UNIT - III

3 Bayesian Learning, Computational Learning and Instance Based Learning

3.1 : Bayesian Learning and Bayes Theorem

Q.1 What is Bayesian neural network ?

Ans. : Bayesian Neural Network (BNN) refers to extending standard networks with posterior inference. Standard NN training via optimization is equivalent to Maximum Likelihood Estimation (MLE) for the weights.

Q.2 What are the features of Bayesian learning methods ?

Ans. : 1. Each observed training example can incrementally decrease or increase the estimated probability that a hypothesis is correct.

2. Prior knowledge can be combined with observed data to determine the final probability of a hypothesis.
3. Bayesian methods can accommodate hypotheses that make probabilistic predictions.
4. New instances can be classified by combining the predictions of multiple hypotheses, weighted by their probabilities.
5. Even in cases where Bayesian methods prove computationally intractable, they can provide a standard of optimal decision making against which other practical methods can be measured.

Q.3 What is the practical difficulty in applying Bayesian methods ?

Ans. : Practical difficulty in applying Bayesian methods are as follows :

1. Require initial knowledge of many probabilities. When these probabilities are not known in advance they are often estimated based on background knowledge, previously available data, and assumptions about the form of the underlying distributions.
2. The significant computational cost required to determine the Bayes optimal hypothesis in the general case.

Q.4 What is Bayes theorem ? How to select Hypotheses ?

Ans. : • In machine learning, we try to determine the best hypothesis from some hypothesis space H, given the observed training data D.

- In Bayesian learning, the best hypothesis means the most probable hypothesis, given the data D plus any initial knowledge about the prior probabilities of the various hypotheses in H.
- Bayes theorem provides a way to calculate the probability of a hypothesis based on its prior probability, the probabilities of observing various data given the hypothesis, and the observed data itself.
- Bayes' theorem is a method to revise the probability of an event given additional information.
- Bayes's theorem calculates a conditional probability called a posterior or revised probability.
- Bayes' theorem is a result in probability theory that relates conditional probabilities. If A and B denote two events, $P(A|B)$ denotes the conditional probability of A occurring, given that B occurs. The two conditional probabilities $P(A|B)$ and $P(B|A)$ are in general different.
- This theorem gives a relation between $P(A|B)$ and $P(B|A)$. An important application of Bayes' theorem is that it gives a rule how to update or revise the strengths of evidence-based beliefs in light of new evidence a posteriori.
- A prior probability is an initial probability value originally obtained before any additional information is obtained.

- A posterior probability is a probability value that has been revised by using additional information that is later obtained.
- If A and B are two random variables

$$P(A/B) = \frac{P(B/A)P(A)}{P(B)}$$

- In the context of classifier hypothesis h and training data I.

$$p(h/I) = \frac{P(I/h)P(h)}{P(I)}$$

Where (h) = Prior probability of hypothesis h

(I) = Prior probability of training data I

(h|I) = Probability of h given I

P (I|h) = Probability of I given h

Choosing the Hypotheses

- Given the training data, we are interested in the most probable hypothesis. The learner considers some set of candidate hypotheses H and it is interested in finding the most probable hypothesis $h \in H$ given the observed data D.
- Any such maximally probable hypothesis is called a maximum a posteriori (MAP) hypothesis h_{MAP}
- Maximum a posteriori hypothesis (h_{MAP}).

$$h_{MAP} = \underset{h \in H}{\text{argmax}}\, P(h/I)$$

$$= \underset{h \in H}{\text{argmax}} \frac{P(I/h)P(h)}{P(I)}$$

$$= \underset{h \in H}{\text{argmax}}\, P(I/h)P(h)$$

- If every hypothesis is equally probable, $P(h_i) = P(h_j)$ for all h_i and h_j in H.
- P(I/h) is often called the likelihood of the data I given h. Any hypothesis that maximizes P(I/h) is called a maximum likelihood (ML) hypothesis, h_{ML}.

$$h_{ML} = \underset{h \in H}{\text{argmax}}\, P(I/h)$$

Q.5 At a certain university, 4% of men are over 6 feet tall and 1% of women are over 6 feet tall. The total student population is divided in the ratio 3:2 in favour of women. If a student is selected at random from among all those over six feet tall, what is the probability that the student is a woman ?

Ans. : Let us assume following :

M = {Student is Male},

F = {Student is Female},

T = {Student is over 6 feet tall}.

Given data : P(M) = 2/5,

P(F) = 3/5,

P(T|M) = 4/100

P(T|F) = 1/100.

We require to find P(F|T) ?

Using Bayes' Theorem we have :

$$P(F/T) = \frac{P(T/F)\,P(F)}{P(T/F)\,P(F) + P(T/M)\,P(M)}$$

$$= \frac{\frac{1}{100} \times \frac{3}{5}}{\frac{1}{100} \times \frac{3}{5} + \frac{4}{100} \times \frac{2}{5}} = \frac{\frac{3}{500}}{\frac{3}{500} + \frac{8}{500}}$$

$$P(F/T) = \frac{3}{11}$$

Q.6 Bag contains 5 red balls and 2 white balls. Two balls are drawn successivly without replacement. Draw the probability tree for this.

Sol. : Let R_1 = for the event of getting a red ball on the first draw, W_2 for getting a white ball on the second draw, and so forth. Here's the probability tree.

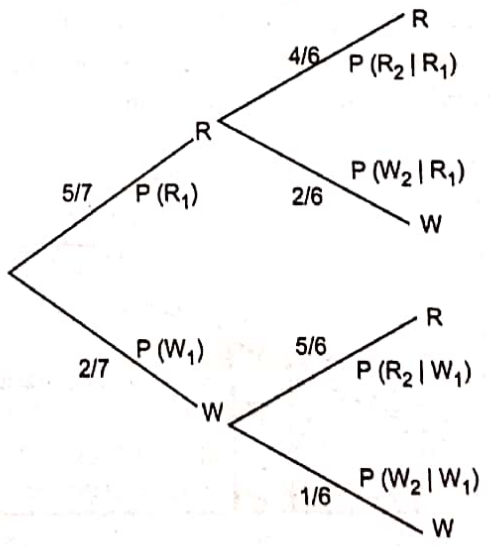

Fig. Q.6.1

3.2 : Maximum Likehood and Least Squared Error Hypotheses

Q.7 What do you mean by least square method ?

Ans. : Least squares is a statistical method used to determine a line of best fit by minimizing the sum of squares created by a mathematical function. A "square" is determined by squaring the distance between a data point and the regression line or mean value of the data set.

Q.8 What is maximum likelihood estimation ?

Ans. : Maximum-Likelihood Estimation (MLE) is a method of estimating the parameters of a statistical model. When applied to a data set and given a statistical model, maximum-likelihood estimation provides estimates for the model's parameters.

Q.9 Briefly discuss least square method. List disadvantages of least square method.

Ans. :

- The method of least squares is about estimating parameters by minimizing the squared discrepancies between observed data, on the one hand, and their expected values on the other.
- Considering an arbitrary straight line, $y = b_0 + b_1 x$, is to be fitted through these data points. The question is "Which line is the most representative" ?
- What are the values of b_0 and b_1 such that the resulting line "best" fits the data points ? But, what goodness-of-fit criterion to use to determine among all possible combinations of b_0 and b_1 ?

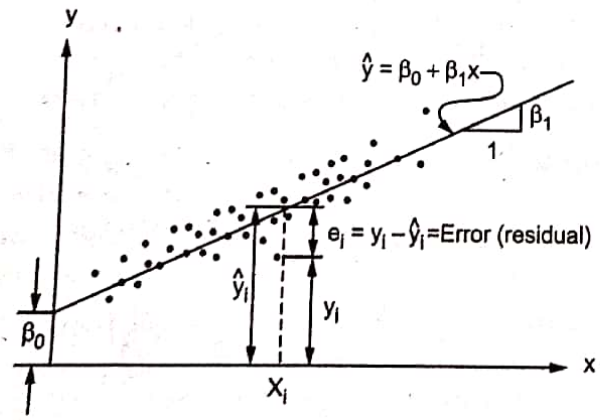

Fig. Q.9.1

- The Least Squares (LS) criterion states that the sum of the squares of errors is minimum. The least-squares solutions yields **y(x)** whose elements sum to 1, but do not ensure the outputs to be in the range [0,1].
- How to draw such a line based on data points observed ? Suppose a imaginary line of $y = a + bx$.
- Imagine a vertical distance between the line and a data point $E = Y - E(Y)$.
- This error is the deviation of the data point from the imaginary line, regression line. Then what is the best values of a and b ? A and b that minimizes the sum of such errors.

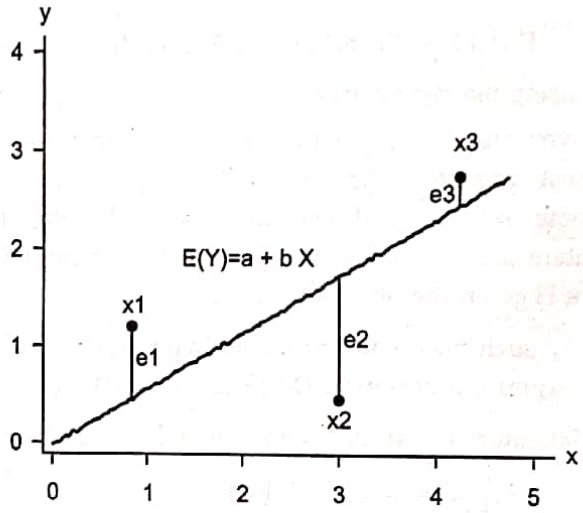

Fig. Q.9.2

- Deviation does not have good properties for computation. Then why do we use squares of deviation ? Let us get a and b that can minimize the sum of squared deviations rather than the sum of deviations. This method is called **least squares.**
- Least squares method minimizes the sum of squares of errors. Such a and b are called least squares estimators i.e. estimators of parameters α and β.
- The process of getting parameter estimators (e.g., a and b) is called **estimation.** Lest squares method is the estimation method of Ordinary Least Squares (OLS).

Disadvantages of least square

1. Lack robustness to outliers.
2. Certain datasets unsuitable for least squares classification.
3. Decision boundary corresponds to ML solution.

Q.10 Fit a straight line to the points in the table. Compute m and b by least squares.

Points	x	y
A	3.00	4.50
B	4.25	4.25
C	5.50	5.50
D	8.00	5.50

Ans. : Represent in matrix form :

$$\begin{bmatrix}3.00 & 1\\ 4.25 & 1\\ 5.50 & 1\\ 8.00 & 1\end{bmatrix}\begin{bmatrix}m\\ b\end{bmatrix} = \begin{bmatrix}4.50\\ 4.25\\ 5.50\\ 5.50\end{bmatrix} + \begin{bmatrix}v_A\\ v_B\\ v_C\\ v_D\end{bmatrix}$$

$$X = \begin{bmatrix}m\\ b\end{bmatrix} = (A^T A)^{-1}(A^T L)$$

$$= \begin{bmatrix}121.3125 & 20.7500\\ 20.7500 & 4.0000\end{bmatrix}^{-1}\begin{bmatrix}105.8125\\ 19.7500\end{bmatrix}$$

$$= \begin{bmatrix}0.246\\ 3.663\end{bmatrix}$$

$$V = AX - L$$

$$= \begin{bmatrix}3.00 & 1\\ 4.25 & 1\\ 5.50 & 1\\ 8.00 & 1\end{bmatrix}\begin{bmatrix}0.246\\ 3.663\end{bmatrix} - \begin{bmatrix}4.50\\ 4.25\\ 5.50\\ 5.50\end{bmatrix} = \begin{bmatrix}-0.10\\ 0.46\\ -0.48\\ 0.13\end{bmatrix}$$

Q.11 Explain with example maximum likelihood estimation.

Ans. : • Maximum-Likelihood Estimation (MLE) is a method of estimating the parameters of a statistical model. When applied to a data set and given a statistical model, maximum-likelihood estimation provides estimates for the model's parameters.

$X_1, X_2, X_3, \ldots, X_n$ have joint density denoted

$f_\theta(x_1, x_2, \ldots, x_n) = f(x_1, x_2, \ldots, x_n|\theta)$

Given observed values $X_1 = x_1, x_2 = x_2, \ldots, X_n = x_n$, the likelihood of θ is the function

$$\text{lik}(\theta) = f(x_1, x_2, \ldots, x_n|\theta)$$

Considered as a function of θ.

- If the distribution is discrete, f will be the frequency distribution function.
- The maximum likelihood estimate of θ is that value of that maximises lik (θ) : It is the value that makes the observed data the most probable.

Examples of maximizing likelihood :

- A random variable with this distribution is a formalization of a coin toss. The value of the random variable is 1 with probability θ and 0 with probability $1 - \theta$. Let X be a Bernoulli random variable and let x be an outcome of X, then we have

$$P(X = x) = \begin{bmatrix}\theta & \text{if } x=1\\ 1-\theta & \text{if } x=0\end{bmatrix}$$

- Usually, we use the notation P(.) for a probability mass and the notation p(.) for a probability density. For mathematical convenience write P(X) as

$$P(X = x) = \theta^x(1-\theta)^{1-x}$$

Q.12 What is gradient search to maximize likelihood in a neural net ?

Ans. : • Develop a method for computers to "understand" speech using mathematical methods. For a D-dimensional input vector o, the Gaussian distribution with mean μ and positive definite covariance matrix $\sum$ can be expressed as

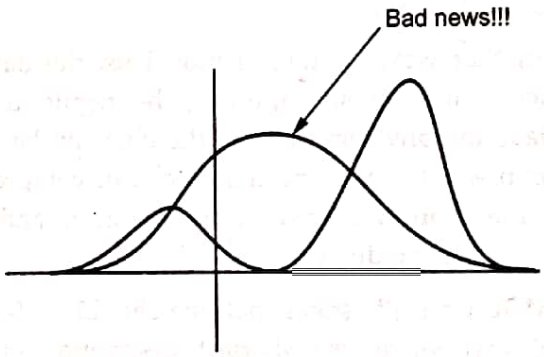

Fig. Q.12.1

$$N(o, \mu \sum) = \frac{1}{(2\pi)^{\frac{D}{2}} |\sum - 1|^{1/2}} e^{-1/2 (0-\mu)^T}$$

$$\sum^{-1} (0-\mu)$$

- The distribution is completely described by the D parameters representing μ and the D(D + 1)/2 parameters representing the symmetric covariance matrix $\sum$.
- Single Gaussian may do a bad job of modeling distribution in any dimension :

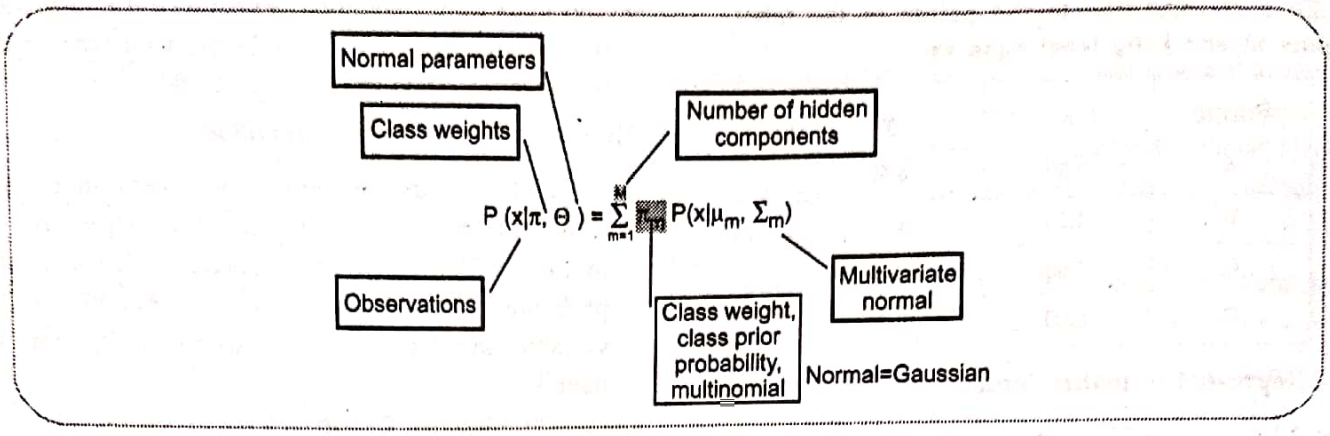

Fig. Q.12.2

- Solution : Mixtures of Gaussians is a solution for this problem.
- A formalism for modeling a probability density function as a sum of parameterized functions.

3.3 : Minimum Description Length Principle

Q.13 Explain minimum description length principle.

Ans. : • The Minimum Description Length (MDL) criteria in machine learning says that the best description of the data is given by the model which compresses it the best.

- Put another way, learning a model for the data or predicting it is about capturing the regularities in the data and any regularity in the data can be used to compress it. Thus, the more we can compress a data, the more we have learnt about it and the better we can predict it.
- The MDL principle states that one should prefer the model that yields the shortest description of the data when the complexity of the model itself is also accounted for.
- The Minimum Description Length principle is motivated by interpreting the definition of h_{MAP} in the light of basic concepts from information theory. Consider definition of h_{MAP}.

$$h_{MAP} = \underset{h \in H}{\text{argmax}}\ P(I/h)p(h)$$

- The MDL principle recommends choosing the hypothesis that minimizes the sum of these two description lengths.
- Assume the codes C_1 and C_2 to represent the hypothesis and the data given the hypothesis, we can state the MDL principle as

$$h_{MDL} = \underset{h \in H}{\text{argmin}}\ L_{C_1}(h) + L_{C_2}(D/h)$$

- The above analysis shows that if we choose C_1 to be the optimal encoding of hypotheses C_H and if we choose C_2 to be the optimal encoding $C_{I/h}$ then $h_{MDL} = h_{MAP}$.

3.4 : Bayes Optimal Classifier and Gibbs Algorithm

Q.14 Define Gibbs algorithm.

Ans. : The Gibbs algorithm defined as follows :

1. Choose a hypothesis **h** from H at random, according to the posterior probability distribution over H.
2. Use **h** to predict the classification of the next instance **x**.

Q.15 What is the Bayes optimal classifier ?

Ans. : • Bayes classifier is a classifier that minimizes the error in a probabilistic manner. If it is Bayes optimal, then the errors are weighed using the join probability distribution between the input and the output sets.

- The Bayes error is then the error of the Bayes classifier.

3.5 : Naïve Bayes Classifier

Q.16 What is Naïve Bayes Classifiers ?

Ans. : • Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong independence assumptions between the features. It is highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem.

- A Naive Bayes Classifier is a program which predicts a class value given a set of attributes.
- For each known class value,
  1. Calculate probabilities for each attribute, conditional on the class value.
  2. Use the product rule to obtain a joint conditional probability for the attributes.
  3. Use Bayes rule to derive conditional probabilities for the class variable.
- Once this has been done for all class values, output the class with the highest probability.
- Naive Bayes simplifies the calculation of probabilities by assuming that the probability of each attribute belonging to a given class value is independent of all other attributes. This is a strong assumption but results in a fast and effective method.
- The probability of a class value given a value of an attribute is called the conditional probability. By multiplying the conditional probabilities together for each attribute for a given class value, we have a probability of a data instance belonging to that class.

3.6 : Bayesian Belief Networks

Q.17 Describe Bayesian belief network.

Ans. : Bayesian belief network describes the probability distribution governing a set of variables by specifying a set of conditional independence assumptions along with a set of conditional probabilities.

Q.18 Explain with example how Bayesian belief network is represented ?

Ans. :

- Bayesian belief networks represent the full joint distributi n over the variables more compactly with a smaller number of parameters.
- It take advantage of conditional and marginal independences among random variables
- A and B are independent then P(A, B) = P(A)P(B)
- A and B are conditionally independent given C

P(A, B | C) = P(A | C)P(B | C)

P(A | C, B) = P(A | C)

- **Example : Alarm system example.**
- Assume your house has an alarm system against burglary. You live in the seismically active area and the alarm system can get occasionally set off by an earthquake.
- You have two neighbors, Mary and John, who do not know each other. If they hear the alarm they call you, but this is not guaranteed.
- We want to represent the probability distribution of events : Burglary, Earthquake, Alarm, Mary calls and John calls

Causal relations :

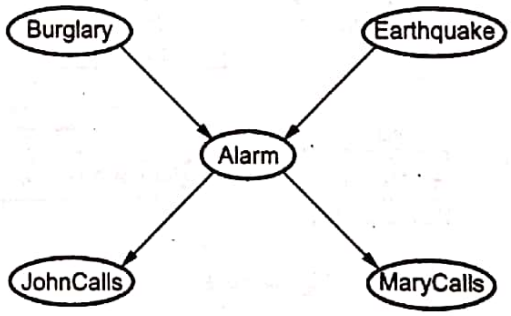

Fig. Q.18.1

Directed acyclic graph :

- Nodes = Random variables

Burglary, Earthquake, Alarm, Mary calls and John calls

- Links = Direct (causal) dependencies between variables.

The chance of Alarm is influenced by Earthquake, The chance of John calling is affected by the Alarm.

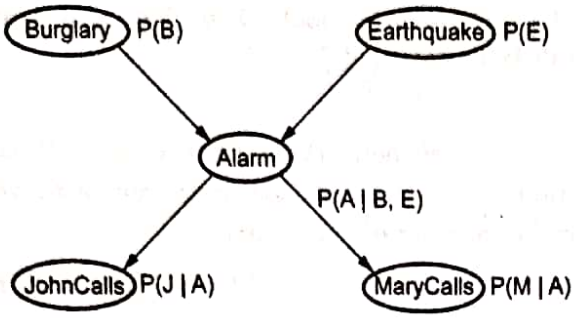

Fig. Q.18.2

Local conditional distributions : Relate variables and their parents

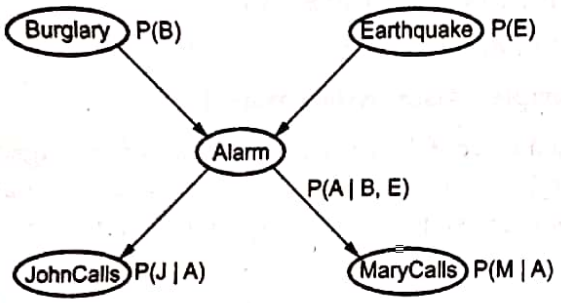

Fig. Q.18.3

Bayesian belief network :

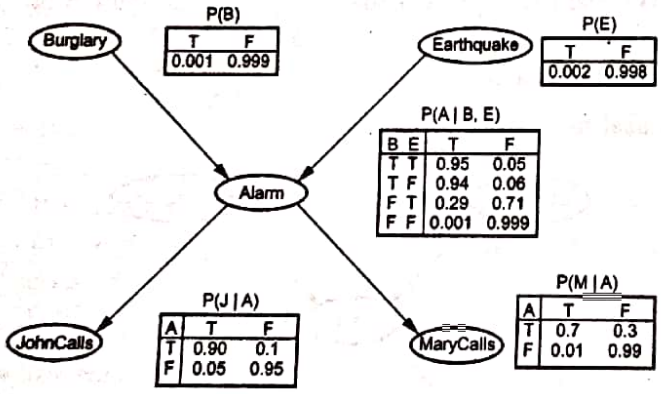

Fig. Q.18.4

3.7 : The EM Algorithm

Q.19 Write short note on EM algorithm.

Ans. : • Expectation-Maximization (EM) is an iterative method used to find maximum likelihood estimates of parameters in probabilistic models, where the model depends on unobserved, also called latent, variables.

- EM alternates between performing an expectation (E) step, which computes an expectation of the likelihood by including the latent variables as if they were observed, and maximization (M) step, which computes the maximum likelihood estimates of the parameters by maximizing the expected likelihood found in the E step.
- The parameters found on the M step are then used to start another E step, and the process is repeated until some criterion is satisfied. EM is frequently used for data clustering like for example in Gaussian mixtures.
- In the **Expectation step**, find the expected values of the latent variables (here you need to use the current parameter values).
- In the **Maximization step**, first plug in the expected values of the latent variables in the log-likelihood of the augmented data. Then maximize this log-likelihood to reevaluate the parameters.
- Expectation-Maximization (EM) is a technique used in point estimation. Given a set of observable variables X and unknown (latent) variables Z we want to estimate parameters θ in a model.
- The expectation maximization (EM) algorithm is a widely used maximum likeli-hood estimation procedure for statistical models when the values of some of the variables in the model are not observed
- The EM algorithm is an elegant and powerful method for finding the maximum likelihood of models with hidden variables. The key concept in the EM algorithm is that it iterates between the expectation step (E-step) and maximization step (M-step) until convergence.
- In the E-step, the algorithm estimates the posterior distribution of the hidden variables Q given the observed data and the current parameter settings; and in the M-step the algorithm calculates the ML parameter settings with Q fixed.
- At the end of each iteration the lower bound on the likelihood is optimized for the given parameter setting (M-step) and the likelihood is set to that bound (E-step), which guarantees an increase in the likelihood and convergence to a local maximum, or global maximum if the likelihood function is unimodal.

- Generally, EM works best when the fraction of missing information is small and the dimensionality of the data is not too large. EM can require many iterations, and higher dimensionality can dramatically slow down the E-step.
- EM is useful for several reasons: conceptual simplicity, ease of implementation, and the fact that each iteration improves $l(\theta)$. The rate of convergence on the first few steps is typically quite good, but can become excruciatingly slow as you approach local optima.
- Sometimes the M-step is a constrained maximization, which means that there are constraints on valid solutions not encoded in the function itself.
- Expectation maximization is an effective technique that is often used in data analysis to manage missing data. Indeed, expectation maximization overcomes some of the limitations of other techniques, such as mean substitution or regression substitution. These alternative techniques generate biased estimates and, specifically, underestimate the standard errors. Expectation maximization overcomes this problem.

3.8 : Introduction of Computational Learning Theory

Q.20 What is computational learning theory ?

Ans. : • Computational learning theory provides a formal framework in which to precisely formulate and address questions regarding the performance of different learning algorithms so that careful comparisons of both the predictive power and the computational efficiency of alternative learning algorithms can be made.

- Three key aspects that must be formalized are the way in which the learner interacts with its environment the definition of successfully completing the learning task and a formal definition of efficiency of both data usage (sample complexity) and processing time (time complexity)

3.9 : Probably Learning an Approximately Correct Hypothesis

Q.21 Define Probably Approximately Correct Learning.

Ans. : A concept class C is said to be PAC learnable using a hypothesis class H if there exists a learning algorithm L such that for all concepts in C, for all instance distributions D on an instance space X, $\forall \varepsilon$, $\delta (0 < \varepsilon, \delta < 1)$, L, when given access to the Example oracle, produces, with probability at least $(1 - \delta)$, a hypothesis h from H with error no more than ε.

Q.22 Define consistent learner.

Ans. : A consistent learner is one that returns some hypothesis h from the hypothesis class H that is consistent with a random sequence of m examples. A consistent learner is a MAP learner, if all hypotheses are a-priori equally likely.

Q.23 Discuss briefly Probably Approximately Correct Learning.

Ans. : • PAC is a nice formalism for deciding how much data you need to collect in order for a given classifier to achieve a given probability of correct predictions on a given fraction of future test data.

- To understand what this model is all about, it's probably easiest just to give an example. Say there's a hidden line on the chalk board.
- Given a point on the board, we need to classify whether it's above or below the line. To help, we'll get some sample data, which consists of random points on the board and whether each point is above or below the line.
- After seeing, say, twenty points, you won't know exactly where the line is, but you'll probably know roughly where it is. And using that knowledge, you'll be able to predict whether most future points lie above or below the line.
- Suppose we have agreed that predicting the right answer "most of the time" is okay. Is any random choice of twenty points going to give you that ability ? No, because you could get really unlucky with the sample data, and it could tell you almost nothing about where the line is. Hence the "Probably" in PAC.

- X is the set of all possible examples. D is the distribution from which the examples are drawn
- H is the set of all possible hypotheses, $c \in H$
- m is the number of training examples. Then

error(h) = Pr(h(x) ≠ c(x) | x is drawn from X with D)

where h is approximately correct if *error(h)* $\leq \varepsilon$

- *Hypothesis h(X) is consistent with m examples and has an error of at most with probability* $1 - \delta$ This is a worst-case analysis. Note that the result is independent of the distribution D.
- **Curse of dimensionality :** If the number of features d is large, the number of samples n, may be too small for accurate parameter estimation.
- For accurate estimation, n should be much bigger than d^2, otherwise model is too complicated for the data, overfitting.

3.10 : Sample Complexity for Infinite Hypothesis Spaces

Q.24 Explain VC dimension.

Ans. : Vapnik-Chervonenkis (VC) dimension provides a measure of the complexity of a space of functions, and which allows the probably approximately correct framework to be extended to spaces containing an infinite number of functions.

- The Vapnik-Chervonenkis dimension is a measure of the complexity or capacity of a class of functions f(α). The VC dimension measures the largest number of examples that can be explained by the family f(α).
- The Vapnik-Chervonenkis dimension, VC(H), of hypothesis space H defined over instance space X is the size of the largest finite subset of X shattered by H. If arbitrarily large finite sets of X can be shattered by H, then VC(H) = ∞.
- The basic argument is that high capacity and generalization properties are at odds :

1. If the family f(α) has enough capacity to explain every possible dataset, we should not expect these functions to generalize very well.
2. On the other hand, if functions f(α) have small capacity but they are able to explain our particular dataset, we have stronger reasons to believe that they will also work well on unseen data.

- **Shattering a set of examples :** Assume a binary classification problem with N examples in R^D and consider the set of $2^{|N|}$ possible dichotomies. For instance, with N = 3 examples, the set of all possible dichotomies is **{(000), (001), (010), (011), (100), (101), (110), (111)}**. A class of functions f(α) is said to shatter the dataset if, for every possible dichotomy, there is a function in f(α) that models it.
- Consider as an example a finite concept class $C = \{c_1, \ldots, c_4\}$ applied to three instance vectors with the results :

	X_1	X_2	X_3
c_1	1	1	1
c_2	0	1	1
c_3	1	0	0
c_4	0	0	0

Then :

$\pi_c(\{X_1\}) = \{(0), (1)\}$ shattered

$\pi_c(\{X_1, X_3\}) = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$ shattered

$\pi_c(\{X_2, X_3\}) = \{(0, 0), (1, 1)\}$ not shattered

- The **VC dimension VC(f)** is the size of the largest dataset that can be shattered by the set of functions f(α). If the VC dimension of (α) is h, then there exists at least one set of h points that can be shattered by (α), but in general it will not be true that every set of h points can be shattered.
- **Example of VCdim : axis aligned rectangles**
- If we had five points, then at most four of the points determine the minimal rectangle that contains the whole set. Then no rectangle is consistent with the labeling that assigns these four boundary points "+" and assigned the remaining point a "-". Therefore,

VCdim (axis-aligned rectangles ε R^2) = 4

- The VC dimension cannot be accurately estimated for non-linear models such as neural networks. The

VC dimension may be infinite requiring infinite amount of data.

3.11 : The Mistake Bound Model of Learning

Q.25 Define mistake bound model.

Ans. : Algorithm A has mistake-bound M for learning class C if A makes at most M mistakes on any sequence that is consistent with a function in C.

Q.26 Explain Weighted Majority Algorithm.

Ans. :

- A classifier combination method
- Takes a weighted vote among a pool of prediction algorithms, e.g., alternative hypotheses in H, or alternative learning algorithms
- It begins by weighting each algorithm by 1
- Whenever an algorithm misclassifies its weight is decreased by β, where $0 < \beta < 1$.
- If A is any set of n prediction algorithms.
- If k is the minimum number of mistakes made by any algorithm in A.
- The number of mistakes made over any training sequence is at most.

3.12 : Introduction of Instance-based Learning Methods

Q.27 What is instance-based learning ? List its advantages and disadvantages.

Ans. : • All learning methods presented so far construct a general explicit description of the target function when examples are provided.

- Instance-based learning methods simply store the training examples instead of learning explicit description of the target function.
- Generalizing the examples is postponed until a new instance must be classified.
- When a new instance is encountered, its relationship to the stored examples is examined in order to assign a target function value for the new instance.
- Instance-based learning includes nearest neighbor, locally weighted regression and case-based reasoning methods.
- Instance-based methods are sometimes referred to as lazy learning methods because they delay processing until a new instance must be classified.
- A key advantage of lazy learning is that instead of estimating the target function once for the entire instance space, these methods can estimate it locally and differently for each new instance to be classified.

Advantages :

1. Instead of estimating for the whole instance space, local approximations to the target function are possible
2. Especially if target function is complex but still decomposable.

Disadvantages :

1. Classification costs are high.
2. Typically all attributes are considered when attempting to retrieve similar training examples.

3.13 : k-Nearest Neighbour Learning

Q.28 What is K-Nearest Neighbour Methods ?

Ans. : • The K-nearest neighbor (KNN) is a classical classification method and requires no training effort, critically depends on the quality of the distance measures among examples.

- The KNN classifier uses Mahalanobis distance function. A sample is classified according to the majority vote of its nearest K training samples in the feature space. Distance of a sample to its neighbors is defined using a distance function

Q.29 What is Euclidean distance ?

Ans. : • The Euclidean distance is the most common distance metric used in low dimensional data sets. It is also known as the L_2 norm.

- The Euclidean distance is the usual manner in which distance is measured in real world.

Q.30 Define Mahalanobis distance.

Ans. : Mahalanobis distance is also called quadratic distance.

- Mahalanobis distance is a distance measure between two points in the space defined by two or more correlated variables. Mahalanobis distance takes the correlations within a data set between the variableinto considerations.
- While Euclidean metric is useful in low dimensions, it doesn't work well in high dimensions and for categorical variables.
- The drawback of Euclidean distance is that it ignores the similarity between attributes. Each attribute is treated as totally different from all of the attributes.

Q.31 List out the steps that need to be carried out during the KNN algorithm.

Ans. : Steps are as follows :

a. Divide the data into training and test data.

b. Select a value K.

c. Determine which distance function is to be used.

d. Choose a sample from the test data that needs to be classified and compute the distance to its n training samples

e. Sort the distances obtained and take the k-nearest data samples.

f. Assign the test class to the class based on the majority vote of its K neighbors

Q.32 What are the advantages and disadvantages of KNN ?

Ans. : Advantages

1. The KNN algorithm is very easy to implement.
2. Nearly optimal in the large sample limit.
3. Uses local information, which can yield highly adaptive behavior.
4. Lends itself very easily to parallel implementations.

Disadvantages

1. Large storage requirements.
2. Computationally intensive recall.
3. Highly susceptible to the curse of dimensionality.

Q.33 Which are the performance factors that influence KNN algorithm ?

Ans. : The performance of the KNN algorithm is influenced by three main factors :

1. The distance function or distance metric used to determine the nearest neighbors.
2. The decision rule used to derive a classification from the K-nearest neighbors.
3. The number of neighbors used to classify the new example.

3.14 : Locally Weighted Regression

Q.34 Write short note on Locally Weighted Regression.

Ans. : • KNN forms local approximation to f for each query point xq.

- Why not form an explicit approximation f(x) for region surrounding xq? Use Locally Weighted Regression.
- Locally Weighted Regression (LWR) is a memory-based method that performs a regression around a point of interest using only training data that are ``local" to that point.
- Locally weighted regression uses nearby or distance-weighted training examples to form this local approximation to f.
- We might approximate the target function in the neighborhood surrounding x, using a linear function, a quadratic function, a multilayer neural network.
- The phrase "locally weighted regression" is called
  a. local because the function is approximated based only on data near the query point,
  b. weighted because the contribution of each training example is weighted by its distance from the query point, and
  c. regression because this is the term used widely in the statistical learning community for the problem of approximating real-valued functions.
- Given a new query instance xq, the general approach in locally weighted regression is to construct an approximation f that fits the training examples in the neighborhood surrounding xq.
- This approximation is then used to calculate the value f(xq), which is output as the estimated target value for the query instance.

Locally Weighted Linear Regression

- Let us consider the case of locally weighted regression in which the target function f is approximated near x, using a linear function of the form

$$\hat{f}(x) = w_0 + w_1 a_1(x) + \ldots + w_n a_n(x)$$

Where $a_i(x)$ denotes the value of the ith attribute of the instance x.

- Minimize the squared error :

$$E_3(x_q) = \frac{1}{2} \sum_{x \in k \text{ nearest nbrs of } x_q} (f(x) - \hat{f}(x)^2 K(d(x_q, x))$$

- Kernel function K is the function of distance that is used to determine the weight of each training example

$$\Delta w_j = \eta \sum_{x \in k \text{ nearest nbrs of } x_q} K(d(x_q, x)) - (f(x) - (\hat{f}(x)) a_j(x)$$

3.15 : Radial Basis Functions

Q.35 What is radial basis function network ?

Ans. : • Radial Basis Function (RBF) network is an artificial neural network that uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters.

- RBF networks form a special class of neural networks, which consist of three layers.
- The input layer is used only to connect the network to its environment.
- The hidden layer contains a number of nodes, which apply a nonlinear transformation to the input variables, using a radial basis function, such as the Gaussian function, the thin plate spline function etc.
- The output layer is linear and serves as a summation unit.

Q.36 List the features of Radial Basis Function (RBF) networks.

Ans. : Features are as follows :

1. They are two-layer feed-forward networks.
2. The hidden nodes implement a set of radial basis functions (e.g. Gaussian functions).
3. The output nodes implement linear summation functions as in an MLP.
4. The network training is divided into two stages : first the weights from the input to hidden layer are determined, and then the weights from the hidden to output layer.
5. The training/learning is very fast.
6. The networks are very good at interpolation

Q.37 What is a radial basis function network ? Explain with architecture.

Ans. : • Radial Basis Function Networks (RBFN) are a variant of the three-layer feedforward neural networks. They contain a pass-through input layer, a hidden layer and an output layer.

- Fig. Q.37.1 shows a schematic diagram of an RBFN.

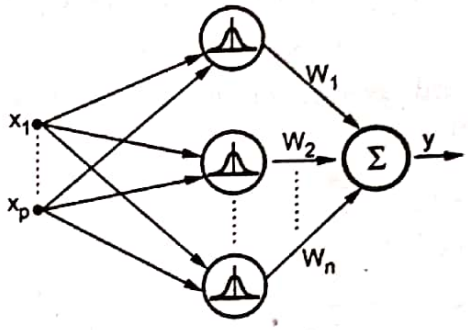

Fig. Q.37.1 Schematic diagram of RBFN

- The transfer function in the hidden layer is called a radial basis function (RBF). The RBF networks divide the input space into hyperspheres, and utilize a special kind of neuron transfer function.
- Radial basis functions are frequently used to create neural networks for regression-type problems. Their characteristic feature is that their response decreases (or increases) monotonically with distance from a central point.
- The hidden unit is given as :

$$W_i = R_i(X) = R_i(\|x - u_i / \sigma_i\), \quad i = 1, 2, \ldots H.$$

Where x = Multi-dimensional input vector

u_i = Vector with the same dimension as x

H = Number of radial basic function

$R_i(.)$ = It is i^{th} radial basis function with a single maximum at the origin.

- There are no connection weights between the input layer and the hidden layer.
- Typically $R_i(.)$ is a Gaussian function.

$$R_i(x) = \exp\left(-\frac{\|x-u_i\|^2}{2\sigma_i^2}\right)$$

Or a logistic function

$$R_i(x) = \frac{1}{1+\exp[\|x-u_i\|^2/\sigma_i^2]}$$

- The output of the RBFN can be computed in two ways :

1. Weight sum of the output value associated with each receptive field :

$$d(x) = \sum_{i=1}^{H} c_i w_i = \sum_{i=1}^{H} c_i R_i(X)$$

where c_i = output value associated with the ith receptive field.

2. Weighted average of the associated with each receptive field :

$$d(x) = \frac{\sum_{i=1}^{H} c_i w_i}{\sum_{i=1}^{H} w_i}$$

$$= \frac{\sum_{i=1}^{H} c_i R_i(x)}{\sum_{i=1}^{H} R_i(x)}$$

- Moody darken RBFN may be extended by assigning a linear function to the output function of each receptive field i.e. making c_i a linear combination of the input variables plus a constant :

$$c_i = a_i^T x + b_i$$

Where a_i is a parameter vector and b_i is a scalar parameter.

- A hidden neuron is more sensitive to data points near its center. For Gaussian RBF the sensitively may be turned by adjusting the spread σ, where a larger spread implies less sensitivity.
- An RBFN approximation capacity may be further improved with supervised adjustments of the center and shape of the receptive field function.

Q.38 Compare RBF network with multilayer perceptron.

Ans. :

No.	RBF networks	Multilayer perceptrons
1	An RBFN has a single hidden layer.	MLP may have one or more hidden layers.
2	Hidden layer is nonlinear and output layer is linear.	Hidden and output layer used as pattern classifier are usually nonlinear.
3	The argument of the activation function of each hidden unit computes the Eucliden norm between the input vector and the centre of that unit.	The activation function of each hidden unit computes the inner product of the input vector and the synaptic weight vector of that unit.
4	RBF networks using exponentially decaying localized nonlinearities construct local approximations to nonlinear input-output mappings.	MLPs construct global approximations to nonlinear input-output mapping.
5	Computation nodes in the hidden layer of an RBF network are quite different and serve a different purpose from those in the output layer of the network.	Computation nodes of an MLP, located in a hidden or an output layer, share a common neuronal model.

3.16 : Case-based Reasoning

Q.39 What is case-based reasoning ? Explain its steps.

Ans. : • Case based reasoning stores information from previous experiences. Using previously gained knowledge to solve current problems. It is similar to human problem solving methods.

- CBR has been applied to problems such as conceptual design of mechanical devices based on a stored library of previous designs.
- Basic Steps :
  1. Identify the problem/case.
  2. Look for a similar, previously experienced case.
  3. Predict a solution, possibly different from past experiences.
  4. Evaluate the solution.
  5. Update the system with the results.

- Case-based reasoning can be used for classification and regression. It is also applicable when the cases are complicated, such as in legal cases, where the cases are complex legal rulings, and in planning, where the cases are previous solutions to complex problems.
- A common example of a case-based reasoning system is a helpdesk that users call with problems to be solved.
- For example, case-based reasoning could be used by the diagnostic assistant to help users diagnose problems on their computer systems.
- When a user gives a description of their problem, the closest cases in the case base are retrieved.
- The diagnostic assistant can recommend some of these to the user, adapting each case to the user's particular situation.
- An example of adaptation is to change the recommendation based on what software the user has, what method they use to connect to the Internet, and the brand of printer.
- If one of the cases suggested works, that can be recorded in the case base to make that case be more important when another user asks a similar question.
- If none of the cases found works, some other problem solving can be done to solve the problem, perhaps by adapting other cases or having a human help diagnose the problem.
- When the problem is finally fixed, what worked in that case can be added to the case base.

3.17 : Remarks on Lazy and Eager Techniques

Q.40 Discuss concept of weak and eager learner.

Ans. : • Eager learning is a learning method in which the system tries to construct a general, input-independent target function during training of the system, as opposed to lazy learning, where generalization beyond the training data is delayed until a query is made to the system.

- Combining several weak learners to give a strong learner. It is a kind of multiclassifier systems and meta-learners. Ensemble typically applied to a single type of weak learner.
- **Lazy learning** (e.g., instance-based learning) : Simply stores training data (or only minor processing) and waits until it is given a test tuple.
- **Eager learning** (the above discussed methods) : Given a set of training tuples, constructs a classification model before receiving new (e.g., test) data to classify.
- **Lazy** : less time in training but more time in predicting.
- Lazy method effectively uses a richer hypothesis space since it uses many local linear functions to form an implicit global approximation to the target function.
- **Eager** : must commit to a single hypothesis that covers the entire instance space.

Fill in the Blanks for Mid Term Exam

Q.1 _____ networks are a type of artificial neural network constructed from spatially localized kernel functions.

Q.2 A _____ estimate of a parameter consists of an interval of numbers along with a probability that the interval contains the unknown parameter.

Q.3 Bayes theorem provides a way to calculate the probability of a hypothesis based on its _____, the probabilities of observing various data given the hypothesis, and the observed data itself.

Q.4 The Minimum Description Length principle recommends choosing the _____ that minimizes the sum of e two description lengths.

Q.5 Given a new instance to classify, the _____ algorithm simply applies a hypothesis drawn at random according to the current posterior probability distribution.

Q.6 The EM algorithm has been used to train Bayesian belief networks as well as _____.

Q.7 PAC-learnability is largely determined by the number of training examples required by the _____.