INF8953CE - Fall 2020

Machine Learning

- Sarath Chandar

5. Classification

5. Classification

Consider a k-class classification problem.

$$c \in \{c_1, \ldots, c_k\}$$

Two stages in classification:-

- Inference stage : use training data to learn a model for $P(C_k|x)$
- Decision stage : use posterior probabilities to make optimal class assignments.

$$\hat{c}(x) = \underset{c \in C}{\operatorname{argmin}} \sum_{k=1}^{k} L(C_k, c) \Pr(C_k | x = x)$$

↑ Price for classifying an obs. belonging to class C_k to class c.

For 0/1 loss fn:

$$\hat{c}(x) = \underset{c \in C}{\operatorname{argmin}} \left[1 - \Pr(c | x = x)\right]$$

$$= \underset{c \in C}{\operatorname{argmax}} \Pr(c | x = x)$$

↳ Bayes classifier.

3 approaches to solving classification problems:-

① → First solve the inference problem of determining the class-conditional densities $p(x|C_k)$ for each class C_k individually.

→ Also infer prior class probabilities $p(C_k)$

→ Then use Bayes theorem:

prior

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

Posterior class probabilities.

where $p(x) = \sum_k p(x|C_k)\,p(C_k)$

(or) we can also learn the joint distribution $p(x, C_k)$ and then normalize to obtain $p(C_k|x)$.

Once you find $p(C_k|x)$, use decision theory to determine the class.

- This is known as "generative model".

↓

we can generate synthetic data points in the input space by using the learnt distribution.

---

② Directly model $P(C_k|x)$ and then use decision theory to determine the class.

Approaches that model $P(C_k|x)$ directly are called "discriminative models."

---

③ Find a function $f(x)$ – called discriminant function – which maps x directly to class label.

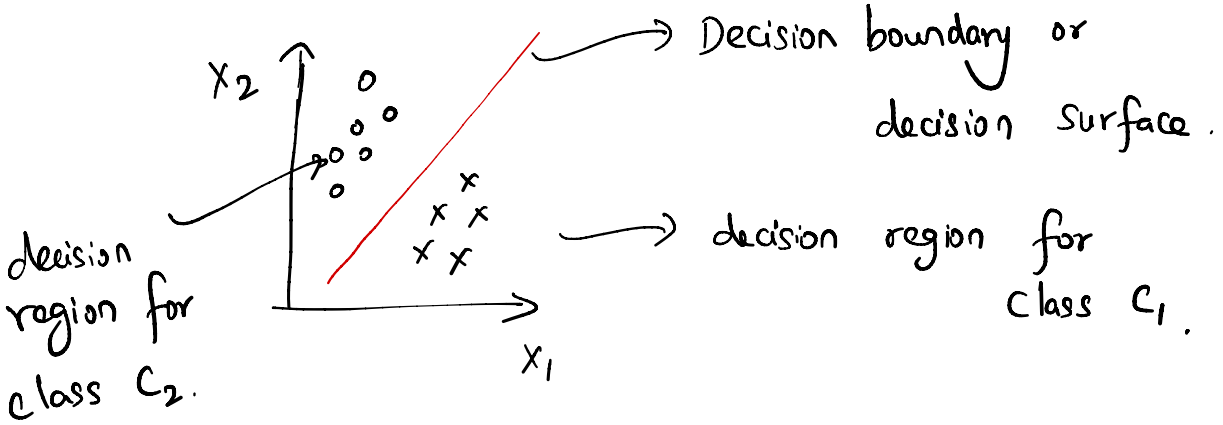
- There is no probability here.
- discriminant based models.

---

classification
- Generative models (pure probability)
- Discriminative models
- Discriminant based models.

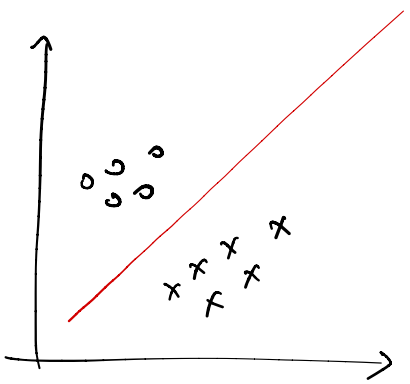
Linear classification :-

Consider a 2-class problem : x - C_1, o - C_2

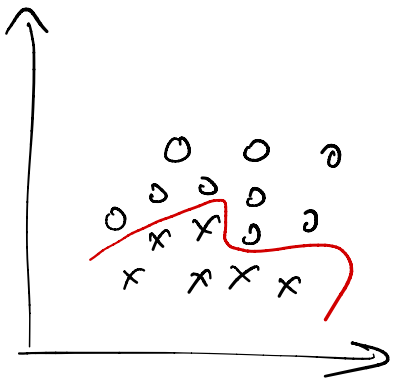


Linear model – decision surfaces are linear functions of the input vector x and hence are defined by (D-1) dimensional hyperplanes with the D-dimensional input space.

Linearly separable problems :-



linearly separable



Not linearly separable.

Target variable for classification :-

For regression, target variable $t \in \mathbb{R}$

For classification, target class $c \in \{C_1, \ldots C_k\}$

2. class problem : (C_1, C_2) Label encoding

$t = 1$ class C_1

$t = 0$ class C_2

$t \in \{0, 1\}$

t - Can be interpreted as the probability that the class is C_1.

Probability takes extreme values of 0 and 1.

$K > 2$ classes :-

① Can we consider $t \in \{1, 2, \ldots K\}$?

Problem: this representation creates artificial distance between classes. C_1 is close to C_2 than C_5.

② 1-of-k coding scheme :- one hot encoding

for $k = 5$ classes,

class 1 : $t = (1, 0, 0, 0, 0)^T$

class 2 : $t = (0, 1, 0, 0, 0)^T$

and so on.

Each t_k can be interpreted as the probability that the class is C_k.

Discriminant functions :-

- takes an input vector x and assigns it to one of k classes, denoted as C_k.
- linear discriminants → decision surfaces are hyperplanes.

2-class scenario :-

Simple linear discriminant function :-

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

w: weight vector, w_0: bias

if $y(x) \geq 0$ then C_1

if $y(x) < 0$ then C_2

decision boundary : $y(x) = 0$

w_0 - can be considered as a threshold.

For example, if $w_0 = -7$, then $w^T x$ has to be atleast $+7$ to be classified as C_1.

Consider two points x_A and x_B both of which lie on the decision surface.

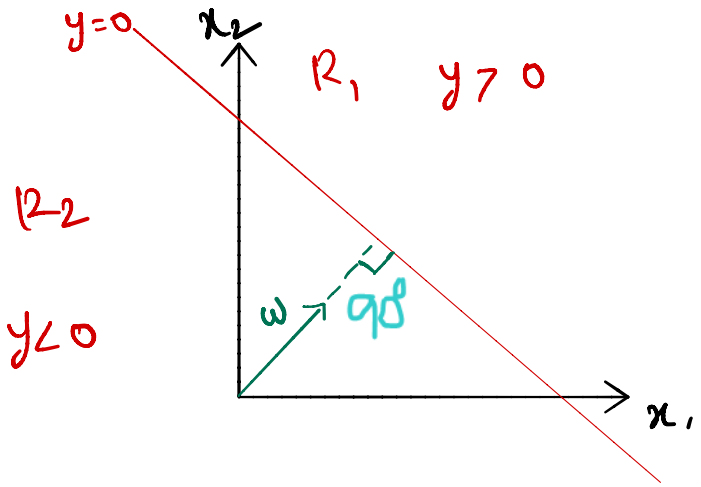
$$y(x_A) = 0 \qquad w^T x_A + w_0 = 0$$

$$y(x_B) = 0 \qquad w^T x_B + w_0 = 0$$

$$w^T (x_A - x_B) = 0$$

$\Rightarrow$ w vector is orthogonal to every vector lying within the decision surface.

$\Rightarrow$ w determines the orientation of the decision surface.



to find the shotest distance to the decision line. ie which is orthogonal is the shortest

We know that ω is orthogonal to the decision Surface.

Any point x' on this decision surface that is closest to the origin can be represented as

$$x' = \alpha \omega$$ for some scalar α.

x' is on the decision surface.

$$\omega^T x' + \omega_0 = 0$$

$$\alpha \omega^T \omega + \omega_0 = 0$$

$$\alpha = \frac{-\omega_0}{\|\omega\|^2}$$

The distance from x' to the origin is

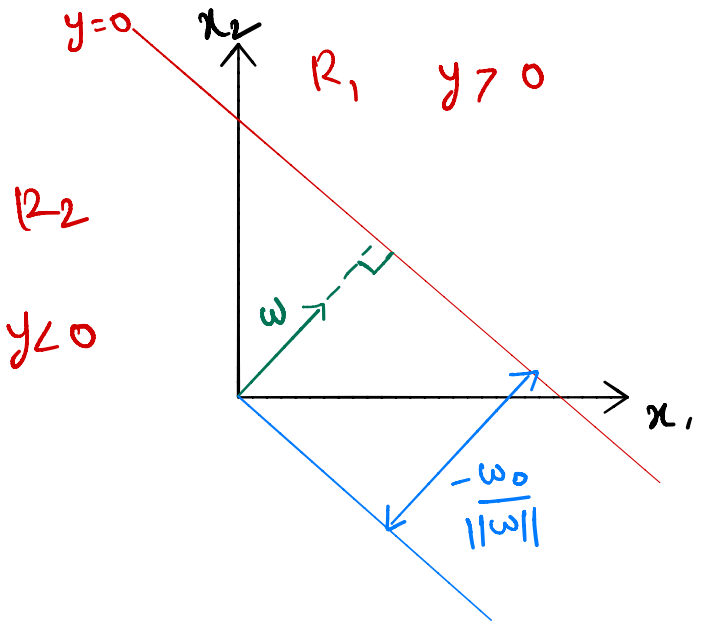
$$\|x'\| = \|\alpha \omega\|$$

$$= \alpha \|\omega\|$$

$$= -\frac{w_0}{\|w\|^2}\|w\|$$

$$= -\frac{w_0}{\|w\|}$$

w_0 determines the location of the decision function.



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

Consider an arbitrary point x and let $x_\perp$ be its orthogonal projection onto the decision surface, so that

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

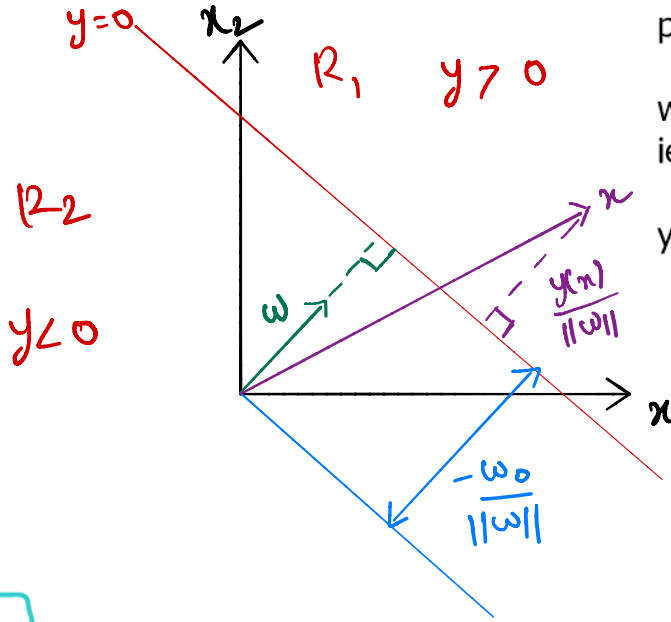
$$w^T x = w^T x_\perp + r \frac{w^T w}{\|w\|}$$

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

$$y(x) = y(x_\perp) + r \frac{w^T w}{\|w\|}$$

$$y(x) = 0 + r \cdot \|w\|$$

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



w- decides the orientation ie perpendicular

w0--> how far it is from the origin ie. displacement

y(x)--> sign, to determine the class

Summary:-

① Decision surface is perpendicular to w.

② Displacement of decision surface is controlled by the bias parameter.

③ Signed orthogonal distance of a general point x from the decision surface is given by $\frac{y(x)}{\|w\|}$.

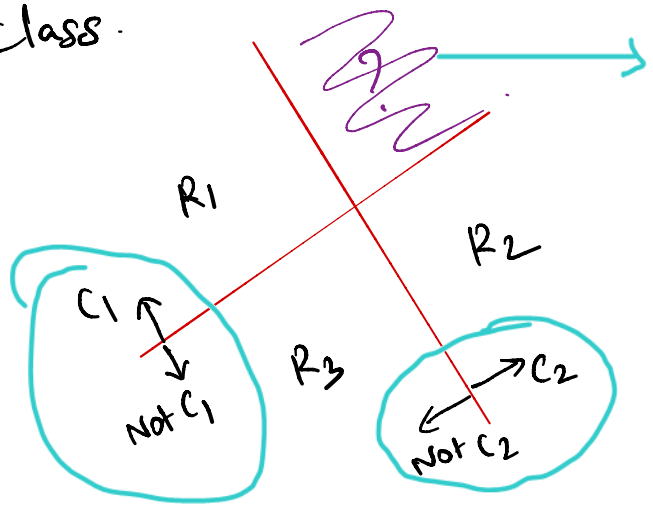
Magnitude--> tells how far from decision surface

Sign--> tells what is the class

Multiple classes :-

① One-versus-the-rest classifier :-

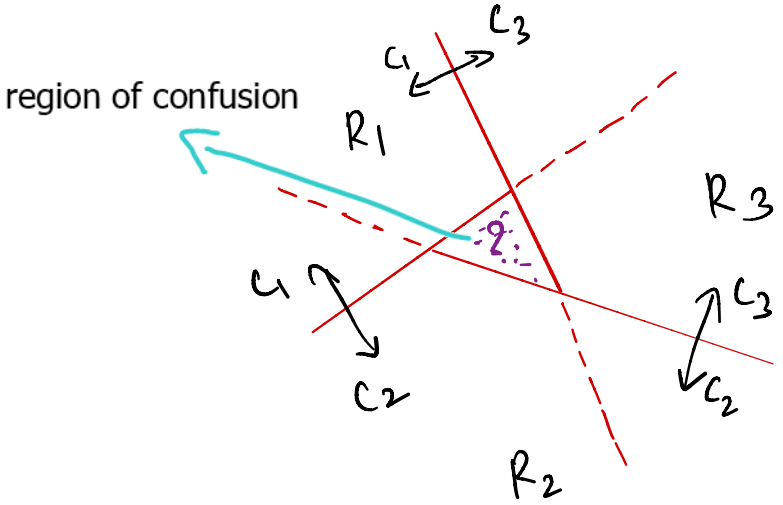
$K-1$ classifiers each of which solves a two-class problem of separating points in class C_k from points not in that class.



② one-versus-one classifier :-

Consider $k(k-1)/2$ binary discriminant functions, one for every possible pair of classes.

- Majority voting among the discriminant functions for classification.



C_1, C_2, C_3

C_1 vs C_2

C_1 vs C_3

C_2 vs. C_3

(3) Consider a single k-class discriminant comprising k-linear functions of the form

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

if $y_k(x) > y_j(x)$ for all $j \neq k$, then x belongs to C_k.

Decision boundary between C_k and C_j:

$$y_k(x) = y_j(x)$$

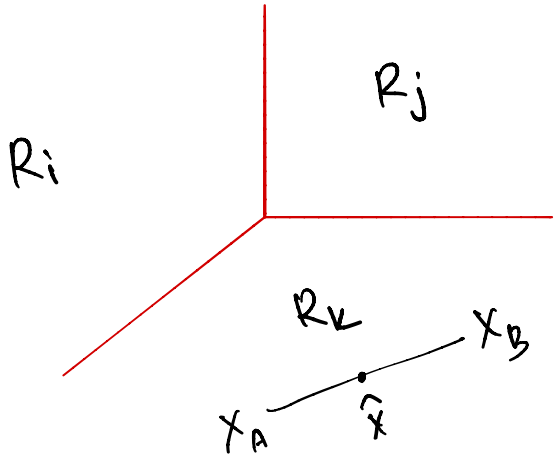
$$w_k x + w_{k0} = w_j x + w_{j0}$$

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

→ similar to decision boundary for two classes.

→ so analogous geometrical properties apply.

Note: Decision regions of such a discriminant are always singly connected and convex.



it is better than the previous 2 approaches because there is no confusion region

Consider $x_A, x_B \in R_k$

Any point $\hat{x}$ that lies on the line connecting x_A and x_B can be expressed as

then x^ lies in that same region as Xa and Xb

$$\hat{x} = \lambda x_A + (1-\lambda) x_B \qquad 0 \le \lambda \le 1$$

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

[by linearity of discriminant]

we know that $y_k(x_A) > y_j(x_A)$ and

$y_k(x_B) > y_j(x_B)$ for all $j \neq k$.

$$\Rightarrow \quad y_k(\hat{x}) > y_j(\hat{x}) \quad \text{for all } j \neq k.$$

$$\text{So} \quad \hat{x} \in R_k.$$

Thus R_k is singly connected and convex.

Least Squares for classification:-

→ classification problem with 'k' classes.

→ 1-of-k binary coding scheme for the target vector t.

→ We know that least squares approximates the conditional expectation $E[t|x]$ of the target values given the input vector. For the binary coding scheme, this conditional expectation is given by the vector of posterior probabilities. So it makes sense to use least squares.

→ However, there is no guarentee for the values to be in the (0,1) range!

Each class C_k is described by its own linear model:

$$y_k(x) = w_k^T x + w_{k0} \quad \text{for } k = 1 \ldots K$$

Vector notation:-

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

$\tilde{W}$ is a matrix whose k^{th} column comprises the $D+1$ dimensional vector $\tilde{w}_k = (w_{k0}, w_k^T)^T$.

$\tilde{x}$ - is the augmented input vector $(1, x^T)^T$.

→ Consider training data set: $\{x^{(n)}, t^{(n)}\}_{n=1}^{N}$

→ define a matrix T whose n^{th} row is the vector $t^{(n)T}$, together with a matrix $\tilde{X}$ whose n^{th} row is $\tilde{x}^{(n)T}$

→ Sum of square error fn:

$$E_D(\tilde{W}) = \frac{1}{2} Tr\left\{(\tilde{X}\tilde{W} - T)^T (\tilde{X}\tilde{W} - T)\right\}$$

Solution :- $\tilde{W} = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T T$

$= \tilde{X}^+ T$

↳ Pseudo inverse.

discriminant function : $y(x) = \tilde{W}^T x = T^T (\tilde{X}^+)^T x.$

Note 1: If every target vector in the training set satisfies some linear constraint

$$a^T t^{(n)} + b = 0$$

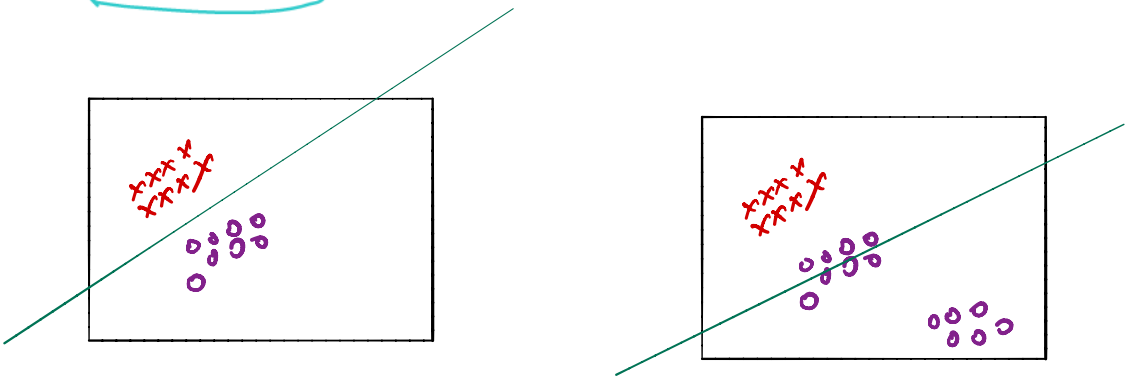
for some constants a, b, then the least squares model prediction for any value of x will satisfy the same constraint so that

$$a^T y(x) + b = 0$$

Since we use 1-of-k encoding, model's predictions will sum to 1 for any value of x. But the values are not constrained to be in the range $(0,1)$!

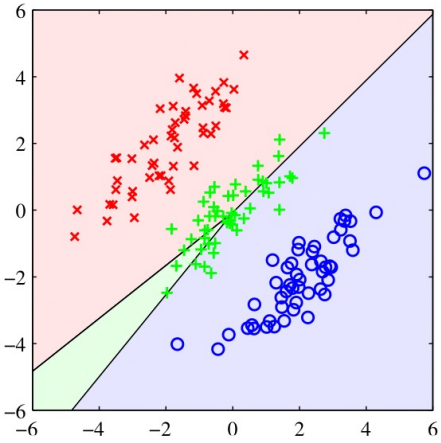
ie there can be negative too.

Note 2: Least-squares solutions lack robustness to outliers.



The sum of square error function penalizes predictions that are "too correct" in that they lie a long way on the correct side of the decision boundary.

Note 3: Doesn't work well in many cases.



only small region assigned to green class!

You should know!

1. 2 stages of classification
   - Inference
   - Decision.
2. 3 approaches to classification
   - Generative model
   - Discriminative model
   - Discriminant-based model.
3. linear classification, linear decision boundary, linear model, linearly separable problems.
4. 1-of-k coding scheme / 1-hot vector.
5. Discriminant functions.
6. Geometric interpretation of $y(x) = wx + w_0$.
7. Multiple classes
   - 1 vs. rest classifier
   - 1 vs. 1 classifier
   - k-class discriminants.