AML ASSIGNMENT

1. What is semi supervised machine learning? Explain with example?

Semi-supervised machine learning is a combination of supervised and unsupervised machine learning methods. In semi supervised learning an algorithm learns from a dataset that includes both labelled and unlabelled data, usually mostly unlabeled.

A common example of an application of semi supervised learning is a text document classifier. This is the type of situation where semi-supervised learning is ideal because it would be nearly impossible to find a large amount of labeled text documents. This is simply because it is not time efficient to have a person read through entire text documents just to assign it a simple classification. So semi-supervised learning allows the algorithms to learn from a small amount of labelled text documents while still classifying a large amount of unlabelled text documents in the training data.

How semi supervised learning works:-

1. Train the model with the small amount of labelled training data until it gives a good result.
2. Then use it with the unlabelled training dataset to predict the outputs which are pseudo labels since they may not be quite accurate.
3. Link the labels from the labelled training data with the pseudo labels created in the previous step.
4. Link the data inputs in the labelled training data with the inputs in the unlabelled data.
5. Then train the model the same way as you did with the labelled set in the beginning in order to decrease the error and improve the models accuracy.

2. How will you decide the k-value in k-NN algorithm?

* Try with different values and choose the best one.

* k-value must always be odd.

3. How does the efficiency and accuracy of kNN search change as k increases?

* If we have a large number of training set the accuracy should increase.

* The larger the training set less the efficiency.

* The time to calculate the prediction will increase as computational complexity increases.

4. Why is kNN a lazy learning algorithm?

* No learning of the model / algorithm.

* It memorizes the training set.

5. Why is kNN anon-parametric algorithm?

* Because it makes no assumptions about the functional form of problem being solved.

6. When do we use kNN algorithm?

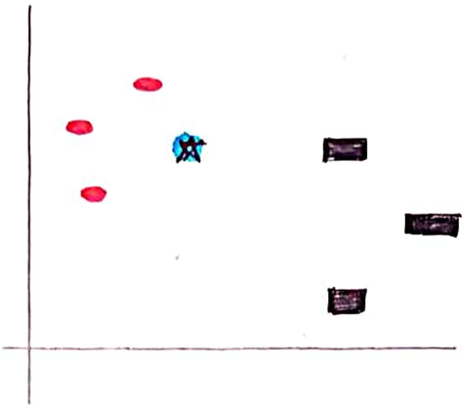
* It is used for both classification and regression problems.

* Widely used for classification problems in industry

* Used for its easy interpretation and low calculation time.

* Hence its predictive power increases.

7. How does the kNN algorithm work to classify the blue star?



- First we need to consider the K-value.
- Then using the euclidean distance formula, the distance from the query point to other points will be calculated.
- The K-nearest points will be considered for classification.
- The blue star will be classified according to the most frequently occuring points.

8. Assume a boolean target function and a 2-D instance space. Determine how the k-nearest neighbour learning algorithm would classify the new instance x_q for $k = 1, 3, 5$ and 7. The + and - signs in the instance space refer to the positive and negative examples respectively.

Distance from query instance	classification
1.00	+
1.35	-
1.40	-
1.60	-
1.90	+
2.00	+
2.20	-
2.40	+
2.80	-

In the case 1 when k=1

The query point x_q will be classified as a positive example because the nearest example to x_q is positive.

when k=3

The query point x_q will be classified as a negative example because the negative examples occur frequently when k=3.

when k=5

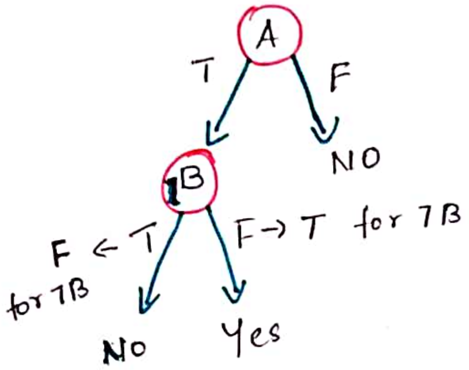
The query point x_q will be classified as a negative example because the negative examples are more frequent when k=5.

when k=7

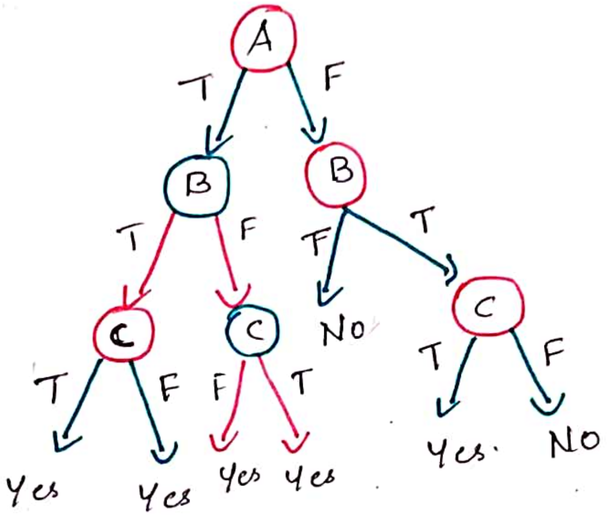
The query point x_q will be classified as negative example because the negative examples are more frequently occuring when k=7. i,e there are 3 positive examples and 4 negative examples. so the negative examples are more frequent.

AML ASSIGNMENT

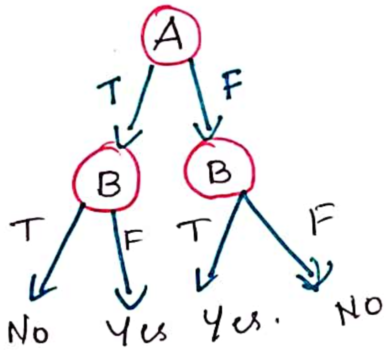
1) $A \wedge \neg B$



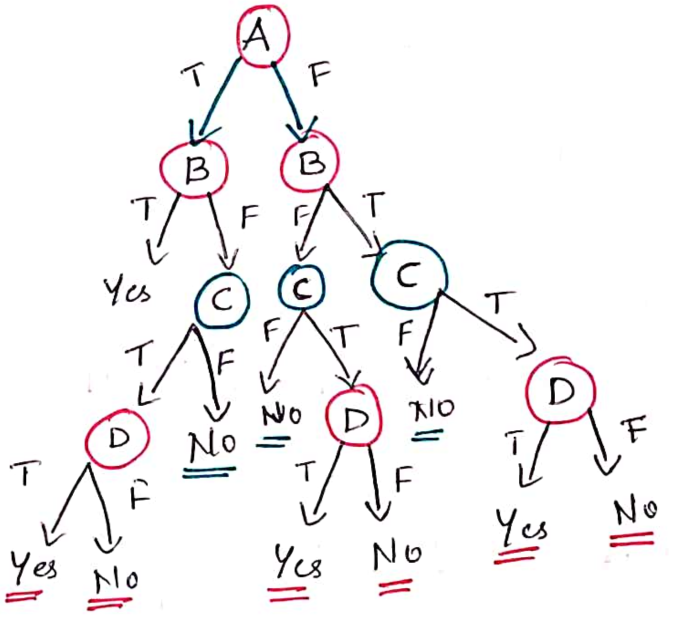
2) $A \vee [B \wedge C]$



3) A XOR B



4) $[A \wedge B] \vee [C \wedge D]$



Q) Distance between two points using Minkowski and Manhattan

Minkowski:

Consider 2 points in 7-dimensional space.

P1: (10, 2, 4, -1, 0, 9, 1)

P2: (14, 7, 11, 5, 2, 2, 18)

If we set p=4 then we cancalculate minkowski distance as follows

distance_pow = (10-14)^4 + (2-7)^4 + (4-11)^4 + (-1-5)^4 +
(0-2)^4 + (9-2)^4 + (1-18)^4

= 4^4 + 5^4 + 7^4 + 6^4 + 2^4 + 7^4 + 17^4

= 256 + 625 + 2401 + 1296 + 16 + 2401 + 83521

= 90516

minkowskidistance = 90516 ^ 1/p → 90516 ^ (1/4)

= 17.3452

How it is used in python library

```
from scipy.spatiale import distance
minkowski_distance = distance.minkowski (P1, P2, p=4)
```

Mahattan

Consider point

A = [2, 3]
B = [4, 1]

manhattan distance = |2-4| + |3-1|

= 4

How it is used in python libaary

```
from scipy.spatial import distance
d = distance.cityblock (a, b)
```

3) Explain different feature extraction techniques.

Feature extraction aims to reduce the number of features in a dataset by creating new features from the existing ones (and then discarding the original features).

Principle Components Analysis (PCA)

PCA is the most used linear dimensionality reduction technique. when using PCA, we take as input our original data and try to find a combination of the input features which can best summarize the original data distribution so that to reduce its original dimensions- PCA is able to do this by maximizing variances and minimizing the reconstruction error by looking at the pair wised distances. In PCA our original data is projected into a set of orthogonal axes and each of the axes gets ranked in the order of importance.

Independent Component Analysis (ICA)

ICA is a linear dimensionality reduction method which takes as input data a mixture of independent components and it aims to correctly identify each of them (deleting all the unnecessary noise). Two input features can be considered independent if both their linear and nonlinear dependence is equal to zero. ICA is commonly used in medical applications such as EEG and fMRI analysis to separate useful signals from unhelpful ones.

Linear Discriminant Analysis (LDA)

LDA is supervised dimensionality reduction technique and Machine Learning Classifier. LDA aims to maximize the distance b/w the mean of each class and minimize the spreading within the class itself. LDA uses therefore within classes and therefore b/w classes as measures. This is a good choice because maximizing the distance b/w the means of each class when projecting the data in lower dimensional space can lead to better classification results (thanks to the reduced overlap b/w the diffrent classes). when using con, is assumed that the input data follows a Gaussian Distribution_ therefore applying LDA to not Gaussian data can possibly lead to poor classificatton results.

Locally Linear Embedding (LLE)

Locally Linear Embedding is a dimensionality reduction technique based on Manifold Learning. A Manifold is an object of D dimen-sions which is embedded in an higher-dimensional space. Manifold Learning aims then to make this object representable in its original D dimensions instead of being represented in an unnecassary greater space.

T-distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE is non-linear dimensionality reduction technique which is typically used to visualize high dimensional datasets. Some of the main applications of t-SNE are Natural language Processing (NLP)

t-SNE works by minimizing the divergence between distribution constituted by the pairwise probability similarities of the input features in the original high dimensional space and its equivalend in the reduced low dimensional space. t-SNE then makes use of the Kullback-Leiber (KL) divergence in order to measure the dissimilarity of the two difference distributions. When t-SNE, the higher dimensional space is modelled using a Gaussian Distibution, while lower dimensional space modelled using a students t distribution.

Autoencoders :- The difference b/w Autoencoders and other dimensionality reduction techniques is that Autoencoders use non-linear transformations to project data from a high dimension to a lower one.

---

4) What do you mean by Hyperparamelers? What is the need for tuning hyperparameters and how hyperparameters are tuned? List the hyperparameters you know?

Hyperparameter is a parameter whose value is used to control the learning process.

Eg:- The k in k nearest neighbors

The C and sigma hyperparameters for support vector machines.

The learning rate for tuning the neural network.

Need for tuning hyperparameters

The goal here is to find an optimal combination of hyperparameters that minimize the predefined loss functions to give better results.

How hyperparameters are tuned ?

Grid Search:- The most basic hyperparameter tuning method. With this technique we simply build a model for each possible combination of all the hyperparameters values provided, evaluating each model and selecting the architecture which produces the best results.

Random search:- Differs from gridsearch in that we longer provide a discrete set of values to explore for each hyperparameter; rather, we provide a statistical distribution for each hyperparameter from which values may be randomly sampled.

Bayesian Optimization :-

It belongs to a class of sequential model-based optimization algorithms that allow for one to use the results of our previous iteration to improve our sampling method of the next experiment. We will initially define a model constructed with hyperparameters λ, which after training is scored v according to some evaluation metric. Next we use the previously evaluated hyperparameter values to compute a posterior expectation as our next model candidate. We iteratively repeat this process until converging to an optimum.

5) What is supervised and unsupervised classifiers. List the classifiers you know?

Supervised learning is a machine learning approach thats defined by its use of labelled datasets. These datasets are designed to train or supervise algorithms into classifying data or predicting outcomes accuratly. Using labelled i/p and o/p the algorithm can measure its accuracy and learn over time.

Classification problems can use algorithms to accuratly assign test data into specific categories such as separating apples from oranges. Or In real world supervised learning algorithms can be used to classify spam in a separate folder from your inbox.

Linear classifiers, K-Nearest Neighbors, Decision Tree, Support Vector Machines.

Unsupervised learning using machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns in data without the need for human intervention. Some examples include kmeans clustering, principle component analysis. Hierarchial clustering.

6) Presence of error and overfitting - Explain with an example?

* Error is the difference in the expected output and the predicted output of the model.
* It is a measure of how well the model performs over a given set of data.
* To calculate error - the loss/cost function (Mean squared Error (or MSE)) is used

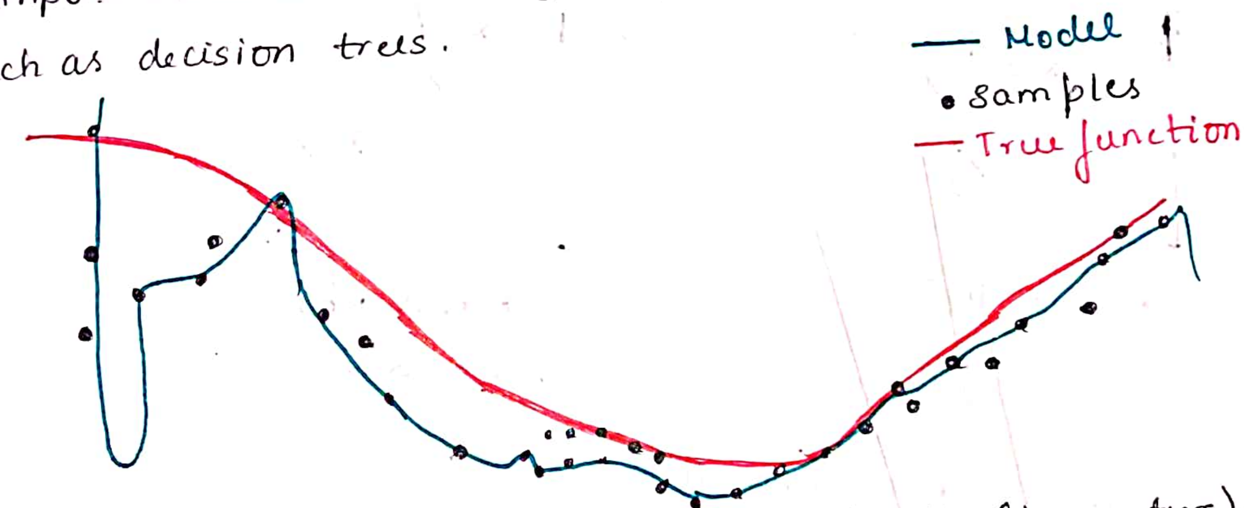
$$MSE = 1/n \sum_{i=1}^{n} (y_i - \bar{y}_i)^2$$

n = Number of data points over which the error is calculated.

y = The expected output of the model
$\bar{y}$ = The predicted output of the model

Overfitting

* When a model learns the pattern and noise in the data to such an extent that it hurts the performance of the model on the new dataset is termed overfitting.
* The model fits the data so well that it interprets noise as patterns in the data.
* Example:- decision boundary is generated by non-linear models such as decision trees.



The model has too much function complexity (parameters) to fit the true function correctly.

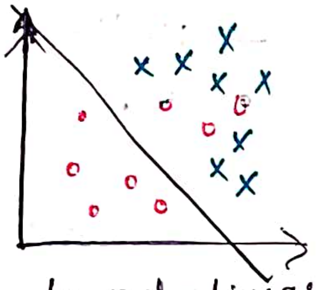
7) Are all non-linear classifiers unsupervised and all linear classifiers supervised?

No not ~~at~~ all non linear classifiers are unsupervised and vice-versa.
Linear classifiers are the algorithm where the plotted data can be classified by drawing a straight line and hence is called linearly separable.

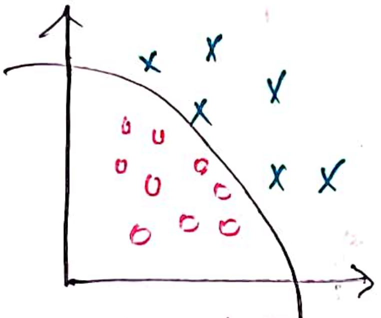
Non linear classifiers cannot separate data using a straight line and hence requires non-linear boundaries between them for classific-ation.

Following are the algorithms which explains first statement:-

Support vector machines : It is a supervised learning algorithm, where we categorize the data using optimal hyperplane for linearly separable data. Svm can also be used for non-linear classifiers. This can be done by tuning its hyperparameters such as kernel, regularization, gamma and margin one such as example is using RBF kernel.



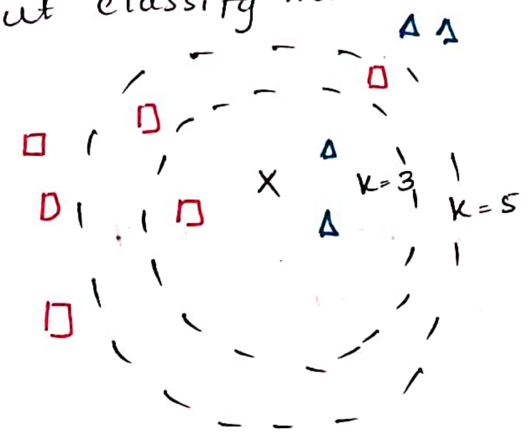
kernel = linear
C = 0.001



Kernel = RBF
C = 0.005

KNN: Supervised learning algorithm

Technique involves classification by considering majority of votes among the "k" closest points. Uses Euclidean / Hamming Distance to classify. But classify non-linear data using its hyperparameters k.



For k=3, the X is classified as Δ.

For k=5, the X is classified as □

PCA: Uses orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values linearly uncorrelated variables called principal

components.
Mapping of higher dimensional data is done to lower dimensional data. We do this by calculating the covariance matrix and eigen vectors and whichever eigen values are less than the threshold, are rejected.

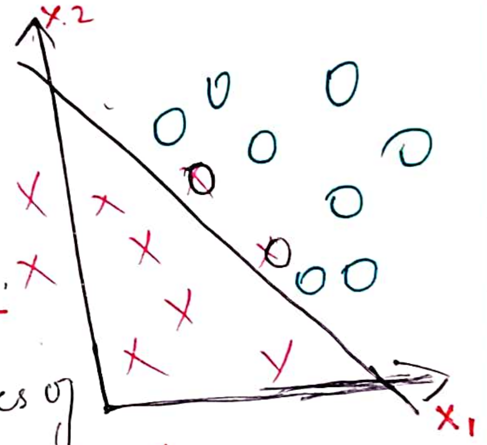
PCA is an unsupervised learning algorithm.

8) What is linearity? (mathematical with eg). And list all the linear classifiers you know?

Let's say you want to classify two classes X and O in the graph. To classify these points we can draw one straight line passing through it.

Where one side all X's are there and on the other side all O's are there.

This is called as linear separable data.



However there can be infinite possibilities of lines which can be drawn to classify the data.

This depends on the classifier we choose. These classifiers are called linear classifiers.

Examples:- Naive Bayes, Logistic Regression, Perceptron, SVM (linear kernel)

9) What is non linearity (mathematical with eg). And list all the non linear classifiers you know?

Let's consider a XOR gate graph which looks as follows, where X1 and X2 are inputs.

These points cannot be separated using a single straight line. Hence this is not linearly separable. Either we can separate it using two lines and a curved boundary. This is an example for "non-linear classification".

In this case we need piece-wise linear (i,e linear in parts).

Examples :- Multi-layer perceptron, Decision Trees, Random Forests, KNN.