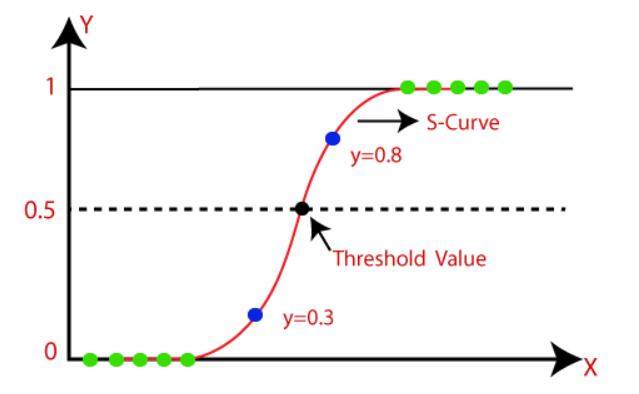
**1] Logistic Regression:**

* Logistic regression is a supervised ML technique to solve classification problems.
* It is used to predict categorical dependent variables using a given set of independent variables.
* For example, 0 or 1, yes or no, etc.
* It is similar to linear regression except that it is used for classification tasks.
* Instead of fitting a regression line, we fit an ‘S’ shaped curve logistic function.



**Sigmoid Function:**

* Logistic regression uses a sigmoid function that maps the predicted values to the probabilities.
* It maps any real value to another value ranging between 0 and 1.
* As the values are between 0 and 1, it forms the ‘S’ shaped curve.
* It uses the concept of threshold value such that the values above the threshold value tends to be 1 and values below the threshold value tends to be 0.

**Assumptions for logistic regression:**

* The dependent variable must be categorical.
* There should not be multi-collinearity.

**Logistic regression equation:**

Sigmoid(z) = 1/(1-ez)

**Types of logistic regression:**

1. **Binomial:** in binomial regression, there can be only two possible dependent variables.
2. **Multinomial:** in this, there can be 3 or more unordered dependent variables. For example- ‘cats’, ’dogs’, or ’cow’, etc
3. **Ordinal:** In this, there can be 3 or more possible ordered dependent variables. For example- ‘high’, ‘medium’, or ‘low.

**2] K Nearest Neighbors (K-NN):**

* K-NN is the supervised machine learning algorithm used for both classification and regression task, but it is used mostly for classification task.
* K-NN assumes the similarity between the new case/ data and available cases and then classifies it into the suitable category.
* It is also called a lazy learner as it does not learn immediately from the training dataset, instead it stores all the available dataset and at the time of classification, it performs action on the training dataset.

**Why do we need K-NN:**

Suppose there are three categories, A, B, and C and there is a new datapoint X. So, K-NN can help in identifying the category of new datapoint.

**Algorithm for K-NN:**

1. Decide the number of K nearest neighbors.
2. Calculate Euclidean distance of new datapoint with all the datapoints.
3. Take the K nearest neighbors as per the calculated Euclidean distance.
4. Among these K neighbors, count the number of the datapoints in each category.
5. Assign the new datapoints to the category for which the number of neighbor is maximum.
6. Model is ready.

**Advantages of k-NN:**

* It is simple to implement.
* It is robust to noisy data.
* It can be more effective if the training data is large.

**Disadvantages of K-NN:**

* It can be sometime complex to determine the value of K.
* The computation cost is high as it calculates Euclidean distance between all the data points.

**KNN model and its parameters:**

* Model = KNearestNeighbors(n\_neighbors = 5, metric = ‘minkowski’ , p = 2)  
  minkowski is distance metric and generalization of other distance metric such as Euclidean distance and Manhattan distance.
* P=2 indicates the Euclidean distance and p=1 indicates the Manhattan distance.

**3] Decision Tree**

* Decision tree is a supervised machine learning algorithm used for classification and regression, but it is best suited for classification tasks.
* It is a tree-like structure where internal nodes represent the features of the dataset and the branches represent decision rules and each leaf node represents the outcome.
* Decision nodes have multiple branches and leaf nodes are the output of those decision nodes and do not have further branches.
* It is called a decision tree because it has a tree-like structure, which starts with root node and expands on further branches.
* In order to build a decision tree, it uses CART (Classification and Regression Tree) algorithm.

A diagram of decision tree

Description automatically generated

**Why use a decision tree:**

* It mimics human thinking ability while making a decision, so it is easy to understand.
* The logic behind the tree can be easily understood as it shows a tree-like structure.

**Decision tree terminologies:**

* **Root node**: The decision tree starts with the root node which represents the complete dataset.
* **Leaf node:** It represents the outcome of the decision.
* **Splitting:** It is the process of dividing the decision node into sub-nodes according to the given conditions.
* **Pruning:** It is the process of removal of unwanted nodes.
* **Parent/Child node**

**Decision tree algorithm:**

**Step 1**: Begin the tree with the root node, which represents the entire dataset.

**Step 2:** Find the best attribute in the dataset using Attribute Selection Measure (ASM).

**Step 3**: Divide the root node into subsets that contain all possible values of the best attribute.

**Step 4:** Generate the decision tree node that contains the best attribute.

**Step 5:** Recursively generate the decision trees using the subsets created in step 3, until the leaf node encounter.

**Attribute Selection Measure:**

**Two methods:**

1. **Information gain**
2. **Gini Index**

**1] Information Gain:**

**Entropy:** It is the measure of impurity or uncertainty in a set of data.

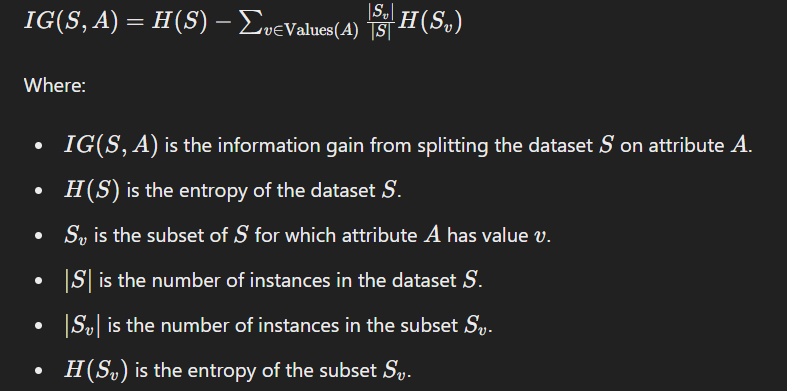
Entropy(s) = -p(yes)log2p(yes) -p(no)log2p(no)

Entropy = 0 indicates no uncertainty(all data points belong to single class)

Entropy = 1 indicates too much uncertainty (equal distribution of values)

**Information gain:** It measures the reduction in entropy after splitting the dataset on an attribute.

The attribute whose information gain is higher, is the best attribute.





A notebook with writing on it

Description automatically generated

A hand holding a notebook with writing

Description automatically generated

A notebook with writing on it

Description automatically generated

A hand holding a notebook with writing on it

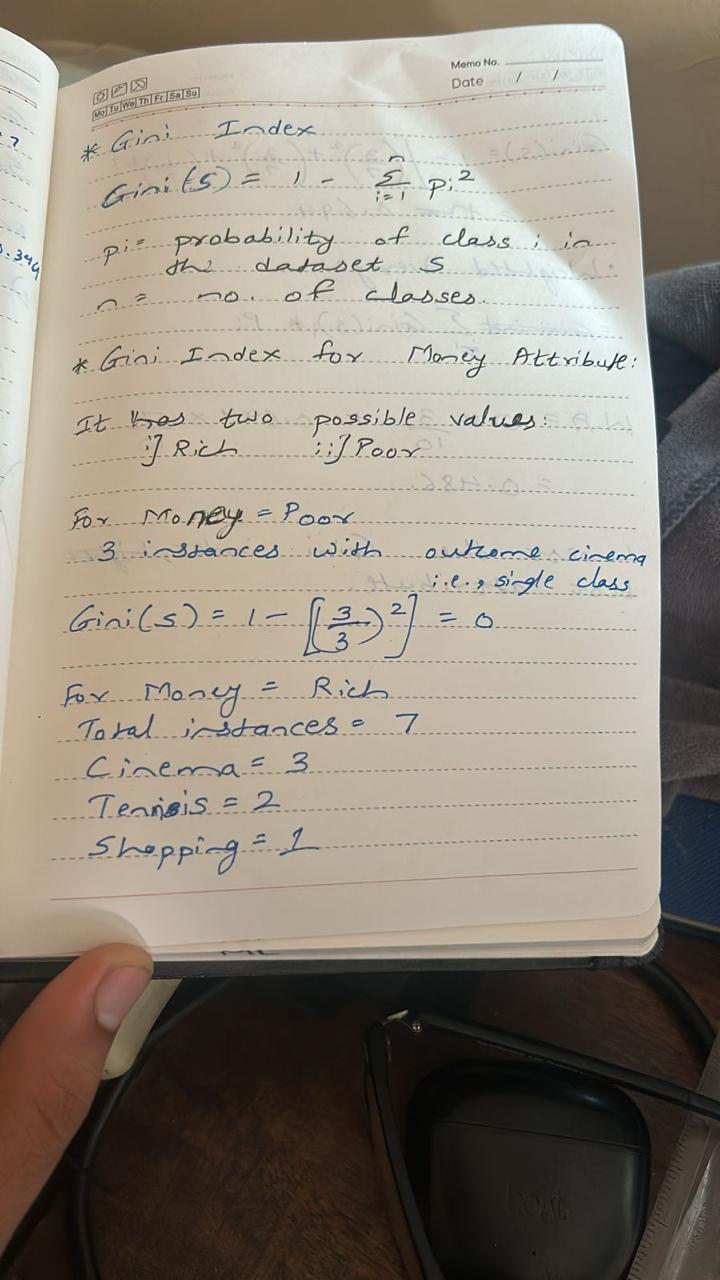
Description automatically generated

**Gini index:** It is the measure of impurity.

Lower Gini index indicates better attribute.

A table with text on it

Description automatically generated



A hand holding a notebook with writing on it

Description automatically generated

**Information gain vs Gini index:**

1. **Balanced dataset:** Use the Gini index for its computational efficiency and simplicity as it does not involve logarithmic calculations.
2. **Imbalanced dataset:** Use Information gain as it is sensitive to class distribution and ensures that the model effectively reduces the uncertainty for the minority class.
3. **Large dataset:** Use the Gini index because of its simplicity.

**Pruning in decision tree:**

Removing or deleting the unnecessary nodes from a tree to get optimal decision tree is known as pruning.

Two types of pruning:

1. Pre-pruning
2. Port-pruning

**1] Pre-pruning:** in this, construction of tree is stopped early based on certain conditions.

Common techniques are:

1. Maximum depth: Limit a maximum depth of the tree.
2. Minimum samples per leaf: Require a minimum number of samples in a leaf node.
3. Minimum Information gain: Set a certain threshold value for information gain.

**2] Post-pruning:** In this, tree is pruned after full construction of the tree, simplifying it without the increase in error.

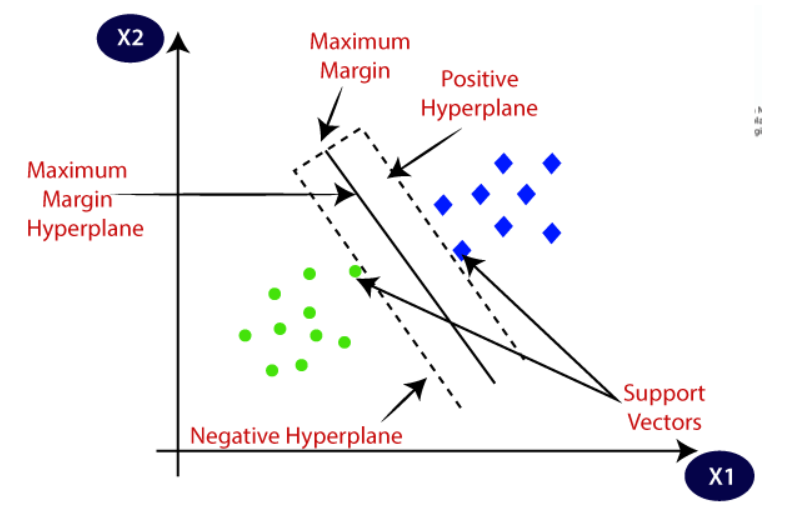
Common techniques:

1. Cost complexity pruning: Prune nodes if the cost complexity(trade-off between tree complexity and error rate) is improved.

Note: For large datasets, use pre-pruning, and for small use post-pruning.

**SVM (Support Vector Machine):**

* The goal of the SVM is to create the best line or decision boundary to segregate n-dimensional feature space into classes, and this decision boundary is called hyperplane.
* SVM chooses extreme points/vectors that is used for creating hyperplane and these two points are called support vectors.



There are two types of SVM:

1. Linear SVM
2. Non-linear SVM

**Hyperplane:** There are multiple decision boundaries, but the best decision boundary that help to classify data points accurately is known as Hyperplane.

**Support Vectors:** The data points that are closest to the hyperplane and which decide the position of the hyperplane are called support vectors.

* SVM tries to maximize the distance between the data points and the hyperplane.

**Kernel Method:**

* The Kernel method is used to convert the input data into higher-dimensional feature space, which makes it simpler to distinguish between classes or generate predictions.
* It uses Kernel functions to map the data into the feature space.
* Kernel approaches in SVMs work on the fundamental principle of implicitly mapping the data into the feature space without computing the coordinates of the data points.

**Characteristics of kernel function:**

1. **Non-negative:** A kernel function is non-negative meaning that it produces non-negative values for all inputs.
2. **Smoothness:** The kernel function is said to be smooth if it produces a smooth transformation of input data into the feature space.
3. **Complexity:** The complexity of the kernel function is an important consideration, as more complex kernel function can lead to overfitting.
4. **Reproducing property:** The kernel function has the property to reproduce input data in the feature space.

**Kernel functions used in SVMs:**

1. **Linear Kernel:**

* It is the simplest and most commonly used kernel function, and it defines the dot product of the input vectors in the feature space.

K(x,y) = x.y

* The dot product is a measure of their similarity or distance in the original feature space.
* The linear kernel function can be used when the data is linearly separable, and in higher dimensional space where a more complex kernel function can lead to overfitting.

1. **Polynomial Kernel:**

* The polynomial kernel employs polynomial functions to convert the data into higher-dimensional feature space.

K(x,y) = (x.y+c)d

Where, c is the constant term and d is the degree of polynomial.

* The polynomial kernel captures more intricate relationships between input features as it is a non-linear kernel function.
* It can be used for both linear data as well as non-linear data.
* The degree of non-linearity of the decision boundary depends on the degree of polynomial.
* Selecting a proper degree of polynomial is the challenging task, as larger degree can lead to overfitting and smaller degree can lead to underfitting.

1. **Gaussian (RBF) kernel:**

* It is a nonlinear kernel function that maps the input data into a higher-dimensional feature space using a Gaussian function.

K(x, y) = exp(-gamma \* ||x - y||^2)

where x and y are the input feature vectors, gamma is a parameter that controls the width of the Gaussian function, and ||x - y||^2 is the squared Euclidean distance between the input vectors.

* The width of the Gaussian function, controlled by the gamma parameter, determines the degree of nonlinearity in the decision boundary.
* One advantage of the Gaussian kernel is its ability to capture complex relationships in the data without the need for explicit feature engineering.
* However, the choice of the gamma parameter can be challenging, as a smaller value may result in under fitting, while a larger value may result in over fitting.

1. **Sigmoid Kernel:**

* This function is equivalent to a two-layer, perceptron model of the neural network, which is used as an activation function for artificial neurons.
* It is also known as hyperbolic tangent kernel.

K(x,y)=tanh(α⟨x,y⟩+c)

A black screen with white text

Description automatically generated

**Naïve Baye’s**

* Naïve is a supervised machine learning algorithm based on Baye’s theorem and used for classification tasks.
* It is mainly used for text classification which includes high-training datasets.
* It is a simple and most effective classification algorithm that helps build fast machine-learning models and make quick predictions.
* It is a probabilistic classifier which means it predicts on the basis of probability of the object.
* Some popular examples of Naïve Baye's are spam filtration, sentimental analysis, and classifying articles.

**Why it is called Naïve Baye’s:**

It comprises of two words:

1. **Naïve:** It is called naïve because it assumes that occurrence of certain feature is independent on the occurrence of other features. Each feature contributes individually to identify a particular class without depending on each other.
2. **Baye’s:** It is called Baye’s because it depends on the principle of Baye’s theorem.

**Baye’s Theorem:**

It is also called Baye’s rule or Baye’s law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.



**P(A|B):** Posterior probability- Probability of hypothesis on the observed event B.

**P(B|A):** Likelihood probability- Probability of the evidence given that the probability of the hypothesis is true.

**P(A):** Prior probability- Probability of hypothesis before observing the evidence.

**P(B):** Marginal probability- Probability of evidence.