**1. What is the concept of supervised learning? What is the significance of the name?**

Ans: Supervised learning is the process where you have input variables (X) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output.

Y = f(X)

The goal is to approximate the mapping function so well that when you have new input data (X) that you can predict the output variables (Y ) for that data.

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process. We know the correct answers; the algorithm iteratively makes predictions on the training data and is corrected by the teacher.

**2. In the hospital sector, offer an example of supervised learning.**

**Ans:**

Classification is the primary way SVM can do medical data analytics. SVM works as a separator of various data sets in a multidimensional environment. It is capable of performing both multiclass and binary classification of data. Classification of medical data is vital for clinical coding transforming it into standardized statistical code. For example, classification subdivides the data in diagnosis or procedure code to analyze critical information

**Examples:**

* detect cancerous cells based on millions of images
* a study of patient’s response to hospital flu outreach. In this problem, the goal is to find a binary classifier that will predict whether the patient will get vaccinated after reminder, or not (this includes using a different provider for vaccination
* classifies medical data to comprise cancer vs. diabetes, cancer vs. hypertension, and diabetes vs. hypertension in handling medical data.

**3. Give three supervised learning examples.**

Ans:

Ans:Some popular examples of supervised machine learning algorithms are:

* Linear regression for regression problems.
* Random forest for classification and regression problems.
* Support vector machines for classification problems.

**4. In supervised learning, what are classification and regression?**

Ans:

**Classification**: A classification problem is when the output variable is a category, such as red or blue or disease and no disease.

**Regression**: A regression problem is the output variable is a real value, such as dollars or weight.

**5. Give some popular classification algorithms as examples.**

**Ans:**

Popular Classification Algorithms**:**

* Logistic Regression.
* Naive Bayes.
* K-Nearest Neighbors.
* Decision Tree.
* Support Vector Machines.

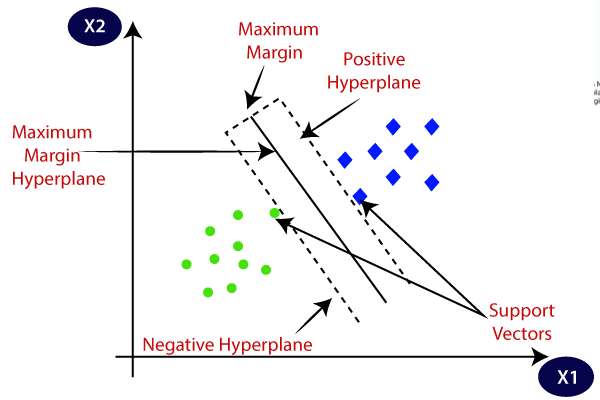
**6. Briefly describe the SVM model.**

**Ans:**

support vector machines (SVMs, also support vector networks) are [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) models with associated learning [algorithms](https://en.wikipedia.org/wiki/Algorithm) that analyze data for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression analysis](https://en.wikipedia.org/wiki/Regression_analysis)**.**

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 support vectors, and hence the 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:



## 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.

**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.

we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized.

* **Support Vectors:**

The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

**7. In SVM, what is the cost of misclassification?**

**Ans:**

Misclassification costs are basically weights applied to specific outcomes. These weights are factored into the model and may actually change the prediction (as a way of protecting against costly mistakes).

In some contexts, certain kinds of errors are more costly than others. For example, it may be more costly to classify a high-risk credit applicant as low risk (one kind of error) than it is to classify a low-risk applicant as high risk (a different kind of error). Misclassification costs allow you to specify the relative importance of different kinds of prediction errors.

support vector machine (SVM) implicitly assumes that the costs of different types of mistakes are the same and minimize the error rate. On the one hand, it is not enough for many practical applications to rely solely on the error rate, which reflects only the average classification ability of a classifier. It is also of great significance to consider the performance of classifiers from the perspective of each sample. On the other hand, many real-world problems, such as credit card fraud detection, intrusion detection, oil-spill detection and cancer diagnosis, usually involve substantially unequal misclassification costs. To solve this problem, many works on the cost-sensitive SVM (CS-SVM) have emerged. The misclassification costs for this model are generally provided by domain experts

**8. In the SVM model, define Support Vectors.**

**Ans:**

Support vectors **are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane**.

**9. In the SVM model, define the kernel.**

**Ans:**

Kernel Function is a method used to take data as input and transform it into the required form of processing data. “Kernel” is used due to a set of mathematical functions used in Support Vector Machine providing the window to manipulate the data

**10. What are the factors that influence SVM's effectiveness?**

**Ans:**

The effectiveness of SVM depends on the selection of kernel, kernel's parameters and soft margin parameter C.

**11. What are the benefits of using the SVM model?**

**Ans:**

**Advantages of support vector machine:**

* Handling high-dimensional data: SVMs are effective in handling high-dimensional data, which is common in many applications such as image and text classification.
* Handling small datasets: SVMs can perform well with small datasets, as they only require a small number of support vectors to define the boundary.
* Modeling non-linear decision boundaries: SVMs can model non-linear decision boundaries by using the kernel trick, which maps the data into a higher-dimensional space where the data becomes linearly separable.
* Robustness to noise: SVMs are robust to noise in the data, as the decision boundary is determined by the support vectors, which are the closest data points to the boundary.
* Generalization: SVMs have good generalization performance, which means that they are able to classify new, unseen data well.
* Versatility: SVMs can be used for both classification and regression tasks, and it can be applied to a wide range of applications such as natural language processing, computer vision and bioinformatics.
* Sparse solution: SVMs have sparse solutions, which means that they only use a subset of the training data to make predictions. This makes the algorithm more efficient and less prone to overfitting.
* Regularization: SVMs can be regularized, which means that the algorithm can be modified to avoid overfitting.
* advantages of SVM and support vector regression include that they can be used to avoid the difficulties of using linear functions in the high-dimensional feature space, and the optimization problem is transformed into dual convex quadratic programs

12. What are the drawbacks of using the SVM model?

Ans:

**Disadvantages of support vector machine:**

* In cases where the number of properties for each data point outstrips the number of training data specimens, the support vector machine will underperform.
* As the support vector classifier works by placing data points, above and below the classifying hyperplane there is *no probabilistic clarification* for the classification.
* Computationally expensive: SVMs can be computationally expensive for large datasets, as the algorithm requires solving a quadratic optimization problem.
* Choice of kernel: The choice of kernel can greatly affect the performance of an SVM, and it can be difficult to determine the best kernel for a given dataset.
* Sensitivity to the choice of parameters: SVMs can be sensitive to the choice of parameters, such as the regularization parameter, and it can be difficult to determine the optimal parameter values for a given dataset.
* Memory-intensive: SVMs can be memory-intensive, as the algorithm requires storing the kernel matrix, which can be large for large datasets.
* Limited to two-class problems: SVMs are primarily used for two-class problems, although multi-class problems can be solved by using one-versus-one or one-versus-all strategies.
* Lack of probabilistic interpretation: SVMs do not provide a probabilistic interpretation of the decision boundary, which can be a disadvantage in some applications.
* Not suitable for large datasets with many features: SVMs can be very slow and can consume a lot of memory when the dataset has many features.
* Not suitable for datasets with missing values: SVMs requires complete datasets, with no missing values, it can not handle missing values.

**13. Notes should be written on**

**1. The kNN algorithm has a validation flaw.**

**Ans:**

Ans: Training a KNN model means basically just loading the training data (as long as you do not do anything fancy like creating a hash table for more efficient neigbor lookup later). There is no optimization, no gradient descent, no weight adjustments etc. However, when doing a prediction you need to solve an optimization problem to find the

K. K nearest neighbors! So that is where the "magic" is happening

In cross-validation, instead of splitting the data into two parts, we split it into 3. Training data, cross-validation data, and test data. Here, we use training data for finding nearest neighbors, we use cross-validation data to find the best value of “K” and finally we test our model on totally unseen test data. This test data is equivalent to the future unseen data points.

**2. In the kNN algorithm, the k value is chosen.**

**Ans:** The k value in the k-NN algorithm defines how many neighbors will be checked to determine the classification of a specific query point. For example, if k=1, the instance will be assigned to the same class as its single nearest neighbor.Defining k can be a balancing act as different values can lead to overfitting or underfitting. Lower values of k can have high variance, but low bias, and larger values of k may lead to high bias and lower variance. The choice of k will largely depend on the input data as data with more outliers or noise will likely perform better with higher values of k. Overall, it is recommended to have an odd number for k to avoid ties in classification, and cross-validation tactics can help you choose the optimal k for your dataset

**3. A decision tree with inductive bias**

**Ans:**

In the case of decision trees, the depth of the trees is the inductive bias. If the depth of the tree is too low, then there is too much generalisation in the model.

Trees that place high information gain attributes close to the root

**14. What are some of the benefits of the kNN algorithm?**

**Ans:**

Advantages of using the k-nearest neighbors algorithm:

* It's easy to understand and simple to implement
* It can be used for both classification and regression problems
* It's ideal for non-linear data since there's no assumption about underlying data
* It can naturally handle multi-class cases
* It can perform well with enough representative data

**15. What are some of the kNN algorithm's drawbacks?**

**Ans:**

Disadvantages of using the k-nearest neighbors algorithm:

* Associated computation cost is high as it stores all the training data
* Requires high memory storage
* Need to determine the value of K
* Prediction is slow if the value of N is high
* Sensitive to irrelevant features

**16. Explain the decision tree algorithm in a few words.**

**Ans:** A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It has a hierarchical, tree structure, which consists of a root node, branches, internal nodes and leaf nodes

**17. What is the difference between a node and a leaf in a decision tree?**

**Node:**

* Specifies some attribute of an instance to be tested
* A decision node has two or more branches

Leaf:

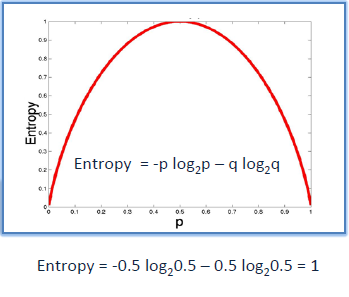
* show the end of a decision path (or outcome). You can always identify a leaf node because it doesn’t split, or branch any further. Just like a real leaf!
* **also called terminal nodes, are nodes that don't split into more nodes**. Leaf nodes are where classes are assigned by majority vote

**18. What is a decision tree's entropy?**

**Ans:** In information theory, the entropy of a random variable is the average level of “information”, “surprise”, or “uncertainty” inherent to the variable's possible outcomes.

In the context of Decision Trees, entropy is **a measure of disorder or impurity in a node**.

Entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is equally divided it has entropy of one

.

**19. In a decision tree, define knowledge gain.**

**Ans:** knowledge gain is Information gain in decisions tree.

The information gained in the decision tree can be defined as **the amount of information improved in the nodes before splitting them for making further decisions**

Information gain calculates the reduction in entropy or surprise from transforming a dataset in some way.

It is commonly used in the construction of decision trees from a training dataset, by evaluating the information gain for each variable, and selecting the variable that maximizes the information gain, which in turn minimizes the entropy and best splits the dataset into groups for effective classification.

Information gain can also be used for feature selection, by evaluating the gain of each variable in the context of the target variable. In this slightly different usage, the calculation is referred to as mutual information between the two random variables.

**20. Choose three advantages of the decision tree approach and write them down.**

**Ans:**

### Advantages of decision trees

* Good for interpreting data in a highly visual way.
* Good for handling a combination of numerical and non-numerical data.
* Easy to define rules, e.g. ‘yes, no, if, then, else…’
* Requires minimal preparation or data cleaning before use.
* Great way to choose between best, worst, and likely case scenarios.
* Can be easily combined with other decision-making techniques.

**21. Make a list of three flaws in the decision tree process.**

**Ans:**

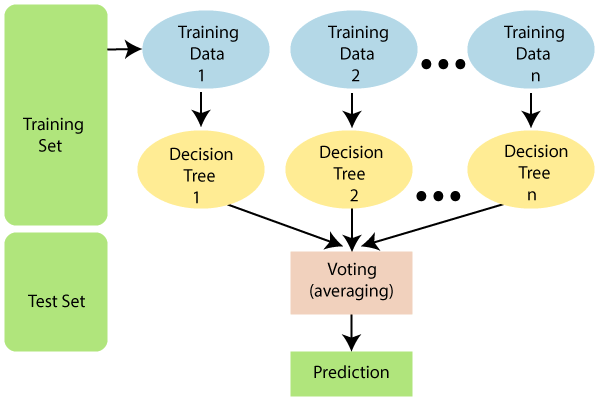
* A small change in the data can cause a large change in the structure of the decision tree causing instability.
* For a Decision tree sometimes calculation can go far more complex compared to other algorithms.
* The Decision Tree algorithm is inadequate for applying regression and predicting continuous values.

**22. Briefly describe the random forest model.**

**Ans:**

Random forest model. combines the output of multiple decision trees to reach a single result.

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset



Assumptions for Random Forest

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random forest classifier:

* There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.
* The predictions from each tree must have very low correlations.

Why use Random Forest?

Below are some points that explain why we should use the Random Forest algorithm:

* It takes less training time as compared to other algorithms.
* It predicts output with high accuracy, even for the large dataset it runs efficiently.
* It can also maintain accuracy when a large proportion of data is missing.

How does the Random Forest algorithm work?

Random Forest works in two-phase first is to create the random forest by combining N decision trees, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.