:- Module 2 :-

Introduction :-

→ The goal in classification is to take an input vector x & to assign it to one of K discrete classes C_k.

→ The input space is divided into decision regions whose boundaries are called as decision boundaries.

→ As we are discussing linear classification, it means that the decision surfaces are linear functions of input vector x & are defined by (D-1) dimensional hyperplanes within the D-dimensional input space.

→ There are 3 distinct approches to the classification problems & the simplest involves constructing a discriminant function which directly assigns each vector 'x' to a specific class.

Discriminant functions :-

→ A discriminant function takes an i/p vector 'x' and assigns it to one of K classes denoted as C_k.

→ Here we anume dinear discriminations in which decision surfaces are hyperplanes.

Two classes :-

Simplest Representation of a dinear discriminant function is

$$y(x) = w^T x + w_0$$

$w \rightarrow$ weight vector

$w_0 \rightarrow$ Bias

→ ϑ/p vector x is assigned to class C_1 if $y(x) \geq 0$ & C_2 if otherwise.

→ The decision boundary is corresponding to a $(D-1)$ dimensional hyperplane within the D-dimensional Onput space.

→ Consider two points x_A & x_B both lie on the decision surfaces. Because $y(x_A) = y(x_B) = 0$ we have $w^T(x_A - x_B) = 0$ & hence 'w' vector is orthogonal to every vector lying on the decision surface, ∴ 'w' determines the Orienta-tion of the decision surface.

→ The normal distance from the Origin to the decision surface is given by

$$\frac{w^T x}{\|w\|} = -\frac{w_0}{\|w\|}$$

→ Consider the diagram shown below.

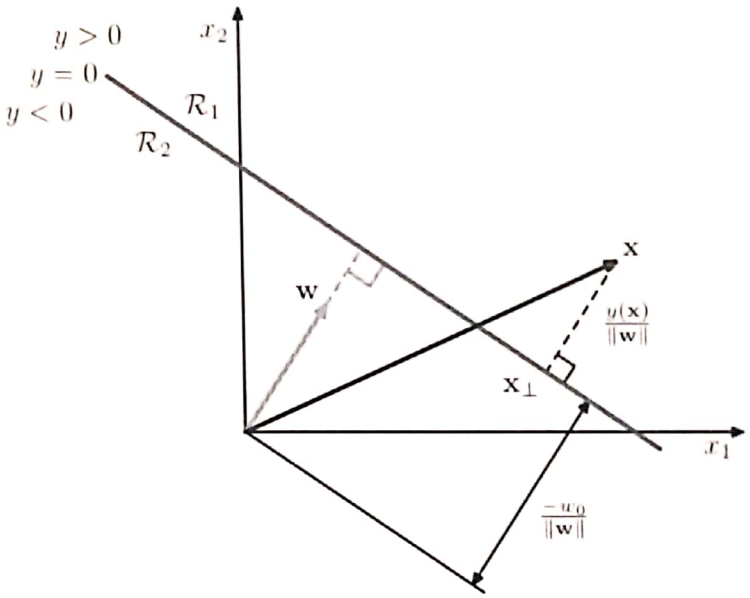


Figure 4.1 Illustration of the geometry of a linear discriminant function in two dimensions. The decision surface, shown in red, is perpendicular to $\mathbf{w}$, and its displacement from the origin is controlled by the bias parameter w_0. Also, the signed orthogonal distance of a general point $\mathbf{x}$ from the decision surface is given by $y(\mathbf{x})/\|\mathbf{w}\|$.

→ Here $y(x)$ gives a signed measure of the perpendicular distance r of the point x from the decision surface.

→ For Example let x be the arbitary point & $x_\perp$ be its orthogonal projection onto the decision surface.

$$x = x_\perp + r\frac{w}{\|w\|}$$

→ multiplying both sides with w^T & adding w_0 we get-

$$x \cdot w^T + w_0 = \left[x_\perp + r \frac{w}{\|w\|} \right] w^T + w_0$$

* using $y(x) = w^T x + w_0$

$$y(x_\perp) = w^T x_\perp + w_0 = 0$$

we get

$$r = \frac{y(x)}{\|w\|}$$

$$y(x) = \tilde{w}^T \tilde{x}$$

ii) Multiple classes :-

→ Consider Extension of linear discriminations to $K > 2$ classes.

→ Consider the use of $K-1$ classifiers, each of which solves a two class problem of seperating points in a particular class C_k from points not in that class.

→ This is known as a one-versus-the-rest classifier.

→ Fig below shows an example of 3 classes with ambiguous classification.

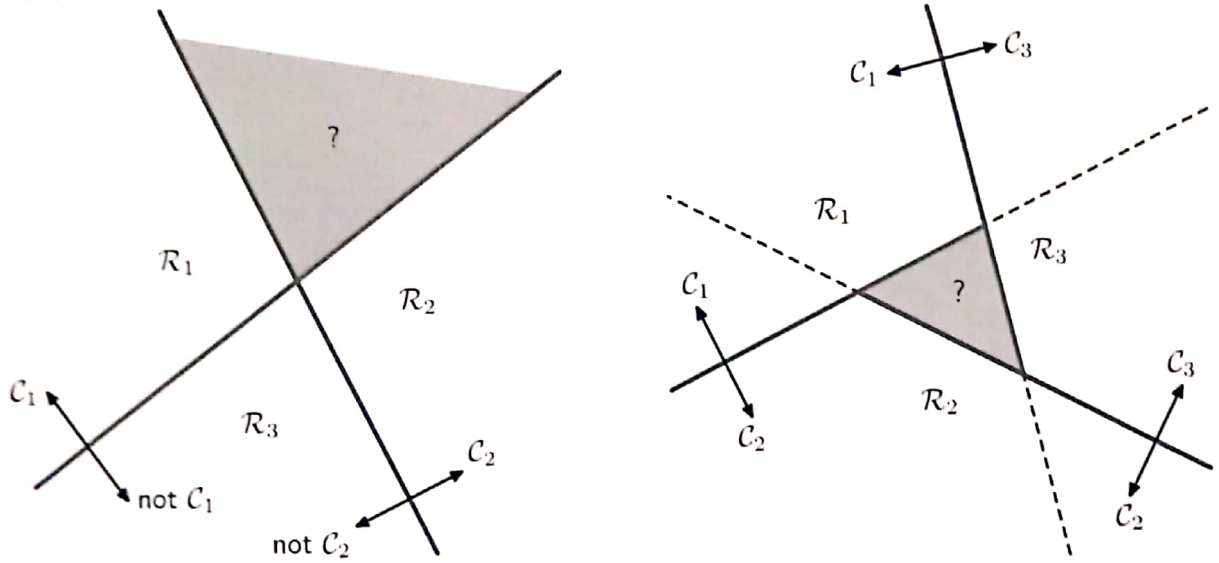


Figure 4.2 Attempting to construct a K class discriminant from a set of two class discriminants leads to ambiguous regions, shown in green. On the left is an example involving the use of two discriminants designed to distinguish points in class $\mathcal{C}_k$ from points not in class $\mathcal{C}_k$. On the right is an example involving three discriminant functions each of which is used to separate a pair of classes $\mathcal{C}_k$ and $\mathcal{C}_j$.

→ An alternate is to introduce $\frac{K(K-1)}{2}$ Binary discriminant functions, one for every possible pair of classes. This is known as a one-versus-one classifies.

→ further Illustrating in Equations.

$$y_k(x) = w_k^T x + w_{k0}$$

$$\& \ (w_k - w_j)^T x + (w_{k0} - w_{j0}) = 0$$

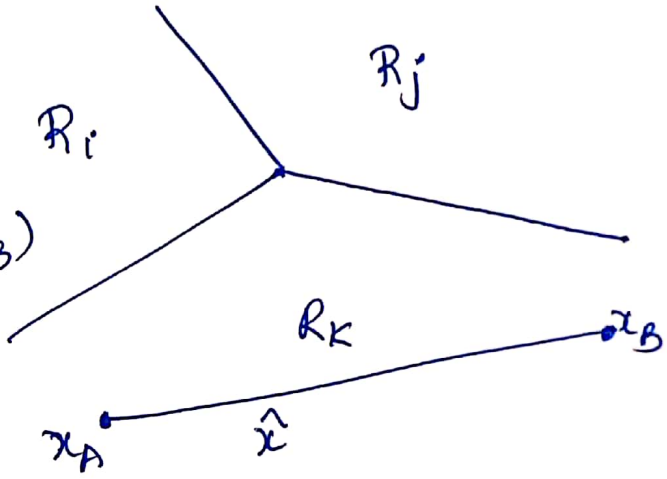
were K & j are two classes C_K & C_j.

→ Consider two points x_A & x_B both of which lie inside decision region R_K. Any point $\hat{x}$ that lies on the line connecting x_A & x_B can be expressed in the form.

$$\hat{x} = \lambda x_A + (1-\lambda) x_B.$$

where $0 \leq \lambda \leq 1$

$$y_K(\hat{x}) = \lambda y_K(x_A) + (1-\lambda) y_K(x_B)$$



note :- Both x_A & x_B lie inside R_K. R_K is singly connected & convex.

Least Squares for classification :-

→ We know that minimization of sum-of-squares error function led to a simple closed-form solution for the parameter values. in regression, so that can be applied for classification.

→ Each class C_K described by its linear model

$$y_K(x) = w_K^T x + w_{K0}$$

where $K = 1, \cdots, K$, grouping these together using vector notation.

$$y(x) = \tilde{w}^T \tilde{x}$$

→ we now determine parameter matrix $\tilde{w}$ by minimizing sum-of-squares error function.

Error function is defined as

$$E_D(\tilde{w}) = \frac{1}{2} Tr\{(\tilde{x}\tilde{w} - T)^T (\tilde{x}\tilde{w} - T)\}$$

* Setting the derivative with respect to $\tilde{w}$ to zero & rearranging we obtain

$$\tilde{w} = (\tilde{x}^T \tilde{x})^{-1} \tilde{x}^T T = \tilde{x}^{\dagger} T$$

↓

Pseudo-inverse of the matrix $\tilde{x}$.

note :- Sum-of-squares error function penalizes predictions that are 'too correct' & which lie along the decision boundary.

Principle Component Analysis :-

→ It is a way of identifying pattern in data and Expressing the data in such a way to highlight their similarities & differences

→ It is also way to reduce the dimension [Dimensionality Reduction] to reduce the complexity of the Analysis.

Example problem :-

1. Given the following data use PCA to reduce the dimension from 2 to 1.

Feature	1	2	3	4
x	4	8	13	7
y	11	4	5	14

Step 1 :- from the data set find n & N

n = no. of features = 2

N = no. of samples = 4

Step 2 :- Computation of mean of variables

$$\bar{x} = \frac{4+8+13+7}{4} = 8$$

$$\bar{y} = \frac{11+4+5+14}{4} = 8.5$$

Step 3 :- Computation of covariance matrix

i) find ordered pairs → x, y

(x, x) (x, y) (y, x) (y, y)

ii) find covariance of all ordered pairs

$$Cov(x,x) = \frac{1}{N-1} \sum_{K=1}^{N} (x_{iK} - \bar{x}_i)(x_{jK} - \bar{x}_j)$$

or

a) $$Cov(x,x) = \frac{1}{N-1} \sum_{K=1}^{N} (x_i - \bar{x})^2$$

$$= \frac{1}{4-1} \left[(4-8)^2 + (8-8)^2 + (13-8)^2 + (7-8)^2 \right]$$

$$= 14$$

b) $$Cov(x,y) = \frac{1}{4-1} \left[\begin{array}{l} (4-8)(11-8.5) + (8-8)(4-8.5) \\ + (18-8)(5-8.5) + (7-8)(14-8.5) \end{array} \right]$$

$$= -11$$

$$cov(y,x) = cov(x,y) = -11$$

$$Cov(y,y) = \frac{1}{4-1}\left[(11-8.5)^2+(4-8.5)^2+(5-8.5)^2 + (14-8.5)^2\right]$$

$$= 23$$

Covariance matrix :-

$$S = \begin{bmatrix} cov(x,x) & cov(x,y) \\ cov(y,x) & cov(y,y) \end{bmatrix}$$

$$= \begin{bmatrix} 14 & -11 \\ -11 & 23 \end{bmatrix}$$

Step 4 :- Eigen value, Eigen vector, normalized eigen vector.

i) Eigen value :-

$$\det(S - \lambda I) = 0$$

$$\text{Identity matrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \lambda = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}$$

$$S = \begin{bmatrix} 14 & -11 \\ -11 & 23 \end{bmatrix}$$

$\therefore \det(S - \lambda I) = 0$

$$\det\left[\begin{pmatrix} 14-\lambda & -11 \\ -11 & 23-\lambda \end{pmatrix}\right] = 0$$

$$(14-\lambda)(23-\lambda) - (-11 \times -11) = 0$$

$$\lambda^2 - 37\lambda + 201 = 0$$

$$\therefore \lambda_1 = 30.3849$$

$$\lambda_2 = 6.6151 \qquad [\lambda_1 > \lambda_2]$$

* pick largest value which becomes first principal component.

ii) Eigen vector of λ_1

$$(S - \lambda_1 I)\, U_1 = 0$$

↳ Eigen vector of λ_1

$$\begin{bmatrix} 14-\lambda_1 & -11 \\ -11 & 23-\lambda_1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} (14-\lambda_1)u_1 - 11u_2 \\ -11u_1 + (23-\lambda_1)u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$(14-\lambda_1)u_1 - 11u_2 = 0 \longrightarrow (1)$$

$$-11u_1 + (23-\lambda_1)u_2 = 0 \longrightarrow (2)$$

* use any one linear Equation to find values of eigen vector

$\therefore$ Using Eqn (1)

$$(14-\lambda_1)u_1 - 11u_2 = 0$$

$$\frac{u_1}{11} = \frac{u_2}{14-\lambda_1} = t$$

when $t = 1$

$$u_1 = 11, \quad u_2 = 14-\lambda_1$$

Eigen vector u_1 of $\lambda_1 = \begin{bmatrix} 11 \\ 14-\lambda_1 \end{bmatrix}$

$$= \begin{bmatrix} 11 \\ 14-30.3849 \end{bmatrix} = \begin{bmatrix} 11 \\ -16.3849 \end{bmatrix}$$

* Normalizing Eigen vector u_1

$$e_1 = \begin{bmatrix} \frac{11}{\sqrt{11^2+(-16.838)^2}} \\ \frac{-16.3849}{\sqrt{11^2+(-16.38)^2}} \end{bmatrix} = \begin{bmatrix} 0.5574 \\ -0.8303 \end{bmatrix}$$

III^{rly} $e_2 = \begin{bmatrix} 0.8303 \\ 0.5574 \end{bmatrix}$

Step 5 :- Derive new dataset-

	1	2	3	4
first princi component	P_{11}	P_{12}	P_{13}	P_{14}

$$P_{11} = e_1^T \begin{bmatrix} x_i - \bar{x} \\ y_i - \bar{y} \end{bmatrix} = e_1^T \begin{bmatrix} 4 - 8 \\ 11 - 8.5 \end{bmatrix}$$

$$= \begin{bmatrix} 0.5574 & -0.8303 \end{bmatrix} \begin{bmatrix} -4 \\ 2.5 \end{bmatrix} = -4.3052$$

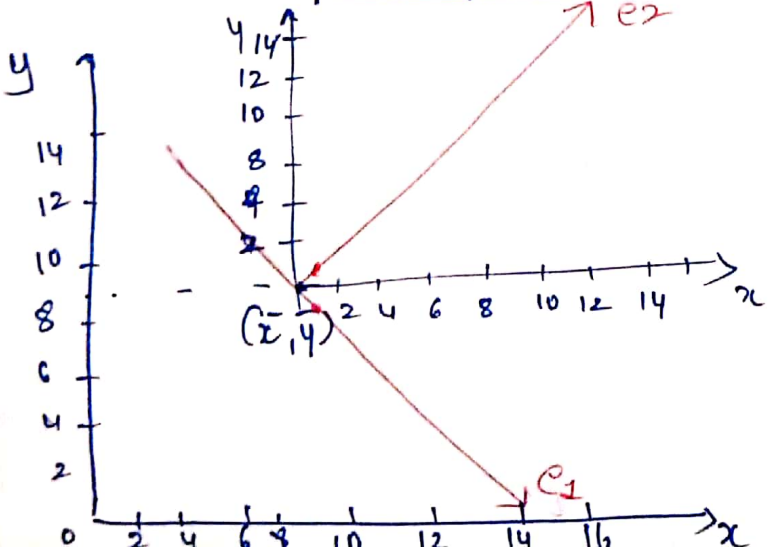
$$P_{12} = \begin{bmatrix} 0.5574 & -0.8303 \end{bmatrix} \begin{bmatrix} 8 - 8 \\ 4 - 8.5 \end{bmatrix} = 3.73$$

$P_{13} = 5.69$

$P_{14} = -5.12$

	1	2	3	4
PC1	-4.3	3.7	5.6	-5.12

$\Rightarrow$ new data set with dimension 1



Linear Discriminant Analysis :-

→ LDA is a dimensionality reduction technique used as a pre-processing step for pattern-classification & Machine learning applications.

→ LDA is similar to PCA but LDA in addition finds the axes that maximizes the seperation between multiple classes.

Goal :- To project a feature space (N-dimensional data onto a smaller subspace k (k ≤ n-1) while maintaining the class discriminating information

Example :- Lets take a 2-D dataset-

$C_1 \Rightarrow x_1 = (x_1, x_2) = \{(4,1), (2,4), (2,3), (3,6), (4,4)\}$

$C_2 \Rightarrow x_2 = (x_1, x_2) = \{(9,10)(6,8)(9,5)(8,7)(10,8)\}$

Step 1 :- Compute within-class scatter matrix (S_W)

$$S_W = S_1 + S_2$$

$S_1 \Rightarrow$ Covariance matrix for class C_1

$S_2 \Rightarrow$ Covariance matrix for class C_2.

i) finding S_1

$$S_1 = \sum_{x \in C_1} (x - \mu_1)(x - \mu_1)^T$$

$\mu_1 \rightarrow$ mean of class C_1

$$\mu_1 = \left\{ \frac{4+2+2+3+4}{5} , \frac{1+4+3+6+4}{5} \right\}$$

$$\mu_1 = [3.00 \quad 3.60]$$

III^{rly} $\mu_2 = [8.4 \quad 7.60]$

$$\therefore (x - \mu_1) = \begin{bmatrix} 1 & -1 & -1 & 0 & 1 \\ -2.6 & 0.4 & -0.6 & 2.4 & 0.4 \end{bmatrix}$$

Now, For Each x we calculate $(x - \mu_1)(x - \mu_1)^T$.

$$\therefore \begin{bmatrix} 1 \\ -2.6 \end{bmatrix} \begin{bmatrix} 1 & -2.6 \end{bmatrix} = \begin{bmatrix} 1 & -2.6 \\ -2.6 & 6.76 \end{bmatrix} \rightarrow (1)$$

$$\begin{bmatrix} -1 \\ 0.4 \end{bmatrix} \begin{bmatrix} -1 & 0.4 \end{bmatrix} = \begin{bmatrix} 1 & -0.4 \\ -0.4 & 0.16 \end{bmatrix} \rightarrow (2)$$

$$\begin{bmatrix} -1 \\ -0.6 \end{bmatrix} \begin{bmatrix} -1 & -0.6 \end{bmatrix} = \begin{bmatrix} 1 & 0.6 \\ 0.6 & 0.36 \end{bmatrix} \rightarrow (3)$$

$$\begin{bmatrix} 0 \\ 2.4 \end{bmatrix} \begin{bmatrix} 0 & 2.4 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 5.76 \end{bmatrix} \rightarrow (4)$$

$$\begin{bmatrix} 1 \\ 0.4 \end{bmatrix} \begin{bmatrix} 1 & 0.4 \end{bmatrix} = \begin{bmatrix} 1 & 0.4 \\ 0.4 & 0.16 \end{bmatrix} \rightarrow (5)$$

Adding (1) + (2) + (3) + (4) + (5) & taking average we get covariance matrix S_1

$$S_1 = \begin{bmatrix} \frac{4}{5} & -2/5 \\ -2/5 & (13.2/5) \end{bmatrix} = \begin{bmatrix} 0.8 & -0.4 \\ -0.4 & 2.6 \end{bmatrix}$$

Similary for the class 2, the co-variance matrix is given by

$$S_2 = \begin{bmatrix} 1.84 & -0.04 \\ -0.04 & 2.64 \end{bmatrix} \quad \& \quad \mu_2 = [8.4 \quad 7.6]$$

$$S_W = S_1 + S_2$$

$$S_W = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix}$$

Step 2:- Compute Between class scatter matrix (S_B)

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

$$\mu_1 = [3 \quad 3.6) \quad \mu_2 = [8.4 \quad 7.6]$$

$$S_B = \begin{bmatrix} -5.4 \\ -4 \end{bmatrix} \begin{bmatrix} -5.4 & -4 \end{bmatrix} = \begin{bmatrix} 29.16 & 21.6 \\ 21.6 & 16.00 \end{bmatrix}$$

Step 3:- Find the best LDA projection vector

* Similar to principal component Analysis we find this using eigen vectors using Largest Eigen value.

$$S_W^{-1} S_B \cdot W = \lambda \cdot W \longrightarrow (1)$$

(W → Projection vector)

i> Solving the eigen value

$$|S_W^{-1} S_B - \lambda I| = 0$$

$$S_W^{-1} = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix}^{-1} = \begin{bmatrix} 0.38 & 0.03 \\ 0.03 & 0.19 \end{bmatrix}$$

$$S_B = \begin{bmatrix} 29.16 & 21.6 \\ 21.6 & 16.00 \end{bmatrix}$$

$$\lambda I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \lambda = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}$$

$$S_W^{-1} \cdot S_B = \begin{bmatrix} 11.72 & 8.6 \\ 4.9 & 3.6 \end{bmatrix}$$

$$\therefore |S_W^{-1} S_B - \lambda I| = \begin{vmatrix} 11.72 - \lambda & 8.6 \\ 4.9 & 3.6 - \lambda \end{vmatrix} = 0$$

Solving for λ

$$(11.72-\lambda)(3.6-\lambda)-(4.9\times 8.6)=0$$

$$42.19-11.72\lambda-3.6\lambda+\lambda^2-42.14=0$$

$$\lambda^2-15.32\lambda+0.05=0.$$

$$\therefore \lambda_1=15.31 \quad , \lambda_2=3.26\times10^{-3}$$

Substituting highest Eigen value in Eqn (1)

$$\begin{bmatrix} 11.72 & 8.6 \\ 4.9 & 3.6 \end{bmatrix}\begin{bmatrix} w1 \\ w2 \end{bmatrix} = 15.31\begin{bmatrix} w1 \\ w2 \end{bmatrix}$$

$$11.72\,w1+8.6\,w2=15.31\,w1 \longrightarrow (2)$$

$$4.9w1+3.6w2=15.31\,w2 \longrightarrow (3)$$

$$11.72w1-15.31w1+8.6w2=0$$

$$-3.59w1+8.6w2=0 \longrightarrow (4)$$

$$4.9w1+3.6w2-15.3w2=0$$

$$4.9w1-11.7w2=0$$

$$\begin{bmatrix} -3.59 & 8.6 \\ 4.9 & -11.7 \end{bmatrix}\begin{bmatrix} w1 \\ w2 \end{bmatrix}=\begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$-3.59w1+8.6w2=0$$

$$w1=\frac{8.6}{3.59}w2=2.39w2$$

$$4.9w1-11.7w2=0$$

Let $w2=1$

$$\therefore w1=\frac{11.7}{4.9}=2.38$$

$$\begin{bmatrix} w1 \\ w2 \end{bmatrix}=\begin{bmatrix} 2.38 \\ 1 \end{bmatrix}$$

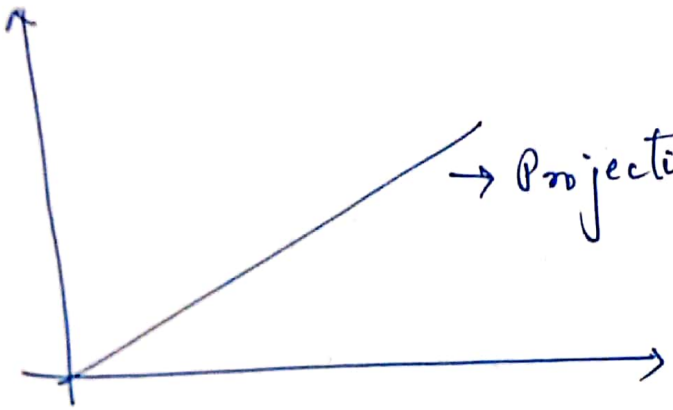
normalizing

$$\begin{bmatrix} w1 \\ w2 \end{bmatrix}=\begin{bmatrix} \frac{2.38}{\sqrt{1^2+2.38^2}} \\ \frac{1}{\sqrt{1^2+2.38^2}} \end{bmatrix}$$

$\therefore$ we get $\begin{bmatrix} w1 \\ w2 \end{bmatrix} = \begin{bmatrix} 0.91 \\ 0.39 \end{bmatrix}$

Step 4 :- Dimension Reduction

$y = w^T x$ → Input data sample.
↓
Projection vector



The perceptron algorithm :-

→ Here let us consider two-class model in which input vector x is first transformed using a fixed nonlinear transformation to give a feature vector $\phi(x)$ & then construct a generalized linear model

$$y(x) = f(w^T \phi(x)).$$

where the non-linear activation function $f(\cdot)$ is a step function of form

$$f(a) = \begin{cases} +1 & a \geq 0 \\ -1 & a < 0. \end{cases}$$

→ In Perceptron algorithm it convinent to use $t = +1$ for C_1 & $t = -1$ for C_2 as target values matching the choice of activation function.

→ To determine the parameter w correctly we can minimize error function but alternatively we can also use perceptron criterion.

→ To derive perceptron criterion based Error we 1st consider

pattern $x_n \rightarrow C_1 \Rightarrow$ when $w^T \phi(x_n) > 0$

$x_n \rightarrow C_2 \Rightarrow$ when $w^T \phi(x_n) < 0$

$t \in \{-1, +1\}$

note :- v Satisfy condition $w^T \phi(x_n) t_n > 0$

All patterns must.

→ perception criterion is given by

$$E_p(w) = -\sum_{n \in M} w^T \phi_n t_n$$

M → denotes the set of misclassified patterns

→ we apply gradient descent algorithm to this error function. The change in weight is given by

$$w^{(\tau+1)} = w^{(\tau)} - \eta \nabla E_p(w)$$
$$= w^{(\tau)} + \eta \phi_n t_n .$$

where η → learning rate

τ → Integer values of steps of algorithm.

note :- As the weight are change during training the misclassified pattern will change.
set of

Probabilistic Discriminative Models :-

→ These are 2 Models

i) Generative

ii) discriminative.

Analogy :-

Task → Determine the language that someone is speaking

i) Generative approach :- Learn Each language & determine as to which language the speech belongs to .

ii) Discriminative approch :- Determine the linguistic differences without learning any language

→ We wish to lear $f : x \rightarrow y$ eg.., $P(y|x)$

i) In generative we model the joint distribution of all data

ii) In discriminative we model only points at the boundary.

Note :- Discriminative models make predictions on the unseen data based on conditional probability

→ Discriminative classification is also called as "informative".

Probability Basics :-

* Prior, conditional & joint probability for random variables.

1) Prior Probability - $P(x)$

2) Conditional probability - $P(x_1|x_2)$, $P(x_2|x_1)$

3) Joint probability :- $x = (x_1, x_2)$ $P(x) = P(x_1, x_2)$

4) Relationship $P(x_1, x_2) = P(x_2|x_1)\,P(x_1)$

$= P(x_1|x_2)\,P(x_2)$

5) Independence

$P(x_2|x_1) = P(x_2)$, $P(x_1|x_2) = P(x_1)$, $P(x_1, x_2) = P(x_1)P(x_2)$

Bayes rule :-

Generative.

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

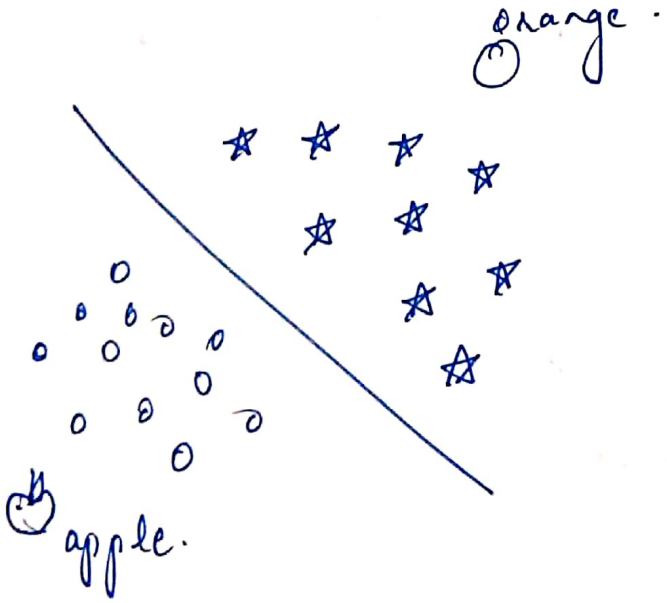
$P(c|x)$ → discriminative

$$\text{Posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{Evidence}}.$$

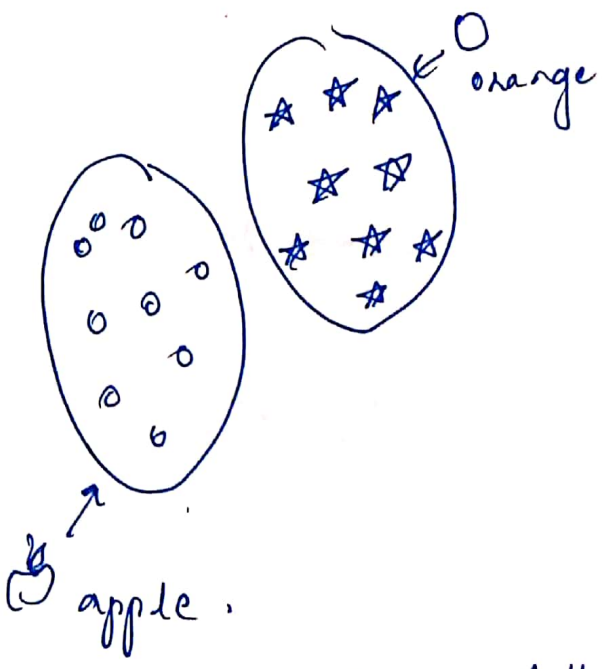
→ Given some data points, the discriminative model learns to classify the data points into their respective classes by learning the decision boundary that separate the classes.

→ The generative models can also classify the given data points, but instead of learning the decision boundary, they learn the characteristics of the classes.

Discriminative model



Generative Model



* Above shows Example of image classification, whether an Input Image is an apple/orange.

i) Discriminative Model learns the optimal

boundary that seperates the apple & orange classes

ii) Generative model learn their distribution by learning the characteristics of apple & orange classes.

→ while both types of models are used to predict $P(y|x)$. Calculation methods for both Models are different.

i) Generative Model → first find joint probability distribution $P(x,y)$, then use Bayes theorem to calculate $P(y|x)$

ii) Discriminative Model → Will learn the conditional probability $P(y|x)$ directly.

Ex :- Data set (0,1) (0,1) (1,0) (1,1)

$P(x,y)$	$y=0$	$y=1$	
$x=0$		$1/2$	$\Rightarrow P(x_1)$
$x=1$	$1/4$	$1/4$	$\Rightarrow P(x_2)$
	$P(y)=1/4$	$P(y)=1/8$	

and $P(y|x) = \frac{P(x,y)}{P(x)}$ or $\frac{P(x,y)}{P(y)}$

$P(x_1) = 1/2 \qquad P(x_2) = 1/2$

$\therefore$

$P(y\|x)$	$y=0$	$y=1$
$x=0$		1
$x=1$	$1/2$	$1/2$

$\frac{P(x,y)}{P(y)} = \frac{y_2}{y_2}$

Summary :-

Generative Model

→ Works good for small data & delivers high accuracy.

Discriminative Model

→ Accuracy is high

→ Resource saving

→ Fewer parameters to be determined.

Probabilistic Models → Discriminative → Fixed Basis function

→ Logistic regression

→ Iterative reweighted least-Square

→ multiclass logistic regression

Tools used for ←

Logistic Regression :-

→ Here we model the posterior probabilities directly assuming that they have a sigmoid-shaped distribution [without Modeling class priors & class conditional densities]

→ The sigmoid-shaped function (σ) is a model function logistic regression.

→ 1st non-linear transformation of inputs using a vector of basis function $\phi(x)$

$$P(C_1|\phi) = y(\phi) = \sigma(w^T\phi)$$

$$P(C_2|\phi) = 1 - P(C_1|\phi)$$

∴ If we have M-dimensional feature space ϕ, we use M+1 adjustable parameters to represent it directly.

→ Here $\sigma(\cdot)$ → logistic sigmoid function. or logistic regression.

→ We can use maximum likelihood to determine the parameters of the logistic regression model.

$$\frac{d\sigma}{da} = \sigma(1-\sigma)$$

→ For a data set $\{\phi_n, t_n\}$, $t_n \in \{0, 1\}$ & $\phi_n = \phi(x_n)$ with $n = 1 \ldots N$.

likelihood function can be written as

$$P(t|w) = \prod_{n=1}^{N} y_n^{t_n} \{1 - y_n\}^{1-t_n}$$

where $t = \{t_1, \ldots t_N\}^T$

$y_n = P(c_1|\phi_n)$ & Error Equation

$$\nabla E(w) = \sum_{n=1}^{N} (y_n - t_n)\phi_n$$

Iterative reweighted least squares :-

→ Logistic regression has no closed-form solution of $\nabla E(w) = 0$. due to non-linearity of logistic sigmoid function.

→ ∴ Error function can be minimized by an efficient iterative technique.

i.e., $w^{(new)} = w^{(old)} - H^{-1} \nabla E(w)$

where $H \rightarrow$ Hessian matrix $\rightarrow$ 2nd derivatives of $E(w)$ wrt 'w'

$$H = \nabla^2 E(w)$$

& $\nabla E(w) = \phi^T \phi w - \phi^T t$.

$$\therefore H = \nabla \nabla E(w) = \sum_{n=1}^{N} \phi_n \phi_n^T = \phi^T \phi$$

$\downarrow$

design matrix