What is decision tree? State the advantages, and limitations.

- A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks.
- It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.
- A decision tree is a flowchart-like tree structure where each internal node denotes the feature, branches denote the rules and the leaf nodes denote the result of the algorithm.

Types of Decision Trees

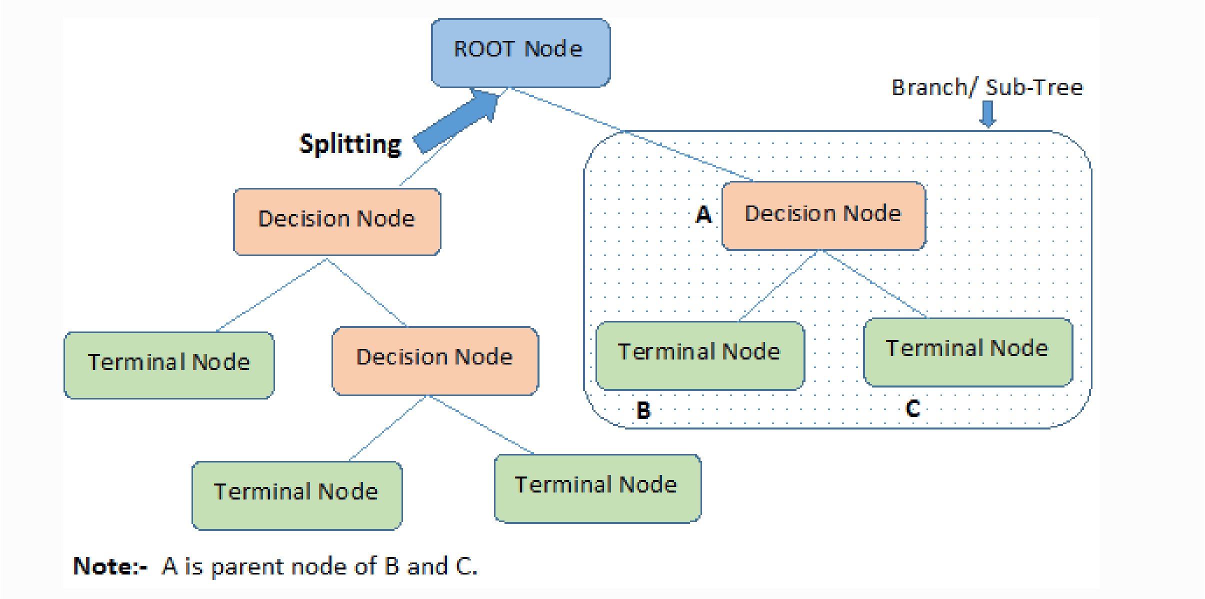
Types of decision trees are based on the type of target variable we have. It can be of two types:

1) Regression based decision trees : Regression is used when the data that we have and the data that we are making predictions on are continuous.
2) Classification based decision trees. : Classification is used when the data we have and the data that we are making predictions on are discrete(or)categorical**.**

Important Terminology related to Decision Trees:

- **Root Node:** It represents the entire population or sample and this further gets divided into two or more homogeneous sets.
- **Splitting:** It is a process of dividing a node into two or more sub-nodes.
- **Decision Node:** When a sub-node splits into further sub-nodes, then it is called the decision node.
- **Leaf / Terminal Node:** Nodes do not split is called Leaf or Terminal node.
- Pruning: When we remove sub-nodes of a decision node, this process is called pruning. You can say the opposite process of splitting.
- **Branch / Sub-Tree:** A subsection of the entire tree is called branch or sub-tree.
- **Parent and Child Node:** A node, which is divided into sub-nodes is called a parent node of sub-nodes whereas sub-nodes are the child of a parent node

Following is the illustration of a decision tree :



Note:- A is parent node of B and C.

Advantages of the Decision Tree:

1. It is simple to understand as it follows the same process which a human follow while making any decision in real-life.
2. It can be very useful for solving decision-related problems.
3. It helps to think about all the possible outcomes for a problem.
4. There is less requirement of data cleaning compared to other algorithms.
5. Great way to choose between best, worst, and likely case scenarios.
6. Can be easily combined with other decision-making techniques.

Disadvantages of the Decision Tree:

1. The decision tree contains lots of layers, which makes it complex.
2. It may have an overfitting issue, which can be resolved using the Random Forest algorithm.
3. For more class labels, the computational complexity of the decision tree may increase.
4. Lack of Smoothness.
5. Limited Expressiveness.
6. Generally, decision trees provide lower prediction accuracy compared to other predictive algorithms.

What is the need of decision tree?

- A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks.
- It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.
- A decision tree is a flowchart-like tree structure where each internal node denotes the feature, branches denote the rules and the leaf nodes denote the result of the algorithm.

Types of Decision Trees

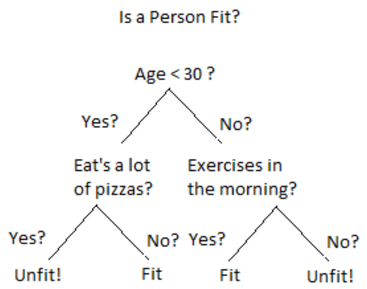
Types of decision trees are based on the type of target variable we have. It can be of two types:

1) Regression based decision trees :Regression is used when the data that we have and the data that we are making predictions on are continuous.

2) Classification based decision trees. : Classification is used when the data we have and the data that we are making predictions on are discrete(or)categorical.

- Decision trees are a popular choice in machine learning for their ability to mimic human thinking and their ease of understanding.
- They are particularly useful for classification problems, where an object needs to be categorized, and for regression issues in predictive analytics.
- The tree-like structure of decision trees allows for a clear analysis of the decision-making process, making them a valuable tool in machine learning.
- Decision trees are used in the context of Non-Intrusive Load Monitoring (NILM) to improve model performance and build trust by providing explanations for the predicted outcomes.
- Decision trees can handle missing values in the dataset without requiring imputation techniques. They can make decisions based on available features at each node
- Unlike other algorithms, decision trees take less time for modeling as they need little analysis, coding, and dummy variables because for each data point, there will be a whole set.
- We can standardize the data even if we don't possess it. We can imbue categorical and numerical data as it works well with both.

Example :



Explain decision tress algorithm.

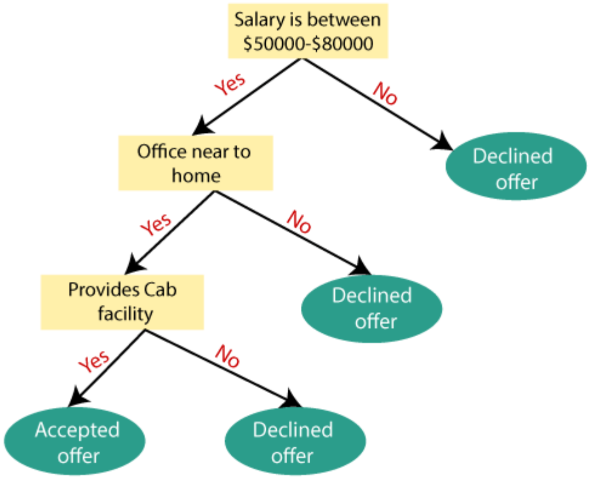
- A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks.
- It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.
- The decision tree operates by analyzing the data set to predict its classification.
- It commences from the tree's root node, where the algorithm views the value of the root attribute compared to the attribute of the record in the actual data set.
- Based on the comparison, it proceeds to follow the branch and move to the next node.
- The algorithm repeats this action for every subsequent node by comparing its attribute values with those of the sub-nodes and continuing the process further. It repeats until it reaches the leaf node of the tree.

The complete mechanism can be better explained through the algorithm given below:

- Step-1: Begin the tree with the root node, says S, which contains the complete dataset.
- Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).
- Step-3: Divide the S into subsets that contains possible values for the best attributes.
- Step-4: Generate the decision tree node, which contains the best attribute.
- Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf nodeClassification and Regression Tree algorithm.

Example:

- Suppose there is a candidate who has a job offer and wants to decide whether he should accept the offer or Not.
- So, to solve this problem, the decision tree starts with the root node (Salary attribute by ASM).
- The root node splits further into the next decision node (distance from the office) and one leaf node based on the corresponding labels.
- The next decision node further gets split into one decision node (Cab facility) and one leaf node.
- Finally, the decision node splits into two leaf nodes (Accepted offers and Declined offer).



What is information gain and entropy in decision tree?

- A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks.
- It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.
- While implementing a Decision tree, the main issue arises that how to select the best attribute for the root node and for sub-nodes.
- So, to solve such problems there is a technique which is called as Attribute selection measure or ASM. By this measurement, we can easily select the best attribute for the nodes of the tree.

There are two popular techniques for ASM, which are:

- **Information Gain**
- **Gini Index**

Information Gain :

- Information gain is a measure of the changes in entropy after the segmentation of a dataset based on an attribute.
- Information gain is a statistical characteristic that measures how well an attribute divides the training instances according to their target types.
- Building a decision tree is discovering attributes that will return the best information gain and the smallest entropy.
- It calculates how much information a feature provides about a class.
- The decision tree algorithm aims to maximize the value of information gain, and a node/attribute with the highest information gain is split first.

The formula is as below: $\text{Information Gain(H, A)} = H - \sum \frac{|H_V|}{|H|} H_v$

where

- A is the specific attribute or class label
- |H| is the entropy of dataset sample S
- |HV| is the number of instances in the subset S that have the value v for attribute A

- Information gain is used in both classification and regression decision trees. In classification, entropy is used as a measure of impurity, while in regression, variance is used as a measure of impurity.
- The information gain calculation remains the same in both cases, except that entropy or variance is used instead of entropy in the formula.

Entropy:

- Entropy is the measure of the degree of randomness or uncertainty in the dataset. In the case of classifications, It measures the randomness based on the distribution of class labels in the dataset.
- The higher the entropy, the harder it will be to solve that information. For example, when we flip a coin, we can't be sure about the outcome. We are simply performing a random act that will provide a random result.

Entropy can be calculated as:

Entropy(s)= -P(yes)log2 P(yes)- P(no) log2 P(no)

Where,

S= Total number of samples

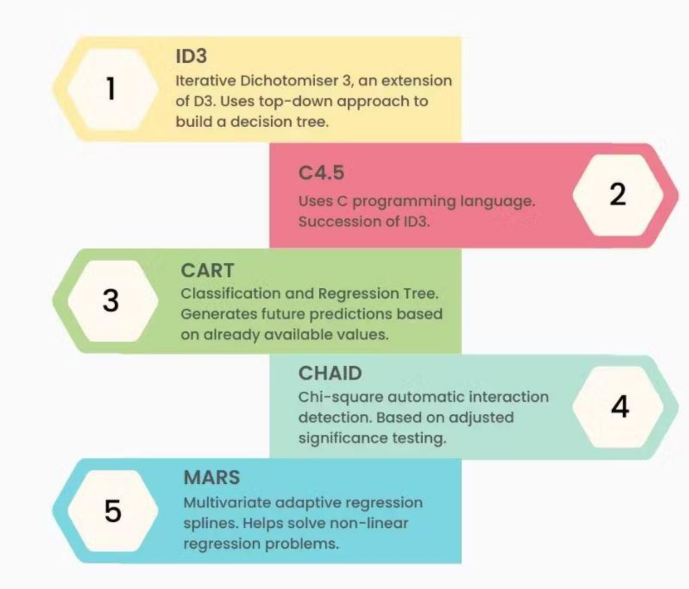
P(yes)= probability of yes

P(no)= probability of no

- The entropy is 0 when the dataset is completely homogeneous, meaning that each instance belongs to the same class. It is the lowest entropy indicating no uncertainty in the dataset sample.
- Therefore, entropy is highest when the distribution of class labels is even, indicating maximum uncertainty in the dataset sample.

Which are algorithms used in decision tree?

- Decision trees use different algorithms to split a node into different sub-nodes. Building sub-nodes increases the homogeneity of the forthcoming sub-nodes.
- The algorithm selection largely depends on the target variable type.


1
ID3
Iterative Dichotomiser 3, an extension of D3. Uses top-down approach to build a decision tree.
2
C4.5
Uses C programming language. Succession of ID3.
3
CART
Classification and Regression Tree. Generates future predictions based on already available values.
4
CHAID
Chi-square automatic interaction detection. Based on adjusted significance testing.
5
MARS
Multivariate adaptive regression splines. Helps solve non-linear regression problems.

The following are the algorithms used in decision trees.

ID3

- ID3 or Iterative Dichotomiser 3 is an algorithm used to build a decision tree by employing a top-down approach.
- The tree is built from the top and each iteration with the best feature helps create a node.

Here are the steps:

- The root node is the start point known as a set S.
- Each iteration of the algorithm will iterate through unused attributes of the root node and calculate the information gain (IG) and entropy (S).
- It will select the attribute with the tiniest entropy or higher information gain.
- We divide set S by choosing the attribute to produce the data subset.
- The algorithm will continue if there is no repetition in the attributes chosen.

C4.5

- The C4.5 algorithm is an improved version of ID3.
- C in the algorithm indicates that it uses C programming language and 4.5 is the algorithm’s version.
- It is one of the more popular algorithms for data mining. It is also used as a decision tree classifier and to generate a decision tree.

CART

- Classification and Regression Tree or CART is a predictive algorithm used to generate future predictions based on already available values.
- These algorithms serve as a base of machine learning algorithms like bagged decision trees, boosted decision trees, or random forests.

There are marked differences between regression trees and classification trees.

- Regression trees: Predict continuous values depending on information sources or previous data. For instance, to predict the price of an item, previous data needs to be analyzed.
- Classification trees: Determine whether an event occurred. It usually has outcomes as either yes or no. This type of decision tree algorithm is often used in real-world decision-making.

CHAID

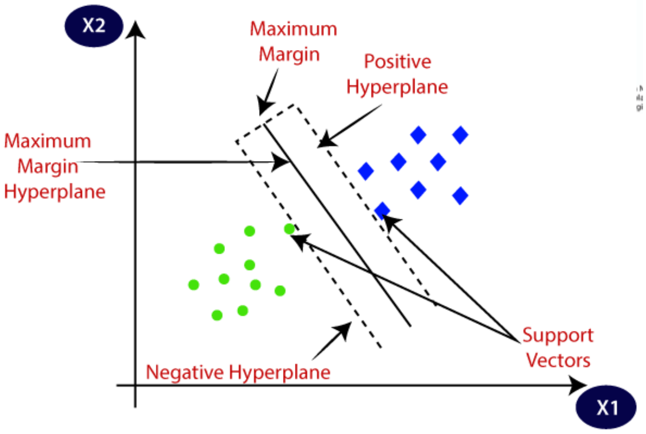
- Chi-square automatic interaction detection (CHAID) is a tree classification method that finds the importance between the parent nodes and root nodes.
- It is measured by adding the squares of standardized differences between the expected and observed frequencies of the target variable.
- It works using the categorical target variables, Success or Failure, and can work on two or more splits.
- If the Chi-square value is high, the statistical importance of the variation of the parent node and root nodes will also be high. It will generate CHAID.

MARS

- Multivariate adaptive regression splines or MARS is a complex algorithm that helps solve non-linear regression problems.
- It lets us find a set of linear functions that provide the best prediction. It is a combination of simple linear regression functions.

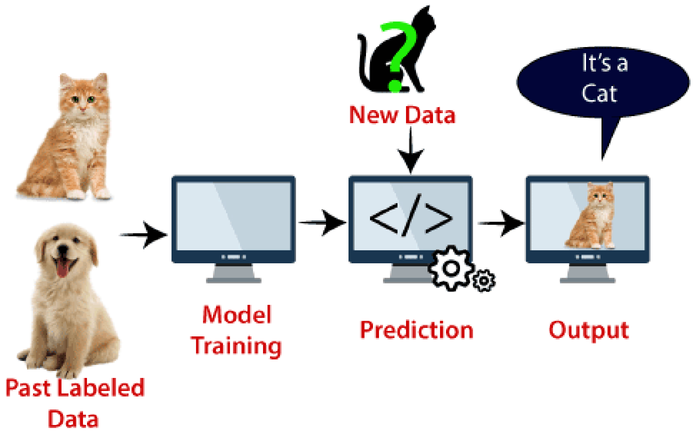
What is SVM? Explain in detail.

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems.
- SVMs can be used for a variety of tasks, such as text classification, image classification, spam detection, handwriting identification, gene expression analysis, face detection, and anomaly detection.
- SVMs are adaptable and efficient in a variety of applications because they can manage high-dimensional data and nonlinear relationships.
- The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
- SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.
- Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



Example:

- Suppose we see a strange cat that also has some features of dogs, so if we want a model that can accurately identify whether it is a cat or dog, so such a model can be created by using the SVM algorithm.
- We will first train our model with lots of images of cats and dogs so that it can learn about different features of cats and dogs, and then we test it with this strange creature.
- So as support vector creates a decision boundary between these two data (cat and dog) and choose extreme cases (support vectors), it will see the extreme case of cat and dog.
- On the basis of the support vectors, it will classify it as a cat. Consider the below diagram:



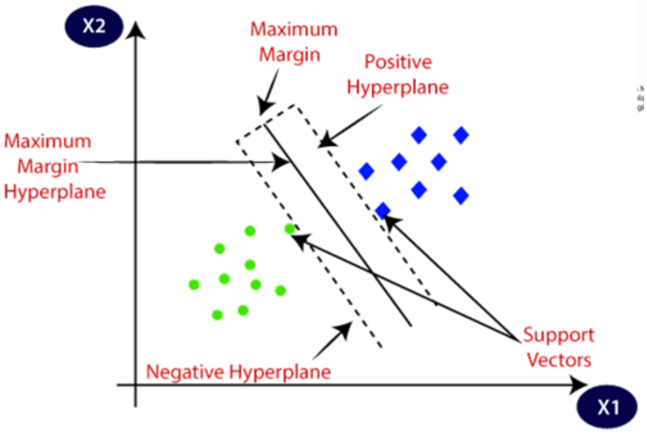
Types of SVM

SVM can be of two types:

- **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
- **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

Explain Hyperplane and Support Vectors in the SVM algorithm

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems.
- The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
- SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.
- Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



Hyperplane and Support Vectors in the SVM algorithm:

Hyperplane:

- There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points.
- This best boundary is known as the hyperplane of SVM.
- The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line.
- And if there are 3 features, then hyperplane will be a 2-dimension plane.
- We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

Support Vectors:

- Support vectors are the closest data points to the hyperplane, which makes a critical role in deciding the hyperplane and margin.
- These points play a vital role in defining the decision boundary and the margin of separation.

Which are the Pros and Cons of SVM Classifiers?

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems.
- SVMs can be used for a variety of tasks, such as text classification, image classification, spam detection, handwriting identification, gene expression analysis, face detection, and anomaly detection.

Advantages of SVM

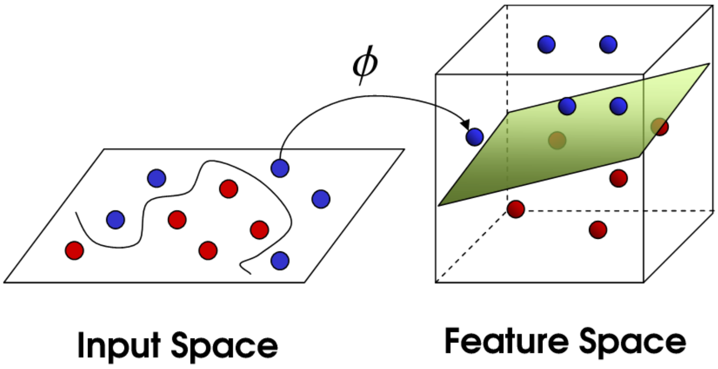
- SVM works better when the data is Linear
- It is more effective in high dimensions
- With the help of the kernel trick, we can
- solve any complex problem
- SVM is not sensitive to outliers
- Can help us with Image classification
- SVM classifiers perform well in high-dimensional space and have excellent accuracy. SVM classifiers require less memory because they only use a portion of the training data.
- SVM performs reasonably well when there is a large gap between classes.
- High-dimensional spaces are better suited for SVM.
- When the number of dimensions exceeds the number of samples, SVM is useful.
- SVM uses memory effectively.

Disadvantages of SVM

- Choosing a good kernel is not easy
- It doesn’t show good results on a big dataset
- The SVM hyperparameters are Cost -C and gamma. It is not that easy to fine-tune these hyper-parameters. It is hard to visualize their impact
- SVM requires a long training period; as a result, it is not practical for large datasets.
- The inability of SVM classifiers to handle overlapping classes is another drawback.
- Large data sets are not a good fit for the SVM algorithm.
- When the data set contains more noise, such as overlapping target classes, SVM does not perform as well.
- The SVM will perform poorly when the number of features for each data point is greater than the number of training data samples.

What is kernel trick in SVM?

- The “Kernel Trick” is a method used in Support Vector Machines (SVMs) to convert data (that is not linearly separable) into a higher-dimensional feature space where it may be linearly separated.
- This technique enables the SVM to identify a hyperplane that separates the data with the maximum margin, even when the data is not linearly separable in its original space.
- The kernel functions are used to compute the inner product between pairs of points in the transformed feature space without explicitly computing the transformation itself.
- This makes it computationally efficient to deal with high dimensional feature spaces.
- The most widely used kernels in SVM are the linear kernel, polynomial kernel, and Gaussian (radial basis function) kernel.
- The choice of kernel relies on the nature of the data and the job at hand. The linear kernel is used when the data is roughly linearly separable, whereas the polynomial kernel is used when the data has a complicated curved border.
- The Gaussian kernel is employed when the data has no clear boundaries and contains complicated areas of overlap.



- The kernel trick is a powerful technique that enables SVMs to solve non-linear classification problems by implicitly mapping the input data to a higher-dimensional feature space.
- By doing so, it allows us to find a hyperplane that separates the different classes of data

What is cost function of SVM?

- The cost function is a measure of the error or loss incurred by the SVM model when it makes predictions on the training data.
- The goal of the SVM algorithm is to minimize this cost function to find the best decision boundary (hyperplane) that can separate the classes in the training data.
- The cost function in SVMs can be expressed as a regularized optimization problem:

$$(\theta) = C \sum_{i=1}^{m} \left[y^{(i)} cost_1(\theta^T x^{(i)}) + (1 - y^{(i)}) cost_0(\theta^T x^{(i)}) \right] + \frac{1}{2} \sum_{i=1}^{n} \theta_i^2$$

- The Cost Function is used to train the SVM. By minimizing the value of J(theta), we can ensure that the SVM is as accurate as possible.
- In the equation, the functions cost1 and cost0 refer to the cost for an example where y=1 and the cost for an example where y=0. For SVMs, cost is determined by kernel (similarity) functions.
- The cost function is a measure of the error or loss incurred by the SVM model when it makes predictions on the training data.