MOD-I Pattern Recognition.

Basics of Pattern Recognition.

What is Pattern Recognition?
- Science of identifying patterns & regularities in data
- Involves classifying I/P data into identifiable categories based on key features/ patterns.

Pattern →
Set of attributes/features - help in identifying/categorizing data
Ex: A handwritten digit like "5" is a pattern.
The strokes & curves are features.

Recognition →
labeling / classifying a given pattern into a known class.
Ex - Recognizing whether a shape is circle/pattern

classifier →
An algorithm/model that assigns labels to I/P pattern
Ex - Decision Tree, K-NN, SVM, Neural Networks

Types of Pattern Recognition →

Supervised learning → Training with labeled data. The system learns from ex.

unsupervised learning → No labels provided. The system groups data based on similarity.

Reinforcement learning → learning from feedback (reward / punishment) in an environment.

Steps in PR process:

1. Data collection → Gathering I/P data
2. Preprocessing → cleaning & transforming data (eg. noise removal, normalizn.)
3. Feature Extraction → Selecting imp. info. from raw data.
4. classification → Using algo. to assign categories to the data
5. Post-processing → Improving/interpreting the output (eg. majority voting)

Application of PR →

Handwriting recognition (OCR)
Face Recognition
Speech Recognition
Medical diagnosis
Fraud detection

Common Algorithms used →

K-Nearest Neighbor (K-NN)
Support Vector Mic (SVM)
Naive Bayes
Decision Trees
Artificial Neural Networks
Clustering (eg. K-Means - unsupervised learning)

Features selection vs. feature extraction

- selects subset of original features.
- ~~Reduces dimensionality~~ Remove irrelevant features.
- easy to interpret
- Techniques :- Filter (chi-sq.)
- less intensive (computn cost)
Ex:- Selecting top 10 features based on correlation scores.

- creates new features from original data.
- ~~create~~ transforming original features into a new set of features.
- Harder to interpret.
- Techniques - Autoencoders
- more intensive.
Ex:- Reducing 100 features to 10 using PCA.

MOD-II

① Bayesian Decision Theory

- fundamental statistical approach to the problem of pattern classification.
- uses probability & Bayes' Theorem
- to minimize risk (error) of making wrong decisions.
- Key components :-

Classes ($\omega_1, \omega_2, \ldots \omega_n$): The categories / labels.

Prior Probability $P(\omega)$: The prob. that a random sample belongs to class ω before seeing the data.

- Likelihood $P(x|\omega)$: The probability of observing a feature vector x given class ω.
- Posterior Probability $P(\omega|x)$: Prob. that the pattern belongs to class ω given feature x (calculated using Bayes theorem).

Bayes' Theorem

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

where :-

$$P(x) = \sum_j P(x|\omega_j) \cdot P(\omega_j)$$

Bayesian Decision Rule (Min. Error)

- choose the class ω_i for which $P(\omega_i|x)$ is max

or, - choose ω_i if $P(\omega_i|x) > P(\omega_j|x)$, $\forall j \neq i$

② Classifiers :-

- used to assign i/p patterns (like images, text) to predefined categories / classes.
- most classifiers operate in supervised learning setting where the model is trained using labeled data.
- include K-NN, SVM, Decision Trees, Naive Bayes, Neural Networks.

- classifier creates decision boundary
diff. classes in the feature space.
- shape of this boundary depends on type of classifier.
- classifier effectiveness - measured using tools like confusion matrix, f1-score, recall, etc.

Discriminant Functions

- score fun. used to make decisions.
- is a formula / rule helps us decide which class a data point belongs to.
- helps in classifying objects / patterns by finding which class has highest score / value for a given I/P.
- ex:- If you're classifying emails as spam or not spam, a discriminant fun. will give a score for each class. whichever score is higher determines the final label.

- Decision Rule :-
Assign x to class w_i if $g_i(x) > g_j(x)$, $\forall j \neq i$

For bayesian classifier :-
$g_i(x) = P(w_i|x)$ or $\ln P(x|w_i) + \ln P(w_i)$

Decision Surface :

- boundary separates regions in the feature space where diff. decision rules apply.

Ex:- for 2D feature space, it could be a line (linear classifier) or curve (non-linear)

- created by discriminant functions.
- can be diff. shapes :- Line (2D), Plane (3D) or complex curve depending on classifier (linear / non-linear)
- Ex:- If you're classifying fruits as apples or Oranges based on color & size, the decision surface would be a line / curve that separates apples from oranges in a color-size graph.

(3) Normal Density & Discriminant functions

Normal Density

→ Gaussian Distribution.
→ Bell-shaped curve
→ Shows how data is spread out - Most values are around avg. (mean) & few are far away.
→ It is used when we assume that the data for each class follows a natural pattern (ex: cats vs dogs).
→ N.D. formula - helps us calculate probability that a data point belongs to a class based on its features.

$$P(x|\omega_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\right]$$

where:-
μ_i : mean vector
Σ_i : covariance matrix
d : dimensionality of x.

→ ex:- Height of stu. in class follows normal curve. most are near avg., few are very tall/short.

Discriminant function for Gaussian

If covariance matrices are equal for all classes the discriminant becomes linear:-

$$g_i(x) = w_i^T x + w_{i0}$$

where:-
$w_i = \Sigma^{-1}\mu_i$
$w_{i0} = -\frac{1}{2}\mu_i^T \Sigma^{-1}\mu_i + \ln P(\omega_i)$

If covariance are different, it becomes a quadratic function of x.

(4) Discrete features in Detail:

When features are discrete (categorical) rather than continuous:-

✓ use Probability Tables:-

→ Each feature has a finite no. of values.

→ compute conditional probabilities directly.

$$P(x|\omega_i) = P(x_1, x_2, \ldots, x_n|\omega_i)$$

If assuming independence b/w features (Naive Bayes)

$$P(x|\omega_i) = \prod_{j=1}^{n} P(x_j|\omega_i)$$

Example

For spam detection using words (discrete feature)

x_1 : "free" appears / not

x_2 : "offer" appears / not

Learn probabilities like $P(x_1 = 1|\omega_{spam})$

use Naive Bayes classifier:-

$$g_i(x) = \ln P(\omega_i) + \sum_j \ln P(x_j|\omega_i)$$

MOD-III

(A) Parameter Estimation Methods

a) Maximum Likelihood Estimation (MLE)
b) Gaussian Mixture Models (GMM)
c) Expectation-Maximization (EM) Algorithm
d) Bayesian Estimation.

Parameter Estimation:

- process of estimating parameters (like mean, variance, etc.) of a probability distribution from observed data.
For ex:- Estimating mean (μ) and variance (σ^2) of a normal distribution.

Types of Estimation.

1. Point estimation → gives a single value estimate (eg. Sample mean).
2. Interval estimation → gives a range of values (confidence interval).

Common estimators

- Sample mean ($\bar{x}$) estimates pop^n mean (μ)
- Sample variance (s^2) estimates pop^n variance (σ^2)
- Sample proportion ($\hat{p}$) estimates pop^n proportion (p)

→ It helps us make decisions & predictions when we don't know the true values for the entire population.

a) Maximum Likelihood Estimation

- to estimate unknown parameters of a probability distribution by maximizing the likelihood that the observed data came from that distribution.

→ Likelihood - probability of data given certain parameters.

In MLE, we try to maximize this likelihood.

→ Steps → write the likelihood function.
→ Take the log (called log-likelihood)
→ Differentiate the log-likelihood
→ Solve the eq^n by setting derivative = 0
→ gives MLE estimate.

Common ex:- coin Toss Ex. (Bernoulli Trial)
- Suppose we toss a coin 10 times, & get 7 heads and 3 tails.
- Let p = probability of head
- Likelihood : $L(p) = p^7 \cdot (1-p)^3$
- Take log, differentiate, and solve → get MLE for $p = 7/10$.

MLE → widely used in:
ML, statistics, Data science, PR.

Properties of MLE:-
consistent, efficient,

Limitations:- can be difficult for complex distribution

Real-Life Analogy:-
Imagine guessing someone's weight by trying diff. values & seeing which guess best explains how their clothes fit = MLE is similar.

(b) **Gaussian Mixture Models** (GMM)

- way to group or cluster data that assumes data comes from a mixture of several normal (Gaussian) distributions.
- Each grp. (or cluster) is represented by bell-shaped curve (Gaussian distribution).
- Gaussian distribution is bell-shaped curve often seen in statistics.
  Defined by Mean (μ): center of curve
  Standard Deviation (σ): width of the curve.
- Why mixture? bcz. in real life, data is often a mix of diff. grps. GMM says:-
  Let's assume data comes from several Gaussian distributions combined together.
- Ex. - Imagine you're looking at the heights of people in a room that has:
  Adults, Teenagers, Kids.
  Each grp. will have a diff. avg. height
  So, GMM would try to separate the data into 3 groups each with its mean & std. deviation.

Application → Image Segmentation
Speech Recognition.

(c) **Expectation - Maximization method** (EM)

- to find missing / hidden info. in data by repeating 2 steps :- • Guessing
  • Improving the guess.
- It is mainly used in problems where some data is hidden / incomplete, like in GMM.

- Ice-cream Analogy.

Imagine 2 ice-cream m/c are making cones. You see only the cones but you don't know which m/c made which cone. You want to guess!

- The avg. size of cones made by each m/c.
- Which m/c likely made which cone.

But you don't know which cone came from which m/c - that's the hidden data.

EM helps you figure out that.

- 2 main steps in EM.

i) Expectation Step (E-Step) - "Guess who did what"
→ calculate probability.

ii) Maximization Step (M-Step) - "Update your guess"
→ recalculate best possible parameters.

- Repeat doing E-Step & M-Step again & again until the results stop changing much.

- Applications → GMM
→ Clustering
→ Image Processing

(d) Bayesian Estimation

→ Method of estimating unknown values (called parameters) using :- what we already know (prior knowledge)
New data we observe (evidence)
It uses Baye's Theorem to update our beliefs.

→ Weather Forecast (ex)

- You believe there's 70% chance it will rain tomorrow (prior belief)
- You now check the weather app, & it shows dark clouds (new evidence).

Using Bayesian estim", you update your belief based on this new evidence.

Now, you might say :- now I believe there's 90% chance it will rain.

→ Formula (Baye's Th.)

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}.$$

Prior → what you believe before seeing the data.
Likelihood → how likely is the new data, assuming your belief is true.
Posterior → updated belief after seeing the new data.
Evidence → A normalizing factor

→ Application :- Spam filters,
ML, Robotics & AI.