Page No.
Date : / /

Bayes Decision Problem

↓

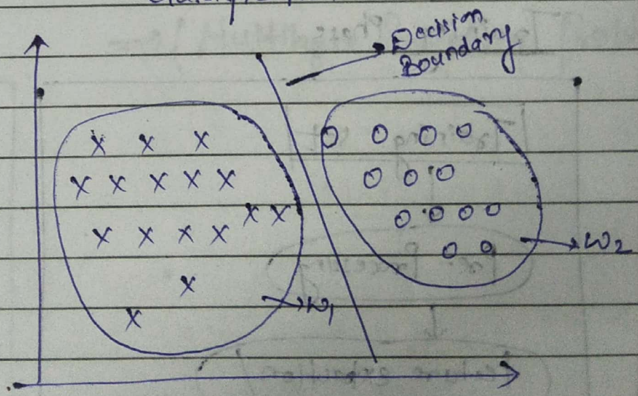
Statistical Pattern Recognition

→ Many of the features taken together, they can describe an object to some degree of accuracy. These features are called feature vectors.

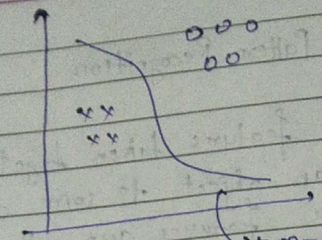
→ They had to be concatenated in a particular order & whichever order we concatenate them throughout our model is modelling of the pattern as well as recognition of pattern. problem.

→ When, I put all these diffⁿ features in a particular order, they are called feature vector.

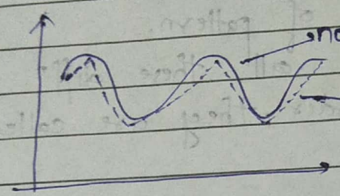
Dimension → no. of features considered. for classification.



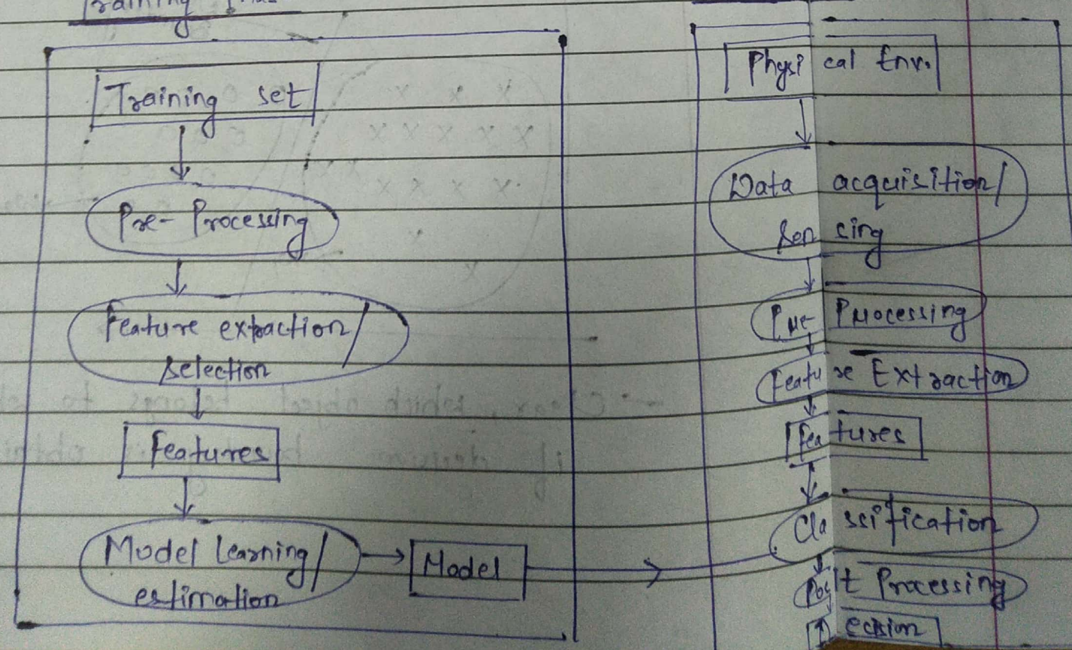
→ Clear, which object belongs to which class if decision boundary is obtained.



non-linear boundary
(may be cubic or quadratic)
↓
above this degree,
we won't be able to perform
well.



Training Phase   Test Phase



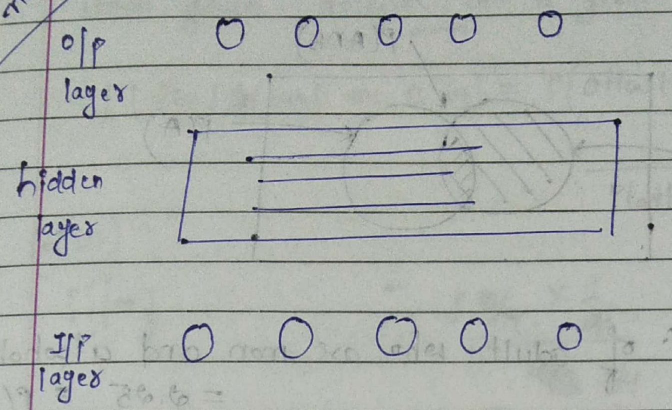
Structure in neural network.

o/p layer

hidden layer

I/P layer

Structure in neural network.



$$\boxed{O_i = f\left(\sum w_{ij} x_i\right)}$$

→ Overlapping boundaries ~~are~~ makes it tough to be distingaished for patterns.

(neither linear nor non-linear)

⟹ (Multilayer Perceptron Problem)

l Env.

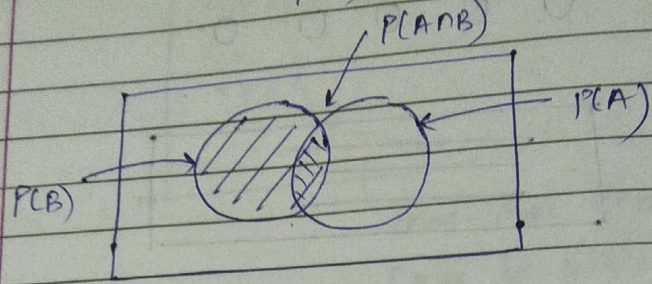
quisition/
g

ocessing

Extraction

Conditional Problality :-

$$P(A|B) = \frac{P(A\cap B)}{P(B)}$$



Eg. The % of adult who are man and alcoholic $= 2.25 = P(A\cap M)$

Men and women are assumed to be equal.

$$P(A|M) = \frac{P(A\cap M)}{P(M)} = \frac{2.25}{100\times 0.5} = \frac{0.0225}{0.5}$$

$$= \frac{45\ \cancel{225}\times 10}{\cancel{5}\times 10000} = 0.045$$

Baye's Theorem

$$P(A|B) = \frac{P(A\cap B)}{P(B)}$$

$$P(B|A) = \frac{P(A\cap B)}{P(A)}$$

Now,

$$\boxed{P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}}$$

Date : / /

Q. What is the probablity of 2 girls in a couple. when given atleast, one is girl.

$$P(2G \mid \text{atleast one is } G) = \frac{P(\text{atleast one is } G \mid 2G) \times P(2G)}{P(\text{atleast one is } G)}$$

(2G) → $\frac{1}{3}$

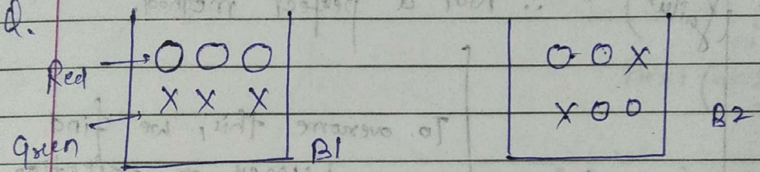
$$= \frac{1 \times \frac{1}{4}}{3/4}$$

GB
BG
GG
BB

possible combination.

$$= \frac{1}{3}$$

Q.



If, I randomly draw a green ball, What is the probablity of having it in B1.

$$P(B1 / G) = \frac{P(G \mid B1) \times P(B1)}{P(G)}$$

$$= \frac{\frac{1}{2} \times \frac{1}{2}}{\left(\frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \frac{\cancel{2}}{\cancel{6}_{3}}\right)} = \frac{\frac{1}{4}}{\left(\frac{1}{4} + \frac{1}{6}\right)}$$

$$= \frac{\frac{1}{4}}{\frac{6+4}{24}} = \frac{1}{4} \times \frac{\cancel{24}^{6}}{10} = \frac{3}{5}$$

Bayer Decision Theory

ω_1 Accept $P(\omega_1)$

ω_2 Reject $P(\omega_2)$

$$P(\omega_1) > P(\omega_2) \Rightarrow \omega_1$$
$$P(\omega_1) < P(\omega_2) \Rightarrow \omega_2$$

(a priority probablity)

* if (.90 is success probablity, .10 is rejection ")

⇒ machine will always accept in this case. $\because P(\omega_1) > P(\omega_2)$

(failure) ∴ Not a perfect method.

To overcome this, we find success or rejection based on some feature using feature vector.

Given class:-

feature is known & we need to find class.

$$\underbrace{P(x|\omega_1) \qquad P(x|\omega_2)}_{\text{Class Cond}^n \text{ PDF}}$$

(Probablity Distribution function)

$$= P(\omega_i \cap x) \leftarrow P(\omega_i, x) = P(\omega_i|x).P(x)$$
$$= P(x|\omega_i).P(\omega_i)$$

Page No.
Date : / /

$$P(\omega_i/x) \cdot P(x) = P(x/\omega_i) \cdot P(\omega_i)$$

$$\Rightarrow \boxed{P(\omega_i/x) = \frac{P(x/\omega_i) \cdot P(\omega_i)}{P(x)}}$$

Bayes form.

if, x i.e. a feature is given then we need to find its class.

⇒ [In current Eg. classes = 2, feature = 1]

$$P(x) = \sum_{i=1}^{2} P(x/\omega_i)\, P(\omega_i)$$

** { In general, $i \in [1 \text{ to } n]$ }

no. of classes

Here, (n=2)

1. $P(\omega_1/x) > P(\omega_2/x) \rightarrow \omega_1$ (accept)
2. $P(\omega_1/x) < P(\omega_2/x) \rightarrow \omega_2$ (reject)

Now,

1. $$\frac{P(x/\omega_1)\, P(\omega_1)}{P(x)} > \frac{P(x/\omega_2)\, P(\omega_2)}{P(x)}$$

$$\Rightarrow \boxed{P(x/\omega_1)\, P(\omega_1) > P(x/\omega_2)\, P(\omega_2)}$$

presence of 'x' in bucket 'ω₁'

presence of 'x' in 'ω₂'

already learned at the time of training

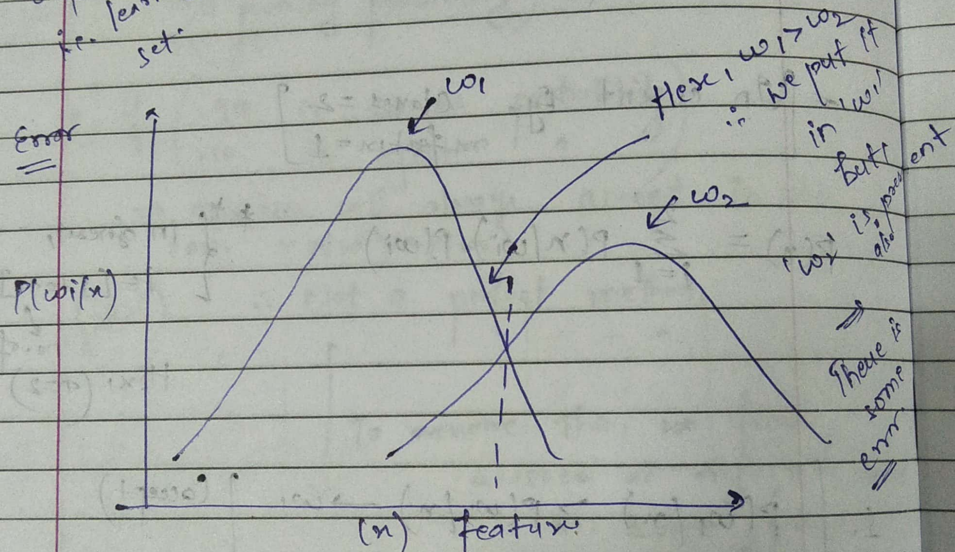
Now, assume, $\boxed{P(\omega_1) = P(\omega_2)}$ in ideal case.

Eq^n reduces to,

$$\boxed{P(x|\omega_1) > P(x|\omega_2)} \rightarrow \omega_1 \text{ (accept)}$$

* completely depends on feature 'x' i.e. learning of training set.

Error



✓ Error when x is placed in 'ω_1' :- $\boxed{P(\omega_2|x)}$

✓ Error when x is placed in 'ω_2' :- $\boxed{P(\omega_1|x)}$

(overall, error)

$$P(error) = \int_{-\infty}^{\infty} P(error, x)\, dx$$

$$\boxed{P(error) = \int_{-\infty}^{\infty} P(error/x) \cdot P(x)\, dx.}$$

$$P(error/x) = \min\{P(\omega_1/x), P(\omega_2/x)\}$$

ω_1	ω_2
$\downarrow$	$\downarrow$
accept	Reject
$P(\omega_1)$	$P(\omega_2)$

$P(\omega_1) > P(\omega_2) \Rightarrow \omega_1$

$\Rightarrow \omega_2$

$$\underbrace{P(x/\omega_1) \qquad\qquad P(x/\omega_2)}$$

Class Conditional PDF (Probablity Density Function)

$P(\omega_1/x)$ $\qquad$ $P(\omega_2/x)$

$$P(\omega_1, x) = \frac{P(x/\omega_1) \cdot P(\omega_1)}{P(x)}$$

$$P(\omega_2/x) = \frac{P(x/\omega_2)\,P(\omega_1)}{P(x)}$$

$$P(x) = \sum_{i=1}^{2} P(x/\omega_i)\cdot P(\omega_i)$$

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

a postiori probablity.

$$\frac{P(x/\omega_i)\,P(\omega_i)}{\cancel{P(x)}} > \frac{P(x/\omega_2)\,P(\omega_2)}{\cancel{P(x)}} \Rightarrow \omega_1$$

** # Priori Prob.

Que

1.	Quality of Course	good	fair	bad
2.	Prob (Priori)	0.2	0.4	0.4
3.	Class Conditionals $P(x/\omega_i)$	good	fair	bad
4.	Interesting	0.8	0.5	0.1
5.	Boring	0.2	0.5	0.9

Loss function :-

	$P[\lambda(a_i/\omega_i)]$	good	fair	bad
1.				
2.	Taking	0	5	10
3.	Not taking	20	5	0

Generalisation :-

* Use more than 2 states of nature
  ↳ classes

* Use more than 1 features
  ↳ feature vector

* Allow other action other than merely deciding states of nature

* Introduce a loss function more general than Probablity of error.

c → states of nature

$\omega_1, \omega_2, \omega_3 \ldots \omega_c$

a → Actions

$\{\alpha_1, \alpha_2, \ldots, \alpha_a\}$

Loss function →

$\lambda(\alpha_i/\omega_j)$ → loss incurred error

taking action α_i when true state of nature is ω_j.

$X \Rightarrow d$ - dimensional feature vector

Now,

X

Action α_i

$$R(\alpha_i/x) = \sum_{j=1}^{c} \lambda(\alpha_i/\omega_j)\, P(\omega_j/x)$$

function

NOTE:-

* action with least risk factor will be the desired one.

Risk Function / conditional Risk / Expected loss

* (It is calculated for all values of α_i)

$\Rightarrow$ Min Risk Classifier

Eg. Class :- ω_1 ω_2

Action:- α_1 α_2

$$\lambda(\alpha_i/\omega_j) = \lambda_{ij}$$

$$\boxed{R(\alpha_i/x) = \sum_{j=1}^{c} \lambda(\alpha_i/\omega_j)\, P(\omega_j/x)}$$

$$\rightarrow R(\alpha_1/x) = \sum_{j=1} \lambda_{11}(\alpha_1/\omega_1) P(\omega_1/x) + \lambda_{12}(\alpha_1/\omega_2) . P(\omega_2/x)$$

$$\Rightarrow \boxed{R(\alpha_1/x) = \lambda_{11} P(\omega_1/x) + \lambda_{12} P(\omega_2/x)}$$

Date: / /

$\longrightarrow R(\alpha_2|x) = \lambda_{21} P(\omega_1|x) + \lambda_{22} (P(\omega_2|x))$

* We assumed that $R(\alpha_2|x) > R(\alpha_1|x) \Longrightarrow$ we should select 'ω_1'.

Now,

$$\boxed{(\lambda_{21}-\lambda_{11}) \cdot P(\omega_1|x) > (\lambda_{12}-\lambda_{22}) P(\omega_2|x)}$$

$\simeq$ ly

belongs to wrong class α_2 but ω_1 $\Rightarrow$ +ve.

belong to same $\Rightarrow 0$

$\Rightarrow (\lambda_{21}-\lambda_{11}) > 0$

$$P(\omega_1/x) > P(\omega_2/x) \Rightarrow \omega_1$$

$$(\lambda_{21}-\lambda_{11}) P(\omega_1|x) > (\lambda_{12}-\lambda_{22}) P(\omega_2|x)$$

$$\xrightarrow{11} \omega_1$$

Min Error Rate Classification :-

$\alpha_i \rightarrow$ True state of nature is ω_i

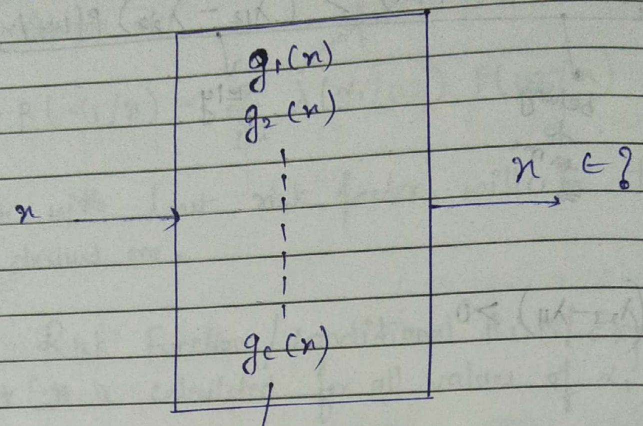
$$\lambda(\alpha_i|\omega_j) = \begin{cases} 0 & ; i=j \\ 1 & ; i\neq j \end{cases} \quad \forall\, i,j = 1,2,\dots,c$$

$$R(\alpha_i|x) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j) \cdot P(\omega_j|x)$$

$$= \boxed{\sum_{i\neq j} P(\omega_j|x)} \Longrightarrow$$ for this case, $\lambda(\alpha_i|\omega_j)=1$

for $i=j$, $\lambda(\alpha_i|\omega_j) = 0$.

$$= \boxed{1 - P(\omega_i/x)}$$ → easier to calculated probability of one.



C is the no. of classes
(CLASSIFIER)

$g(x)$ → Discriminant F^n

$\omega_1, \omega_2, \omega_3, ----, \omega_c$ → No. of classes

$g_i(x)$: $i = 1 ---- c$

$\boxed{g_i(x) > g_j(x)} \; \forall j \neq i$

$\Rightarrow x \in \omega_i$

(min risk will always be -ve)

* Min Risk Classifier

$$\boxed{g_i(x) = -R(\alpha_i/x)}$$ → ∵ we take risk to be min. ∴ -ve of that will be max

* MIN ERROR RATE

$$\sum_{i \neq j} P(\omega_j / x) \rightarrow 1 - P(\omega_i / x)$$

$$g_i(x) = \sum_{i \neq j} P(\omega_j | x) = 1 - P(\omega_i | x)$$

$$g_i(x) = -R(\alpha_i | x)$$
$$= -(1 - P(\omega_i | x))$$
$$= -1 + P(\omega_i | x)$$

$$\boxed{g_i(x) = P(\omega_i | x)}$$

$$g_i(x) = P(\omega_i | x)$$
$$= \frac{P(x/\omega_i) \cdot P(\omega_i)}{\sum_{j=1}^{c} P(x/\omega_j) P(\omega_j)} \rightarrow P(x)$$

* IMP
we have removed P(x) i.e. denominator bcz all main concern is comparison that's why we have removed denominator bcz it is same for all.

$$\boxed{g_i(x) = P(x/\omega_i) P(\omega_i)}$$

- We have taken log bcz product is a costly operation.

$$\Rightarrow \ln(g_i(x)) = \ln(P(x/\omega_i)) + \ln(P(\omega_i))$$

$$\text{if } (g_1(x) > g_2(x)) \Rightarrow \omega_1$$

$g_1(x) = g_2(x) \rightarrow$ decision boundary.

$$g(x) = P(\omega_1/x) - P(\omega_2/x)$$
$$= P(x/\omega_1) \cdot P(\omega_1) - P(x/\omega_2) \cdot P(\omega_2)$$

{ -ve sign bcz. if we get +ve then ω_1 else ω_2. }

Date: / /

again log both sides.

$$\ln(g(x)) = \ln\left(\frac{P(x|\omega_1)\cdot P(\omega_1)}{P(x|\omega_2)\cdot P(\omega_2)}\right)$$

$$= \ln\left(\frac{P(x|\omega_1)\cdot P(\omega_1)}{P(x|\omega_2)\cdot P(\omega_2)}\right)$$

$$= \ln\left(\frac{P(x|\omega_1)}{P(x|\omega_2)}\cdot\frac{P(\omega_1)}{P(\omega_2)}\right)$$

$$\boxed{\ln(g(x)) = \ln\left(\frac{P(x|\omega_1)}{P(x|\omega_2)}\right) + \ln\left(\frac{P(\omega_1)}{P(\omega_2)}\right)}$$

Now,

Eg. Data set

1	R	Sports	Domestic	Yes
2	R	"	"	N
3	R	"	"	Yes
4	Y	"	"	N
5	Y	"	Imp	Yes
6	Y	SUV	"	N
7	Y	SUV	"	Yes
8	Y	SUV	Domestic	N
9	R	SUV	Imp	N
10	R	Sports	Imp	Yes.

Color :
$P(R/Yes) = 3/5$ $P(R/No) = 2/5$
$P(Y/yes) = 2/5$ $P(Y/No) = 3/5$

Type :
$P(SUV/Yes) = 1/5$ $P(SUV/No) = 3/5$
$P(Sports/Yes) = 4/5$ $P(sports/No) = 2/5$

Origin :
$P(Dom/yes) = 2/5$ $P(Dom/No) = 3/5$
$P(Imp/yes) = 3/5$ $P(Imp/No) = 2/5$

$P(yes) = 5/10$
$P(No) = 5/10$

Q. (Red, SUV, Dom) → find, Yes class or No class.

$$P\left(yes/(Red, SUV, Dom)\right) = \frac{P(yes, Red, SUV, Dom)}{P(Red, SUV, Dom)}$$

$$\left(\frac{3}{5} \cdot \frac{1}{5} \cdot \frac{2}{5}\right)$$

$$P\left(No/(Red, SUV, Dom)\right) = \frac{P(No, Red, SUV, Dom)}{P(Red, SUV, Dom)}.$$

∵ Comparison is to be done, we remove denominator.

$$P(x/yes) \cdot P(yes) = P(Red/yes) \cdot P(SUV/yes) \cdot P(Dom/yes) \cdot P(yes)$$

$$= \frac{3}{5} \cdot \frac{1}{5} \cdot \frac{2}{5}$$

Page No.
Date : / /

** $\because P(yes) = P(No)$
$\therefore$ It has not been taken

$$P(x/No)\, P(No) = P(Red/No) \cdot P(SUV/No) \cdot P(Dom/No) \cdot P(No)$$

$$= \frac{2}{5} \cdot \frac{3}{5} \cdot \frac{3}{5} \cdot$$

On comparing, ans will be No

** Que Find optimal sol^n

$\Rightarrow$ Course should be (good, interesting)

$$P((good, interesting)/taking) \cdot P(taking)$$

$$= P(good/taking) \cdot P(goo\ interesting/taking) \times P(taking)$$

$$= 0$$

$$P(good, interesting)/nontaking) \cdot P(nontaking)$$

$$= P(good/nontaking) \cdot P(interesting/nontaking) \cdot P(nontaking)$$

$$= \cancel{20}$$

Scanned by CamScanner

Date: / /

*Que-Ans optimise solution

Quality of course	good	fair	bad
Prob (Prior)	0.2	0.4	0.4

Class Conditionals $Pr(x/\omega)$	good	fair	bad
→ Interesting	0.8	0.5	0.1
→ Boring	0.2	0.5	0.9

Loss function $\lambda(a_i/\omega_j)$	good	fair	bad
→ Taking the course	0	5	10
→ Not taking	20	5	10

Ans $P_r(\text{Interesting}) = P(\text{Interesting}/\text{good}) \cdot P(\text{good}) + P(I/\text{fair}) \cdot P(\text{fair}) + P(I/\text{bad}) \cdot P(\text{bad})$

$= 0.8 \times 0.2 + 0.5 \times 0.4 + 0.1 \times 0.4$

$= 0.16 + 0.20 + 0.04 = 0.40$

$P_r(\text{Boring}) = 1 - 0.4 = 0.6$

$$P(\text{good}/\text{Interesting}) = \frac{P(\text{Interesting}/\text{good}) \times P(\text{good})}{P(\text{Interesting})}$$

$$= \frac{\cancel{0.8}^{2} \times 0.2}{\cancel{0.4}} = 0.4$$

$P(fair | Interesting) = \frac{P(I|f) \times P(f)}{P(I)}$

$= \frac{0.5 \times 0.4}{0.4}$

$= 0.5$

$P(boring | Interesting) = \frac{P(I|b) \times P(b)}{P(I)}$

$= 1 - (0.5 + 0.4)$

$= 1 - 0.9$

$= 0.1$

$P(\alpha_i | x) = 1 - P(\omega_i | x)$

$=$

\# x = features = Interesting / boring

ω = class = good / fair / bad.

$R(taking | I) = P(g|I)\ \lambda(Taking | good)$
$+ P(f|I)\ \lambda(Taking | fair)$
$+ P(b|I)\ \lambda(Taking | bad)$

$= 0.4 \times 0 + 0.5 \times 5 + 0.1 \times 10 = 3.5$

$R(not\ taking | I) = 0.4 \times 20 + 0.5 \times 5 + 0.1(0)$

$= 10.5$

$\simeq$ly $R(taking | boring) =$
and $R(taking | Boring)$

Minimum Risk = optimal solⁿ

$$P(\text{fair} \mid \text{Interesting}) = \frac{P(I \mid f)\, P(f)}{P(I)}$$

$$= \frac{0.5 \times 0.4}{0.4}$$

$$= 0.5$$

$$P(\text{boring} \mid \text{Interesting}) = \frac{P(I \mid b) \times P(b)}{P(I)}$$

$$= 1 - (0.5 + 0.4)$$

$$= 1 - 0.9$$

$$= 0.1$$

$$P(\alpha_i \mid x) = 1 - P(\omega_i \mid x)$$

$$=$$

\# x = features = Interesting / boring

ω = class = good / fair / bad.

$$R(\text{taking} \mid I) = P(g \mid I)\ \lambda(\text{Taking} \mid \text{good}) + P(f \mid I)\ \lambda(\text{Taking} \mid \text{fair}) + P(b \mid I)\ \lambda(\text{Taking} \mid \text{bad})$$

$$= 0.4 \times 0 + 0.5 \times 5 + 0.1 \times 10 = 3.5$$

$$R(\text{not taking} \mid I) = 0.4 \times 20 + 0.5 \times 5 + 0.1(0)$$

$$= 10.5$$

$||^{ly}$ $R(\text{taking} \mid \text{boring}) =$

and $R(\text{taking} \mid \text{Boring})$

Minimum Risk = optimal sol^n

Page No.
Date : / /

$$P(\text{good} \mid \text{Boring}) = \frac{P(B/g) \times P(g)}{P(B)}$$

$$= \frac{0.2 \times 0.2}{0.6} = \frac{\cancel{2} \times 0.2}{\cancel{6}\,3} = \frac{2}{3} \times 10^{-1} = 0.66 \times 10^{-1}$$

$$= 0.066$$

$$P(\text{fair} / \text{Boring}) = \frac{P(B/f) \times P(f)}{P(B)}$$

$$= \frac{0.5 \times 0.4}{0.6} = \frac{5 \times \cancel{0.4}^{\,0.2}}{\cancel{6}\,3} = \frac{1}{3} = 0.33$$

$$P(\text{bad} / \text{Boring}) = \frac{P(B/b) \times P(b)}{P(B)}$$

$$= \frac{0.9 \times 0.4}{0.6} = \frac{9 \times 0.4}{6} = \frac{3 \times \cancel{0.4}^{\,0.2}}{\cancel{2}} = 0.6$$

$$R(\text{taking} / B) = P(g/B)\,\lambda(\text{taking}/g) + P(f/B)\,\lambda(\text{taking}/f) + P(b/B)\,\lambda(\text{taking}/b)$$

$$= 0.066 \times 0 + 0.33 \times 5 + 0.6 \times 10$$

$$= 1.65 + 6 = 7.65$$

$$R(\text{non-taking} / B) = 0.066 \times 20 + 0.33 \times 5 + 0.6 \times 0$$

$$= 1.32 + 1.65 = 2.97$$

Minimum risk is of non-taking of boring

$\therefore$ Optimal sol^n.

distance b/w two images

Distance :-

Minkowski Distance :-

$$d(x_i, x_j) = \left[(x_{i1} - x_{j1})^h + (x_{i2} - x_{j2})^h + \cdots + (x_{ir} - x_{jr})^h \right]^{1/h}$$

Euclidean Distance :-

Most popular

$$d(x_i, x_j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots}$$

$\boxed{h=2}$

Manhattan Distance :-

$$d(x_i, x_j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots$$

$\boxed{h=1}$

Sq - Euclidean :-

$$= (x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ir} - x_{jr})^2$$

Weighted Euclidean

Given weightage to each and every feature

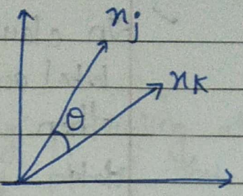
$$d(x_i, x_j) = \sqrt{w_1(x_{i1} - x_{j1})^2 + w_2(x_{i2} - x_{j2})^2 + \cdots + w_r(x_{ir} - x_{jr})^2}$$

Date : / /

Cosine

$$\cos(\vec{x}, \vec{y}) = \frac{\vec{x}\cdot\vec{y}}{|\vec{x}||\vec{y}|}$$

$$= \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$



Supervised Lerning Problem

It can be divided into regression & classification.

- Regression → It is continuous in nature.
  eg. Age of a person.

A regression is a problem when the o/p value has continuous and real value such as, {wt. of a person, salary of a person}

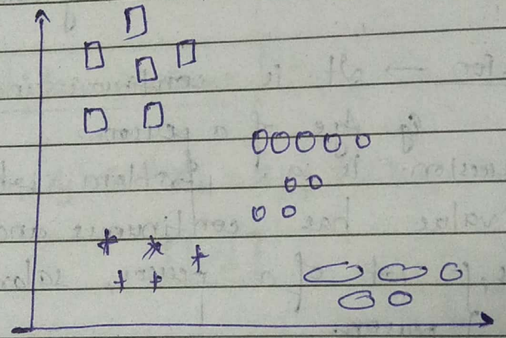
- Classification → A classification is problem is when the o/p variable has a category such as a person is male or female or diagonised it is a cancer or not.

A classification model are tend to draw some conclusion from observed value given one or more i/ps. A classification model will try to predict the value of one or more outcomes.

✓ In classification, data is categorised into diff^n label acc. to some parameters given an i/p & then the labels are predicted from the data.

✓ It can be demonstrated in the form of "if then" rules.

✓ The classification process deals with the problem where data can be divided into binary or multiple discrete classes.



If two classes are present, it is binary.

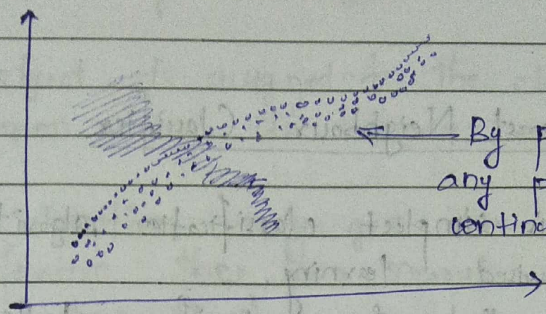
if, classes >2, multiple classes.

⇒ A regression is a process of finding a f^n or distinguishing data into continuous real value instead of using classes or discrete values.

✓ It can also identify the distinguishes movement depending on the historical data. because a regression predicting model predict a quantity. ∴ ~~scale~~ skill of model

Date: / /

must be reported as an error in these predictions.



{Difference}:-

Classification	Regression
1. Mapping fn is use to map value to define predefine classes.	1. Mapping fn is use for mapping of values to continuous o/p.
2. Classifn is used for prediction of discrete value.	2. It is used for prediction of continuous values.
3. Nature of predicted data is unordered.	3. Nature of predicted data is ordered.
4. Here, we measure the accuracy of our classification model to calculate the performance	4. We calculate performance by measuring mean sq. value.

5. Eg. Decision Tree, KNN, logistic Regression

5. Eg. Random forest, Linear Regression

Nearest Neighbour Classifier

→ It is simplest classification algorithm in supervised learning.

→ This method of classifying pattern based on class label of the closest training pattern in the future space.

→ The common algo used for the given neighbour is K-nearest neighbour (KNN) or modified KNN (M-KNN) algo.

→ The accuracy using nearest neighbour classifier is good.

→ It is guaranteed to yield an error rate, not worse than twice of bayer error rate (optimal error rate).

→ There is no training time requ. for this classifier. In other words there is no design time for training the classifier.

→ Every time a test pattern is to be classified, it has to be compared with all the training pattern to find the closest pattern.

→ This classification time (computation time) could be used if training data is large or dimensionality is high.

KNN

- An object is classified by majority of nodes of class of its neighbour.

- The object is assigned to the class most common among its k-nearest neighbour.

- This algorithm may give more correct classification then basis nearest neighbour algo where (k=1).

- The value of 'k' has to be specified by the user & it should be best choice depending on the data.

- larger value of 'k' reduce the effect of noise of the classification.

- The 'k' value can be chosen by using a validation set & chosing the k-value giving best accuracy on that validation set.

* Disadvantage:

Computation time, specially when training data is large.

- Nearest Neighbour majorly use Euclidean distance to find out the nearest neighbour.

KNN

ALGORITHM :-

1. locate the test data

2. Choose the value of k.

3. For each pt. in test data,
   → find the euclidean dist to all training data points.
   → Store the euclidean dist in a list & sort it.
   → Choose the first k pt.
   → Assign a class to the test point based on the majority of classes valid in the chosen pt.

4. END

Pros & Cons of KNN

- Pros :-

1. No assumption about data is useful. for eg. non linear data.
2. Simple algo to explain, interpret & understand
3. High accuracy (relatively)
4. Its accuracy is pretty high but not competitive in comparison to many of supervised learning model.

5. Useful for classification & Regression

- Cons :-

1. Computationally expensive because for every new test point, algo need to calculate euclidean dist to all other pts.

2. High memory requ. Becz. the algo stores all the training data & euclidean dist of each test pt. with every other pt.

3. Prediction stage might be slow

4. KNN is also sensitive to independent features & scale of data.

Naive Algo

→ Naive Based Classification algorithm for binary & multiclass Classification problems, the technique is easiest to understand when describing using binary or categorical i/p values.

→ It is called ~~naive~~ naive based idiot based because the calculation of probablities for each hypothesis are simplified to make their calculation tractable. rather than attempting to calculate the value of each attribute value ($P(d_1, d_2, d_3 | h)$).

→ They are assumed to be conditionally

independent given the target value &
calculated as,

$$P(d1/h) * P(d2/h) \text{ -- and so on.}$$

→ This is very strong assumption i.e. most unlikely in real data that means, attributes do not interact.
Never the less, the approach performs surprisingly well on data where this assumption does not hold.

* Naive ~~bayes~~ BASED representation

· are stored

- It includes,
  - ✓ Class probablity :- The probablities of each class in training data set
  - ✓ The conditional probablity for each value given each class value.
  - ✓ Learning a naive based model from training data is fast because only the probablity of each class & probablity of each class given diff^n i/p value need to be calculated.
  - ✓ Calculate class probablities
  - ✓ Calculate conditional class probablities

Ques.

	OUTLOOK	TEMP	HUMID	WINDY	PLAY
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	F	Yes
3	Sunny	Mild	High	F	Yes
4	Sunny	Cool	Normal	F	Yes
5	Overcast	Cool	Normal	T	Yes
6	Rainy	Mild	High	F	No
7	Rainy	Cool	Normal	F	Yes
8	Sunny	Mild	Normal	F	Yes
9	Rainy	Mild	Normal	T	Yes
10	Overcast	Mild	High	T	Yes
11	Overcast	Hot	Normal	F	Yes
12	Sunny	Mild	High	T	No
13	Sunny.	Cool	Normal	T	No

→ we'll find classes based on each feature.

OUTLOOK :-

$$P(\text{Yes} \mid \text{Rainy}) = \frac{P(\text{Yes} \cap \text{Rainy})}{P(\text{Rainy})} = \frac{2}{5} = \frac{2}{5}$$

$$P(\text{No} \mid \text{Rainy}) = \frac{P(\text{No} \cap \text{Rainy})}{P(\text{Rainy})} = \frac{3}{5}$$

$$P(\text{Yes} \mid \text{Overcast}) = \frac{P(\text{Yes}, \text{Overcast})}{P(\text{Overcast})} = \frac{4}{4} = 1$$

$$P(\text{No} \mid \text{Overcast}) = 0$$

$P(\text{Yes}|\text{Sunny}) = \frac{P(\text{Yes}, \text{Sunny})}{P(\text{Sunny})} = \frac{3}{5}$

$P(\text{No}|\text{Sunny}) = \frac{2}{5}$

TEMP :-

$P(\text{Yes}|\text{Hot}) = \frac{2}{4} = \frac{1}{2}$

$P(\text{No}|\text{Hot}) = \frac{1}{2}$

$P(\text{Yes}|\text{Cool}) = \frac{3}{4}$

$P(\text{No}|\text{Cool}) = \frac{1}{4}$

$P(\text{Yes}|\text{Mild}) = \frac{4}{6} = \frac{2}{3}$

$P(\text{No}|\text{Mild}) = \frac{2}{6} = \frac{1}{3}$

HUMID :-

$P(\text{Yes}|\text{High}) = \frac{3}{7}$ $P(\text{No}|\text{High}) = \frac{4}{7}$

$P(\text{Yes}|\text{Normal}) = \frac{6}{7}$ $P(\text{No}|\text{Normal}) = \frac{1}{7}$

Correct Conditional Probablit

WIND

Sunny

Overc

Rainy

Hot

Mild

Cool

Windy:

$P(\text{Yes} \mid \text{True}) = \frac{3}{6} = \frac{1}{2}$ $\quad$ $P(\text{No} \mid \text{True}) = \frac{3}{6} = \frac{1}{2}$

$P(\text{Yes} \mid \text{False}) = \frac{6}{8} = \frac{3}{4}$ $\quad$ $P(\text{No} \mid \text{False}) = \frac{2}{8} = \frac{1}{4}$

Correct conditional probablities → so that que can be evaluated using Baye's thm.

$P(\text{Yes}) = 9/14$

$P(\text{No}) = 5/14$

We are calculating,

$P(\text{Sunny} \mid \text{Yes}) = \frac{P(\text{Sunny}, \text{Yes})}{P(\text{Yes})}$

	Yes	No	P(Yes)	P(No)
Sunny	3	2	3/9	2/5
Overcast	4	0	4/9	0
Rainy	2	3	2/9	3/5

	Yes	No	P(Yes)	P(No)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5

	Yes	No	P(Yes)	P(No)
Humid (Yes)	3	4	3/9	4/5
Humid (No)	6	1	6/9	1/5

	Yes	No	P(Yes)	P(No)
Windy (Yes)	3	3	3/9	3/5
Windy (No)	6	2	6/9	2/5

$x =$ (Sunny, Cool, Humid High, Windy True)

$$P(Yes|x) = \frac{P(x|Yes).P(Yes)}{P(x)}$$

$$P(Yes|x) = P(x|Yes).\ P(Yes)$$

$$= P(Sunny|Yes).\ P(Cool|Yes).\ P(Humid\ High|Yes).\ P(Windy\ T|Yes).\ P(Yes)$$

$$= \frac{\cancel{3}}{\cancel{9}} \times \frac{\cancel{3}}{\cancel{9}} \times \frac{\cancel{3}}{\cancel{9}} \times \frac{3}{9} \times \frac{\cancel{9}}{14}$$

$$= \frac{1}{9 \times 14} = 0.0079$$

Normal

Page No.
Date: / /

$$P(No|x) = \frac{\cancel{2}}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{\cancel{8}} \times \frac{\cancel{5}}{\cancel{14}\,7}$$

$$= \frac{12}{125 \times 7} = 0.0137$$

∴ Ans will be No

To normalise, we find P(X) i.e. over whole set combining Yes & No.

$$P(X) = \underset{\text{sunny}}{\left(\frac{5}{14}\right)} \times \underset{\text{cool}}{\left(\frac{4}{14}\right)} \times \underset{\text{Humid}}{\left(\frac{7}{14}\right)} \times \underset{\text{windy}}{\left(\frac{6}{14}\right)}$$

$$= (0.02186)$$

Normalised

$$P(Yes|x) = \frac{0.0079}{0.02186} = 0.36139$$

$$P(No|x) = \frac{0.0137}{0.02186} = 0.62671$$

Text Classification

Date: / /

- It is a process of ~~assignment based on~~ assiging tag ~~word~~ or, categories ~~of~~ to text acc. to its content.

- It is one of the fundamental task of NLP with broad appl^n such as sentiment analysis, topic labelling, spam detection, etc.

- (Text classification is the task of) Assigning a set of predefined categories to free text.

- Text classifier can be used to organise structure & categories pretty much anything.

for eg.
Spam Detection
Organise the topics acc. to their category like sports, movie, defence, politics.

Working of Text Classification

- We can do it 2 ways :-

(i) Manual
(ii) Automatic

→ Manual Clasf^n usually provide better quality result but it is very expensive & time consuming.

→ Automatic clasf^n requ. machine learning, NLP (natural language processing) & other technique to automatically classify the text in more effective & faster way.

TEXT CLASSIFICATION

- Rule based
- Machine Learning based
  - Naive Bayes
  - SVM
  - Deep Learning
- Hybrid System

Rule based approach is classified into

→ organise grp by using set of handcrafted linguistic rules. instruct

→ Then rules ~~statistic~~ the system to use ~~semantic~~ statically relevant elements of the text ~~(?)~~ to identify relevant category based on its content.

→ Each rule consist of pattern and a predicted category

Eg. You want to classify news into two grps, sports & politics.

→ They are a combination of manual and

→ They are human comprehensible & improve over time but, it also has some disadvantages for starters, these system requ. deep knowledge of domain.

→ They are also time consuming since, generating the ~~rating~~ rule for a complex system can be quite challenging & usually requ. a lot of analysis & testing which is performed by human experts.

→ Rule based system are also difficult to maintain & do not scale well given that adding new rules can affect the results of pre-existing rules.

Machine Learning

→ Instead of relying on manually crafted rules, text classification in ML, text learns to make clasfn based on past obs.

→ It used pre labeled eg. as training data. A machine learning algo can learn diffn association b/w pieces of text & that a particular o/p/ text/classes.

→ The first step of training a classifier using ML is feature extraction method

modelled

Gath
Dat

Date : / /

to represent features into numerical data in form of vector). The most common approach is "bag of words."

Hybrid System

→ It combines a Bayes classifier trained with ML & a rule based system, which is used to further improve the result.

→ These hybrid system can be easily find by adding specific rules for those conflicting ~~loops~~ text that are not being correctly modelled ~~classified~~ by ~~...~~ Baye's classifier.

Text Classification Workflow

Gathering Data → Explore data → Prepare data → Build, train & Evaluate Model

↓

Tune Hyperparameters

↓

Deploy Model.

→ If you are using a Public API to collect the data then try to understand the limitation

of API before using it.
Eg. Some API set limit on the rate with which you can make queries.
The more training eg. you have is pattern. This helps to generalised the pattern, make sure the no. of samples for every class of feature is not imbalanced i.e. you should have comparable no. of sample per class.

✓ Make sure, your sample covers spaces of possible i/p's not only the common classes.

Explore Data

(1). ✓ Building of Training data is only one part of workflow. Understanding the charcteristics of your data beforehand helps you to build better model.
✓ It will help in acheiving better efficiency & higher accuracy.
✓ ↓ Less data for computing & training. It could also mean requiring resources

(2). ✓ Check the data
(3) ✓ Collect key matrices.
↓
1. no. of samples, no. of classes, no. of samples per class, no. of words per sample,

frequency distribution of words, Distribution of sample length.

4. Choose the Model.
~~Before~~

Preparing Model

→ Before our data can be fetched into a model, it needs to be transformed to a format to a model that can understand.

→ First, the data sample, we have gathered may be in specific order so, we always shuffle the data before doing anything else because, we do not want any info associated with ordering.

→ Machine learning algo take no's as i/p, so we need to convert the text into numerical vectors by two steps.

1. ~~Token~~ Tokenisation
Divide the text into words or smaller subtext which will enable good generalisation of relationship b/w words & label.

2. Vectorisation
Define a numerical major to characterise each text.

Build, train & Evaluate Your Model

→ Building ML Model is all about assembling layers together, data processing building blocks.

→ These layers allow us to specify the sequence of transformation, we want to apply on our ~~model~~. O/P. As our learning algo take a single text i/p and o/p a classification, we can create linear stacks of layers using sequencing model.

Tune Hyperparameters

→ We have to choose, no. of hyper parameters for defining & training a model.

→ We rely on intuitions, eg, and best practice recommendation.

→ Our first choice of hyperparameter values however may not yield the best result.

→ But, it can give us a good starting pt. for training.

→ Every problem is different & ~~unique~~ tuning these hyperparameters will help refine our model to better represent the particular problem.

→ Eg. Some of hyperparameters which can be tuned, are:
1. no. of layers in model
2. no. of units per layer
3. Drop out rate

4. learning rate, etc.

Deploy Your Model

following three things should keep in mind while deploying our model :-

1. Make sure, your production data follows the same distribution as your training & evaluating data.
2. Regularly revaluate by collecting more training data.
3. If your data distribution changes, retrain your model.

Que

Chills	Running Nose	Headache	Fever	Flu
Y	N	Mild	Y	N
Y	Y	No	N	Y
Y	N	Strong	Y	Y
N	Y	Mild	Y	Y
N	N	No	N	N
N	Y	Strong	Y	Y
N	Y	Strong	N	N
Y	Y	Mild	Y	Y

$P(Flu=Y) = \frac{5}{8}$; $P(Flu=N) = \frac{3}{8}$

Chills :-

$F(Chills=Y / Flu=Y) = \frac{3}{5}$; $F(Chills=Y / Flu=N) = \frac{1}{3}$,

$F(Chills=N / Flu=N) = \frac{2}{3}$; $F(Chills=N / Flu=Y) = \frac{2}{5}$

<u>Running Nose:-</u>

$F(RN|Y) = \frac{4}{5}$ $F(RN|N) = \frac{1}{3}$

$F(RN'|Y) = \frac{1}{5}$ $F(RN'|N) = \frac{2}{3}$

<u>Headache:-</u>

$F(Strong|Y) = \frac{2}{5}$ $F(Strong|N) = \frac{1}{3}$

$F(Mild|Y) = \frac{2}{5}$ $F(Mild|N) = \frac{1}{3}$

$F(No|Y) = \frac{1}{5}$ $F(No|N) = \frac{1}{3}$

<u>Fever:-</u>

$F(fever|Y) = \frac{4}{5}$ $P(fever|N) = \frac{1}{3}$

$P(fever'|N) = \frac{2}{3}$ $P(fever'|Y) = \frac{1}{5}$

* → $(x = Yes, No, Mild, No)$ Now, Flu=?

$P(Y|x)$

$= P(x|Y)P(Y)$

$= ~~P(Y|Y)~~$.

$= P(Y|Y) \cdot P(N|Y) \cdot P(Mild|Y) \cdot P(N|Y) \cdot P(\cancel{\ldots}) \cdot P(Y)$

$= \frac{3}{5} \times \frac{1}{5} \times \frac{2}{5} \times \frac{1}{5} \times \frac{5}{8}$

$= 0.006$

$P(N|x)$

$= P(x|N)\, P(N)$

$= \frac{1}{3} \times \frac{2}{3} \times \frac{1}{3} \times \frac{2}{3} \times \frac{3}{8}$

$= 0.018$

$\therefore$ $\boxed{Ans = flu = No}$

Page No.
Date : / /

Naive Bayes → Text Classification

1 → Present each doc by vector of words
→ one attribute per position in doc.

2 → Learning, use training eg. to estimate
$P(+)$, $P(-)$, $P(doc/+)$, $P(doc/-)$

Naive bayes, $P(doc/v_j) = \prod_{i=1}^{length(doc)} P(a_i = w_k / v_j)$

$P(a_i = w_k / v_j)$

where, probablity is the probablity that word in pos the collision i is w_k given v_j.

One more assumption →

$$P(a_i = w_k / v_j) = P(a_m = w_k / v_j) \ \forall i, m$$

You have a set of reviews (a set of doc) f classification,

Doc	TEXT	Class
1	I loved the movie	+
2	I hated the movie	−
3	A great movie, good movie	+
4	poor acting	−
5	Great acting, a good movie	+

10 Unique words.
< I, loved, the movie, hated, a, great, poor, acting, good >

→ Convert a document into features where the attributes are possible words & the values are no. of time. (a word occur in given document)

(I).

	I	loved	the	movie	hated	a	great	poor	acting	good	c
1.	1	1	1	1	0	0	0	0	0	0	+
2.	1	0	1	1	1	0	0	0	0	0	−
3.	0	0	0	2	0	1	1	0	0	1	+
4.	0	0	0	0	0	0	0	1	1	0	−
5.	0	0	0	1	0	1	1	0	1	1	−

(I hated the poor acting)

P/V/Y

$P(+) = \frac{3}{5}$; $P(-) = \frac{2}{5}$

	+	−
I	1	1

P(c/x) = P(x/y) =

$P(+/I) = \frac{1}{3}$; $P(-/I) = \frac{1}{2}$

P(loved/+)

~~P(+/loved)~~ $= \frac{1}{3}$; $P(-/loved) = 0$

$P(+/the) = \frac{1}{3}$; $P(-/the) = \frac{1}{2}$

P(+/movie) = ; P(−/movie) =

TFIDF

↳ to ensure, words like I, a, the ~~does not effect~~ classification.

acting	good	Class
0	0	+
0	0	−
0	1	+
1	0	−
1	1	+

loved :-

$P(\text{Yes}/+) = \frac{1}{3}$ $\quad$ $P(\text{No}/+) = \frac{2}{3}$

$P(\text{Yes}|-) = 0$ $\quad$ $P(\text{No}|-) = 1$

I :-

$P(I|+) = \frac{1}{3}$ $\quad$ $P(I'/+) = \frac{2}{3}$

$P(I|-) = \frac{1}{2}$ $\quad$ $P(I'|-) = \frac{1}{2}$

the :-

$P(\text{the}|+) = \frac{1}{3}$ $\quad$ $P(\text{the}'/+) = \frac{2}{3}$

$P(\text{the}|-) = \frac{1}{2}$ $\quad$ $P(\text{the}'|-) = \frac{1}{2}$

movie:-

$P(\text{movie}|+) = \frac{3}{3} = 1$ $\quad$ $P(\text{movie}'|+) = 0$

$P(\text{movie}|-) = \frac{1}{2}$ $\quad$ $P(\text{movie}'|-) = \frac{1}{2}$

hated :-

$P(\text{hated}|+) = 0$ $\quad$ $P(\text{hated}'/+) = 1$

$P(\text{hated}|-) = \frac{1}{2}$ $\quad$ $P(\text{hated}'|-) = \frac{1}{2}$

a :-

$P(a|+) = \frac{2}{3}$ $\quad$ $P(a'|+) = \frac{1}{3}$

$P(a|-) = 0$ $\quad$ $P(a'|-) = 1$

great:-

$P(great/+) = \frac{2}{3}$ $P(great'/+) = \frac{1}{3}$

$P(great/-) = 0$ $P(great'/-) = 1$

poor:

$P(poor/+) = 0$ $P(poor'/+) = 1$

$P(poor/-) = \frac{1}{2}$ $P(poor'/-) = \frac{1}{2}$

acting:

$P(acting/+) = \frac{1}{3}$ $P(acting'/+) = \frac{2}{3}$

$P(acting/-) = \frac{1}{2}$ $P(acting'/-) = \frac{1}{2}$

good:

$P(good/+) = \frac{2}{3}$ $P(good'/+) = \frac{1}{3}$

$P(good/-) = 0$ $P(good'/-) = 1$

I hated the poor acting

$P(+/x) =$

$P(I/+)\ P(hated/+)\ P(the/+)\ P(poor/+)\ P(acting/+) \cdot P(+)$

$= \frac{1}{3} \times 0 \times \text{———} \times - = 0$

$P(-/x) = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{\cancel{2}} \times \frac{\cancel{2}}{5}$

$= \frac{1}{16 \times 5} = \frac{1}{80} \rightarrow (-)$ class