MACHINE LEARNING - U3.

Q. Explain Statistical Learning Theory

Ans. Statistical Learning Theory (SLT) is an framework that provides the mathematical function for machine learning. It uses concepts from statistics and functional analysis to explain how a learning algorithm can make valid predictions based on a finite sets of data.
The goal is to understand how well the model is trained on past data to generalize new and unseen data

The Central Problem
Imagine you want to build a model that can predict a home's price based on it's size. You collect data on the square footage and selling price of many homes. A learning algorithm uses this data to find a function (model) that best describes the relationship between the size and price
The central problem is that this function must works well not just for the homes you used to train the model, but also for all future, unseen home (the test data). SLT provides the tools to analyze and garauntee this ability to generalize.

The Bias variance Trade-off, this is the one of the most important concept in SLT and machine learning. It describes the conflict between the two types of errors that a model can make

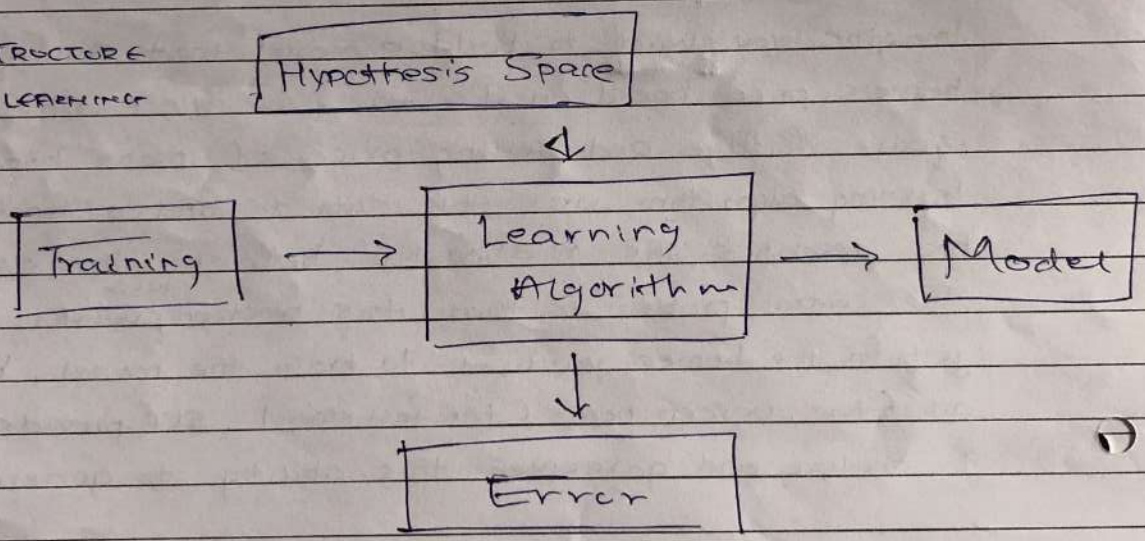
① BIAS:
The error from a model that is too simplistic. A simple model might underfit the training data, essentially and fail to capture important patterns.
Ex: Trying to fit a straight line into complex curved relationship

(i) VARIANCE

The error from a model that is too complex, A complex model can overfit the training data, essentially memorizing the noise and random fluctuations rather than the true underlying relationship. This makes the model perform poorly on new data.

The goal is to find the right balance : A model that is complex enough to capture the true pattern but not so complex that it also memorizes the noise.

•STRUCTURE OF LEARNING



(i) Hypothesis Space

A collection of all possible models (eg: all possible linear models fun or all posible neural model fun) that the learning algorithm can choose from

(ii) Learning Algorithm

A procedure to select the "best" model from the hypothesis space based on the training data

(iv) Error / Risk

A metric used to evaluate how well a model perform.

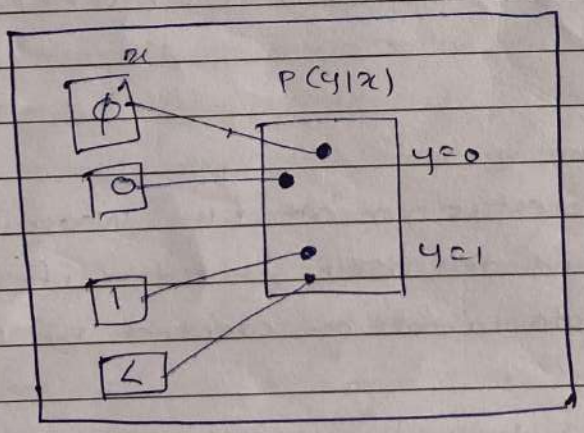
SLT Distinguish between two types

TRAINING ERROR & TEST Error.

This Theory provides garauntee that for a "well-behaved" learning problem, the training error would be approx of test error.

Q. Generative Model & It's Formulation

ANS A generative model is a type of machine learning model that aims to learn underlying patterns or distribution of data into generate new similar kind of data. This is used in unsupervised machine learning ~~algorithm~~ to describe phenom... in ~~data~~, enabling computers to understand real world.



Ex: Imagine you're teaching a child to draw animals. After showing them several pictures of different animals, the child begins to ~~draw~~ understand the general features of each animal. Given some time, the child might draw an animal they've never seen before combining feature they've learned. This is

analagous to how a generative model operates: It learns from the data it is been exposed to and then creates something new based on that knowledge.
The model of distribution of data in statistics can be P(X) or Jointly P(X,Y)

The Formulation of Generative model can be described ~~representing~~ using different different models, each representing a different approach to learning the underlying data distribution. P(X).

1. GANS

This method involves of two AI systems competing with each other.

① Generator: This is the artist, or "forger" whose job is to create new image (or text / Audio / video) that looks as real ass possible

② The Discriminator

This is the detective, or "art critic", whose job is to look at a picture and tell itself, whether if it's a real one from the training data or fake one created by Generata

These Two AI's train together in the game of one-upmanship

- Generator Creates a Fake Image
- Discriminator Tries to catch it

Both model learns and improve from the process. The Generator learns to create fake and Discriminator learns to become better at detecting them.

Q Bayesian Decision Theory

Ans. Bayesian Decision Theory is an statistical approach used to make the best decision under uncertainty, It combines probability to represent uncertain loss function to measure the cost of evidence. The decision rule is to choose the action with the lowest expet loss. Bayes Theorem, it updates beliefs about different atcomes based on available evidence. It provides an optimal and logical framework for decision-making in ML

This process begins with three main concepts.
(i) prior probability (ii) likelihood (iii) posterior probability.
The prior probability $p(w)$ represent what we already believe about an event or class without seeing any new data
Ex: In medical Diagnosis, it could be the overall chance that a person has disease in the general population.
The likelihood $P(x|w)$ tells us how likely it is to observe the data x if the true class is w.
Ex: This means how likely a person's test result is if they actualy have disease.
Finaly the posterior probablity $P(w|x)$ is the upolated belief about the event after sering the data, and can be calculated as

$$P(w|x) = \frac{p(x|w)\, P(w)}{P(x)}$$

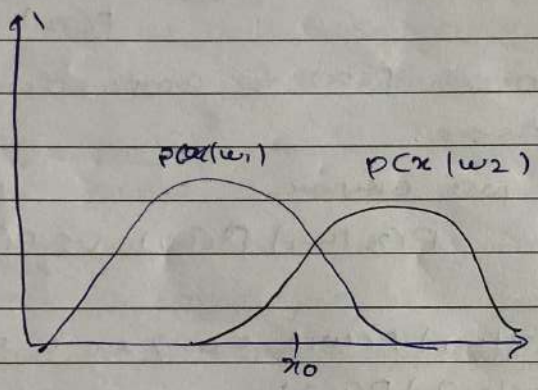
Eventually, the generator becomes so good that the discrim can no longer tell the fakes from the real thing. At this point, the Generator has learned how to create very realistic own content

3. Variational Interfaces – VAE's
This method introduces a probablistic twist to autoencoder architecture. It assumes that data, "x" is generated from some underlying, simpler latent variable, "z", via a decoder network "$p_\theta(x|z)$. Since the true latent distribution $p(z|x)$ is unknown, a VAE uses an encoder network $q_\phi(z|x)$ to approximate it

Q. Bayesian Decision Theory.

Ans. Bayesian theorem is an fundamental in machine learning, especially in the context of Bayesian Inference. It provides a way to update our beliefs about hypothesis on new evidence.
Bayes's Theorem is an fundamental concept of probablity theory, that plays an crucial role in various ML algorithms, especially in the field Bayesian statistics an probablistic modelling. It provides an way to update probabilities based on new evidence or information. In context of ML, Bayes's theorem is often used in probablistic model

This theorem updates our prior belief based on new evidence from the data. Once we know the posterior probab the Bayesian Decision Rule tells us to choose the class with highest posterior probab, meaning the class that is most likely the evidence. Alternatively if the loss or costs are defined, we choose the action that results in smallest expected loss



Graphically, Bayesian Decision Theory can be illustrated using two probability curves. Each curve shows how likely a value of x is for each class. The point where these curves interact is called the decision boundary (x0). If a data point lies on (x0) this mean both classes are likely. If data point lies on one side of the boundary it is classified as class 1 and on other side as class 2. The area under each curve represents the probab of diff outcome for each class.

Beauty of this Theorem is it ensures optimal Decision under uncertainty. Even when outcomes are not garaunteed this model provides the most logical and risk balanced choice based on all available info...

Ex: We want to classify a sample x into one of 2 classes

- Class ω_1 (eg. "spam mail")
- Class ω_2 (eg. "not spam")

Given $P(\omega_1) = 0.4 \quad P(\omega_2) = 0.6$

$P(x|\omega_1) = 0.7 \quad P(x|\omega_2) = 0.2$

$$P(\omega_i|x) = \frac{P(x|\omega_i)\, P(\omega_i)}{P(x)}$$

we Don't need $P(x)$ for comparison, since it is same for both classes.

So we will just compare

$$P(x|\omega_1)\, P(\omega_1) \text{ vs } P(x|\omega_2)\, P(\omega_2)$$

$P(x|\omega_1)\, P(\omega_1) = 0.7 \times 0.4 = 0.28$

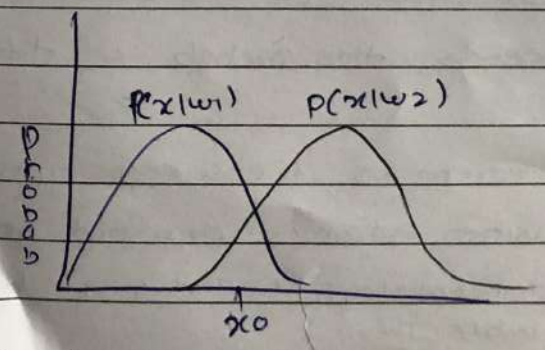
$P(x|\omega_2)\, P(\omega_2) = 0.2 \times 0.6 = 0.12$

Decision Rule

Choose the class with higher posterior product

$0.28 > 0.12 \Rightarrow$ Decide ω_1

So, the feature x is classified as belonging to class ω_1 = Spam mail



Q. Guassian Mixture Model

Ans. Guassian Mixture Model is an powerful statical method used to represent data that seems to come from multiple subgroups or patterns. Instead of Assuming that all data points come from one single source / Guassia ~~model~~ (normal) Distribution, GMM assumes that the data is generated from a mixture of several Guassian Distribution, each represents different cluster or group in the dataset. This makes it an excellent tool for modelling complex, real-world data that cannot be described accurately by one bell-shaped curve

Each Guassian Model component is an mixture defined by three parameter : a mean (μ) that represents the centre of the cluster, a covariance (Σ) that measures the shape of cluster, and a mixing coefficient (π) that tells how much that Guassian ~~model~~ contributes the overall distribution Together these parameters define how mixtures behaves Mathematically, the probability density function for a Guassian Mixture mode is written as

$$p(x) = \sum_{k=1}^{K} \pi_k N(x|\mu_k, \Sigma_k)$$

Here k is the number of Guassian components π_k is the weight (or prior prob) of the k^{th} component $N(x|\mu_k, \Sigma_k)$ is the Guassian Distribution for that Component. The sum of all weightse equals 1, means the model considers all component together as a part of whole.

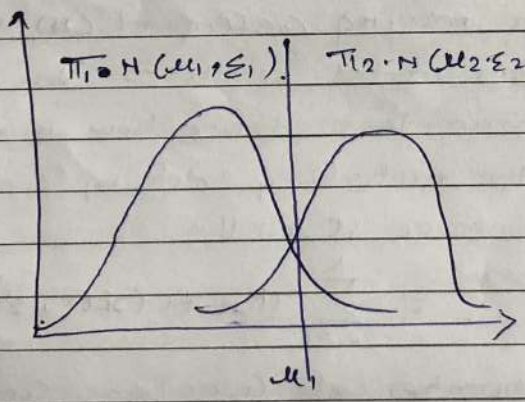
The Goal is to estimate these parameters so that the overall model best fits the observed data. This estimation is typically done using the Expectation-Maximization (EM) algorithm, an iterative process that alternate between two steps

(i) EXPECTATION (E STEP)

Calculates the probability that each data point belongs to each Guassian component based on the current parameters

(ii) MAXIMIZATION (M-STEP)

Updates the parameters of each Guassian distribution and the mixing coefficient to maximize the likehood of the data given these probabilities.



By Repeating these steps until convergence, the GMM learns a set of Guassian distribution that collectively describes the data's structure

Q. ~~Generative Models in a Nutshell~~

Q. Expectation Maximization Method

Ans. The Exception-Maximization (EM) Algorithm is an iterative method used in machine learning to find the maximum likelihood estimates of parameters when data contains hidden or missing variables. It alternates between the two steps (i) Exception - E step

(ii) maximization - M step

(i) E STEP

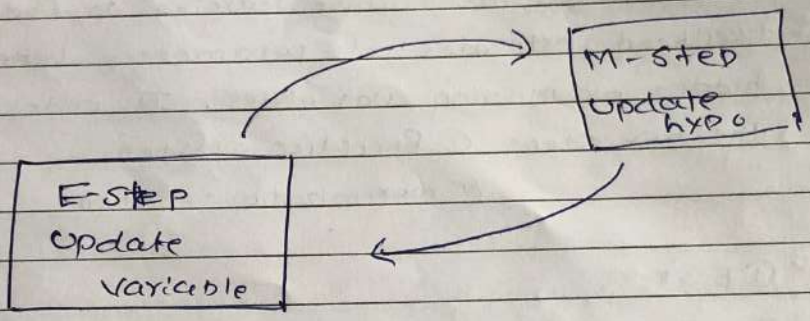
In E-step, the algorithm uses the current parameters estimates to calculate the expected values of the hidden variables. It determines the probability that each data point belongs to each possible hidden state. This means the model estimates how likely each cluster or the component is to have generated a particular observation. Essentially, this step fills in missing information probablistically based on current model

(ii) M STEP

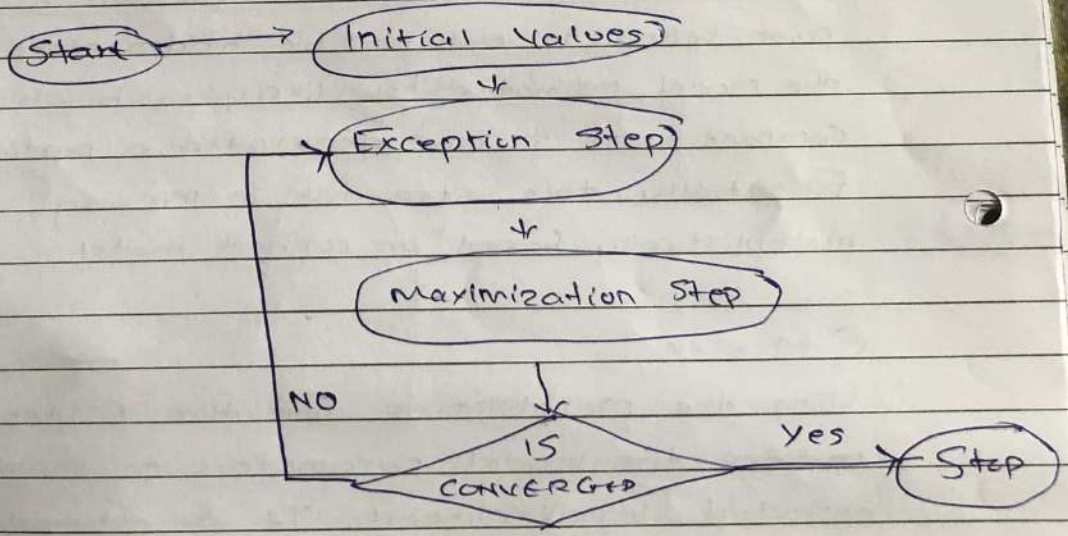
Using the probabilities from the E-step, the M-step updates the model parameters to maximize the expected log-likelihood. It re-estimates parameters like means, variance, or mixing weights so the model better fits the observed data. Each parameter is updated as weighted average, with weights coming from E-step's probablities. This step improves model accuracy before next iteration

These two steps are repeated until convergence, which typically means that.

- The parameter value stops changing significantly or
- The log likelihood improves only by a negligible amount



WORKING OF EM



① Initialization

Begin with initial estimates for the model parameters, such as mean, variance and mixing weights. These serves as the starting point for refining the model through iterative updates.

(iii) E-Step

Calculate the probability that each data point belongs to each hidden or latent component. This step estimates missing data using the current parameter values

(iv) M-Step

Updates the model parameters to maximize the expected log likelihood from the E-step Results

(v) Convergence

Check if changes in parameters or log likelihood are below a small threshold. If yes the algorithm stops otherwise if repeats E & M steps