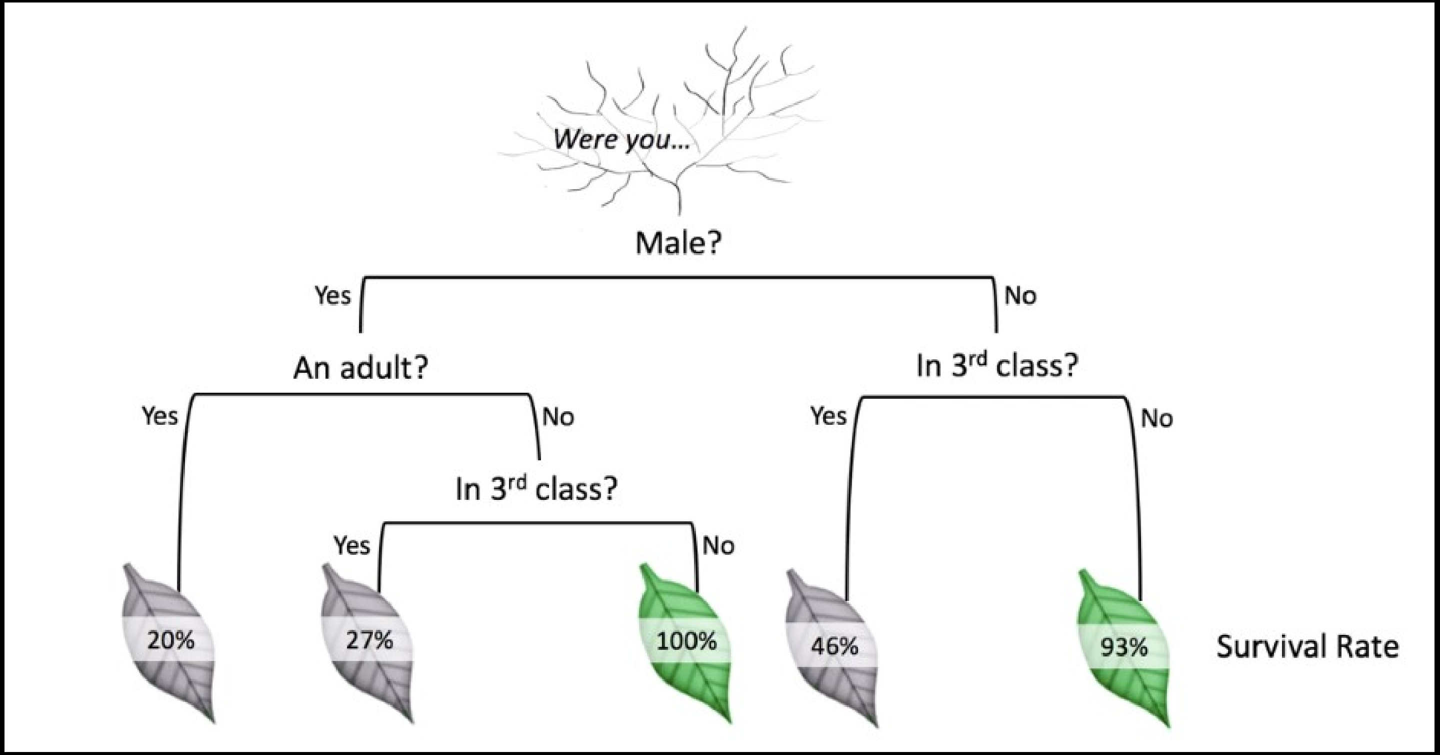
The Gini measure of a node is the sum of the squares of the proportions of the classes.

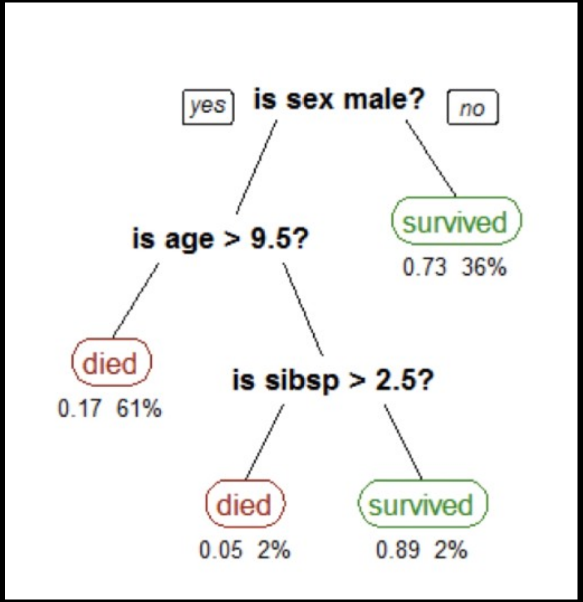
In the ID3, C4.5 and C5.0 algorithms, the root node is chosen based on how much of the total Entropy is reduced, if that node is chosen.

This is called Information Gain!

Information Gain = System Entropy before division - System Entropy after division

3. Stopping Rules

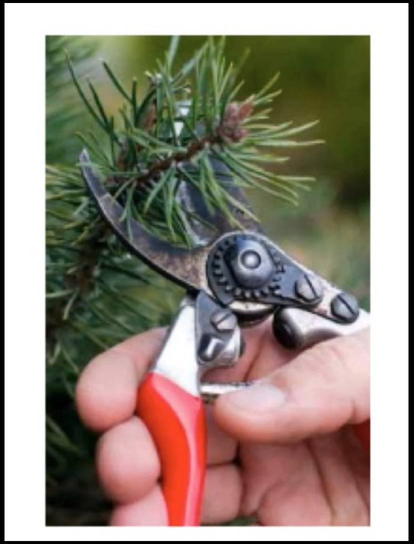




Stop Rules:

- Gini index
- Chi-square
- Gain information
- Variance Reduction

4. Pruning



Pruning

- The decision tree is completed before a perfect classification of the training data is achieved.

- Over-adjustment occurs in the data, generating a model, and then the tree is pruned to become **generalizable**.

And how to define the correct size of the tree?

- Use a validation set.
- Use probabilistic methods.

5. Feature Selection

Decision Tree, Random Forest and Ensemble Methods

- Using *several trees* simultaneously, some Ensemble Method algorithms surpass the predictive ability of Deep Learning models.

- Decision Tree groups or *Ensemble Methods* offer an excellent level of accuracy, with relatively low complexity.

In addition to being a Machine Learning model, **Random Forest** is also widely used to carry out **selection of variables**, which are the best candidates for variable predictors in the Machine Learning model.

All ensemble methods can be used for the purpose of **selecting variables,** using a machine learning model to find the best variables.
You select these variables and use them in the final version of the model.

We can use xgboost to find the main variables and thus apply attribute selection, which is a technique used before building the machine learning model.

6. Ensemble Methods

The challenge is to find the best possible combination of parameters for different models!

Bagging

Bagging is used to build multiple models (usually of the same type) from different subsets in the training data set.

A Bagging classifier is an ensemble meta-estimator that fits base classifiers, each into random subsets of the original data set, and then aggregates its individual predictions (by vote or average) to form a final forecast.

Such a meta-estimator can typically be used as a way to reduce the variance of an estimator (for example, a decision tree) by introducing randomization into your construction procedure and making an ensemble from it.

Building models with **Ensemble** method is not very different from an individual model, the difference is that we have **more parameters** that we have to adjust, to create the best possible version of the model!

Adaboost

An AdaBoost classifier is a meta-estimator that starts by adjusting a classifier in the original data set and then adjusts additional copies of the classifier in the same data set, but where the weights of the incorrectly classified instances are adjusted so that subsequent classifiers focus more on difficult cases.

Important!

First create a **grouping** of Decision Trees (Machine Learning Models), then use a strategy:

Bagging Approach: averaging forecasts (base estimators)

Boosting Approach: Assign weights to the outputs of each estimator (learn from the past errors)

Voting Approach: make a vote among the models and choose the one that has the best performance.

Tip:

Most used metric for Classification: accuracy (bigger, better)

Most used metric for Regression: mean squared error (smaller, better)

7. Others Ensemble Methods

Gradient Boosting: Gradient Boosting or Gradient Boostted Regression Trees (GBRT) is a **non-parametric statistical** learning technique used for Classification and Regression problems.

Stochastic Gradient Boosting: Create subsamples of the training dataset (when it is very large) before growing each tree.

eXtreme Gradient Boosting (XGBoost): GRADIENT BOOSTING evolution! (Kaggle competitions!)

Gradient Boosting = **Gradient Descent** + **Boosting**.

Basically 3 steps are performed in the construction of the model:

1- Generates a Regressor* (even if the ultimate goal is a classifier)
2- Computes the residual error
3- Learn to predict the residue.

Note that: through the loss function, it learns to predict the residue! and as it learns to predict, it reduces the residue

By reducing the residue, we are reducing the difference between the observed value and the predicted value.

Consequently we are finding the best TARGET FUNCTION which is what we want to predict in machine learning!

XGBoost is an optimized distributed gradient augmentation library, designed to be highly efficient, flexible and portable.

It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree augmentation (also known as GBDT, GBM) that solves many Data Science problems quickly and accurately.

The same code is executed in the distributed environment (Hadoop, SGE, MPI) and can solve problems with data from billions of records.

Widely used in Kaggle competitions.

https://xgboost.readthedocs.io/en/latest/

Important!

- Realize that there is no single right answer for building a model in Machine Learning, what will define the success of your model is if it is in the end a GENERALIZABLE model with the LOWEST POSSIBLE ERROR RATE and one with a HIGH LEVEL OF PRECISION!
- Remember, accuracy cannot only be assessed in training, but must also be assessed in testing!
- With a model ready, we can present new data for him to make predictions.

End!