* Decision Tree (DT)

- Supervised ML algorithm used for classification and Regression.

  DTC: Decision Tree Classifier

  DTR: Decision Tree Regressor
- The goal is to create a model that predicts the value of target variable by learning simple decision rules infered from data features.
- More complex the rules more fitter the model.
- Tree can be explained by two terms:

  @ Leaf node: nodes with decisions or final outcomes

  ⓑ Decision node: nodes where the data split.
- **Advantages:**
  1. Easy to understand and interpret
  2. Easily visualized
  3. Cost of using tree is $\log_2$(no. of data points)
  4. Can handle multi output problems
  5. Performs well even if assumptions are somewhat violated by true model from which data was generated.
  6. Use white box model, contrast to Neural network
- **Disadvantages:**
  1. Unstable as small variation in data can result in completely new tree.
  2. Predictions of DT are neither continous nor smooth instead its piecewise constant approximation.
  3. Overfitting can occur due to over complex DT, which is treated using pruning methods.
  4. DT solves NP problems which uses heuristic approach and unlike greedy it is only localy optimal and doesn't guarantee global optimal solution.
  5. DT learners create biased solution trees if any class dominates.

A. Terminologies

- **Instances:** refers to vector of features or attribute that define space.
- **Attribute:** quantity describing an instance.
- **Concept:** function that maps inputs to outputs
- **Target Concept:** function we are trying to find (basically output feature)
- **Hypothesis Class:** set of all possible functions
- **Sample:** set of inputs paired with label
- **Candidate:** Concept which we think is Target concept
- **Testing Set:** use to test candidate concept and determine performance.

- **Entropy:** Shanon Entropy defined for finite set(S), is measure of amount of uncertanity or randomness in data.
- Intutivelly it tells about predictibility of event.

$$\boxed{H(S) = \sum_{i=1}^{n} -p_i \log_2(p_i)}$$

where p_i is probability of class i.

- It is measure of impurity in given sample training data.

- **Information Gain:** Kullback Leibler Divergence is the effective change in entropy after deciding on particular attribute(A) i.e for a Node.
- It measures relative change in entropy with respect to independent variables.
- It measures the effectivnes of an attribute in classifying data which is finded by calculating the change in entropy before and after the classificatior or split.

$$\boxed{IG(S,A) = H(S) - H(S,A)}$$

or

$$\boxed{IG(S,A) = \underbrace{H(S)}_{\text{Before split}} - \underbrace{H(S,A)}_{\text{after split}}}$$

or

$$\boxed{IG(S,A) = H(S) - \sum_{i=1}^{n} p_i \cdot H(S)}$$

In short,

$$\boxed{IG = \text{Entropy(parent node)} - [\text{Average Entropy(child node)}]}$$

- **Gini Index:** Cost function used to evaluate splits in dataset.

$$\boxed{GI = 1 - \sum_{i=1}^{n} (p_i)^2}$$

- where p_i is probability of class.
- it only works with binary split.