MOD-IV

(1) Hidden Markov Models (HMMs) for Sequential Classification :-

- Discrete Hidden Markov Models (DHMM)
- Continuous Density Hidden Markov Models (CDHMM)

(2) Hidden Markov Models (HMMs)

- Statistical model used to describe sequential data (like speech, handwriting, DNA, etc.) where the system is modeled as a Markov process with hidden (unobservable) states.

- Real world Ex :-
  - → Speech recognition.
  - → Bioinformatics (DNA sequences)
  - → Handwriting recognition.

- Components of HMM :

Symbol

N → No. of hidden states.

M → No. of possible observ^n. (for discrete models)

$A = \{a_{ij}\}$ → State transition probability matrix (State i to j)

$B = \{b_j(k)\}$ → observ^n. probability matrix.

$\pi = \{\pi_i\}$ → initial state distric^n.

Together, HMM is represented as :-

$\lambda = (A, B, \pi)$

3 main problems solved by this :-

1) Evaluation problem → Forward Algo.
2) Decoding Problem → Viterbi Algo.
3) Learning Problem → Baum-welch Algo.

(3) Discrete Hidden Markov Models (DHMM)

→ Discrete observn comes from a finite set (like words, phonemes, symbols).

→ Each state emits one of M discrete obs based on a probability distribution.

Ex:- Let's say M = 3 (Symbols: A, B, C).

If you're in State S_1 :-

Probability of A = 0.6
B = 0.3
C = 0.1

Then you emit one of them randomly based on this distribution.

→ Modelling steps :-

1. Define States & obs. symbols.
2. Initialize A, B and π.
3. use forward-Backward to compute req. probabilities.
4. Train using EM if model is unknown.

→ Applications → Text seq. classifn.
→ Speech recognition.
→ POS tagging.

(4) Continuous Density Hidden Markov Models (CDHMM)

* Observn. are continuous-valued vectors
* Each State emits observn. acc. to a continuous prob. distrib., usually a Gaussian / mixture of Gaussians (GMM)

Ex:- If you're in State S, your obs. is a feature vector like :-

$$x = [2.3, -1.5, 0.7]$$

This is drawn from:

$$P(x|S_1) \sim N(\mu_1, \Sigma_1)$$

Or more generally:

$$P(x|S_j) = \sum_{m=1}^{M} c_{jm} \cdot N(x|\mu_{jm}, \Sigma_{jm})$$

- c_{jm} → mixture weights
  μ_{jm}, Σ_{jm} :- mean & covariance of m-th Gaussian in state.

Real-world data like speech, images or sensor signals are not discrete — they are continuous.
CDHMMs provides a more realistic & powerful modelling tool for such data.

MOD-V

① Dimension Reduction Methods

Process of reducing the no. of I/P variables in a dataset while preserving as much information as possible.

Why ??
- removes irrevant features.
- reduces computational cost.
- helps in visualizing high-Dimension data.
- Improves classification.

2 main Approaches :-

Feature selection → choose a subset of original features

Feature Extraction → create new features from combination of old ones.

1) Fisher Discriminant Analysis
2) Principal component Analysis

② Fisher Discriminant Analysis

Goal :- find a new axis that maximizes class separability.

Key Idea :- Maximize b/w-class variance
Minimize within-class variance.

How it works :-
1. compute mean of each class & overall mean.
2. compute b/w-class scatter & within-class scatter
3. find projection w that maximizes the ratio :- $J(w) = \frac{w^T S_B w}{w^T S_W w}$

S_B → b/w-class scatter matrix
S_W → within-class scatter matrix

Pros

- Supervised
- Great for classification task
- Better class separation than PCA.

Cons

- Assumes normally distributed features
- Works best when class covariances are similar.

③ Principal component Analysis

Goal: Transform original data into a new coordinate system such that:-

- 1st new axis captures max. variance.
- Next axes capture remaining variance in decreasing order.

How it works?

1. Standardize data (mean = 0)
2. Compute covariance matrix.
3. Compute eigenvectors & eigenvalues
4. Select top k eigenvectors → new feature space.
5. Transform data into new space

Pros:

- Reduces overfitting
- Captures max. variance
- useful for visualizing data in 2D/3D.

Cons

- Principal comp. are not easily interpretable.
- Assumes linear relationship.

PCA vs LDA

• Unsupervised	• Supervised
• Max. variance	• Maximize class separation
• No class labels needed	• Needs class labels.
• Principal component	• Discriminant vectors.

Parzen-window methods

- Non-parametric technique
- to estimate the probability density function (PDF) of a random variable.

It helps us guess the shape of the data distribution, without assuming any specific shape.

Used in -
- pattern recognition.
- classification
- density estimation.

Imagine you are trying to find how people are spread across a park.
You don't know the exact distribⁿ.
So you place a small window around every person & count how many people fall within that window.
If many people fall into a window, you say "density is high here".
If few people fall in, you say "density is low here".

This is Parzen-window idea!

Adv.
- Doesn't assume any fixed shape (flexible)
- works for any distribution.

Disadv.
- Can be slow for large datasets.
- Choice of window size (h) is critical.
  - Too small → noisy
  - Too big → oversmooths.

(5) K-Nearest Neighbours method

→ K-NN – Supervised learning algo:
used for:- classification
- regression

→ Simplest algo in ML.

→ Ex:- Imagine you have a dataset of fruits.

color	Size	fruit
Red	Big	Apple
Yellow	Big	Banana
Red	Small	Cherry

Now you get a new fruit:-
Color: Red.
Size: Medium

using K-NN:
- You find its k-closest fruits in terms of color & size.
- Suppose k=3, and 2 of the 3 closest fruits are Apples → the new fruit is classified as Apple.

Dist. measures

1) Euclidean dist.:

$$d(p,q) = \sqrt{(p_1-q_1)^2 + (p_2-q_2)^2 + \cdots + (p_n-q_n)^2}$$

2) Manhattan Distance

3) Minkowski Distance.

Choosing K:
- Small K (like 1 or 3)
  Sensitive to noise & outliers.
- Large K (like 10 or 15):
  Smoother decision boundaries but may ignore local patterns.

MOD-VI

Non parametric techniques for density estimation:-

- Density Estimation → process of estimating the probability distribution (PDF) of a random variable based on given data.

Parametric	Non-parametric
- Assumes a fixed form - estimates parameters (mean, variance)	Makes no assumption about the distribution shape

Why Non-parametric → more flexible.
→ works better for unknown distribution.

Common Non-parametric methods:-

1. Histogram Estimation:-
   - divide the data range into eq-sized bins
   - count how many points fall into each bin
   - Estimate density by:-

     $$\text{Density} = \frac{\text{count in bin}}{n \times \text{Bin width}}$$

2. Parzen window method } module-(5)
3. k-NN