PATTERN

Page No.
Date

Intro
Feature Extraction
Bayes decision theory
Max likelihood estimation
Prob Density Estimation
Nearest Neighbour
Discrimination
LDA
Unsupervised

Classification
Clustering
Feature
Neural Network
Bayes classification
Decision tree
LDA
SVD
SVN

→ Computers classify images based on features. Features should not be less, not more, it should be balanced.

→ Satellite Image Classification
Satellite images are 36.25 x 36.25 m². Suppose resolution is 512 x 512. So image is very large. Classification of satellite images is very imp (i) For eg, in case of Tsunami, how much area is affected, is given by satellite images (ii) For eg, noting changes at border area

→ eg of image classification
(i) medical purposes
x-ray, blood classification, tumor etc.
(ii) object and face recognisation, voice recognisation

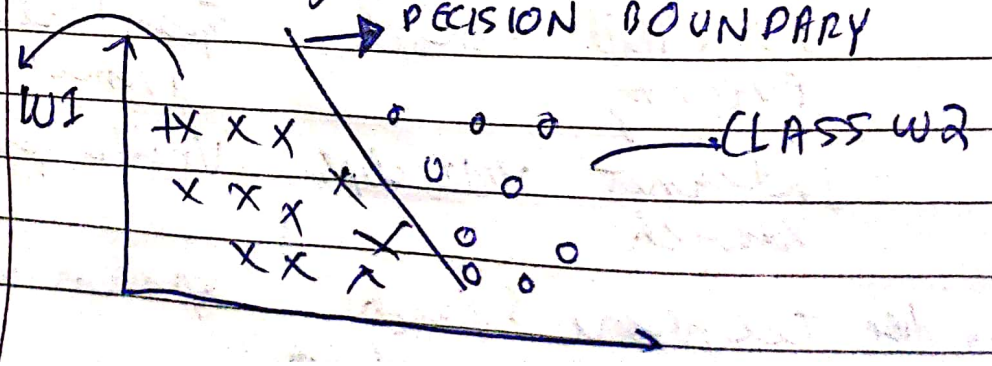
PATTERN DOMAIN	APP	I/P PATTERN	PATTERN CLASSES
(i) Doc Image analysis	OCR	Doc Image	char, words
(ii) Doc classification	Internet search	Text Doc	Semantic categories
	Junk Mail Filtering	email	Spam / Inbox / promotion
(iii) MM Database retrival	Internet search	Video clips	Video games
(iv) Speech recognisation	Telephone assistant	Speech waveform	spoken words

(v) NLP	Info Extraction	Sentences	Parts of speech
(vi) Biometric recognisation	Personal Identification	Face, iris, fingerprint	Authorize user for Access control
(vii) Medical	Comp Aided diagnosis	Microscopic Image	Cancerous / Healthy cell.
(viii) Remote Sensing	Forecasting crop yield	Multispectral Image	Land use categories
(ix) Bio informatics	Sequence Analysis	DNA Sequence	Known types of genes.
(x) Data mining	Searching for meaningful seq	Points in multi-dimensional space	

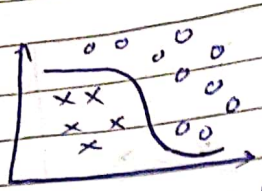
BAYES DECISION PROBLEM

Many of the features, taken together, so, they can describe an object with some degree of accuracy. These are called FEATURE VECTOR. They have to be concatenated in a particular order, and whichever order we concatenate them throughout our problem, that is MODELLING OF PATTERN as well as RECOGNISATION OF PATTERN

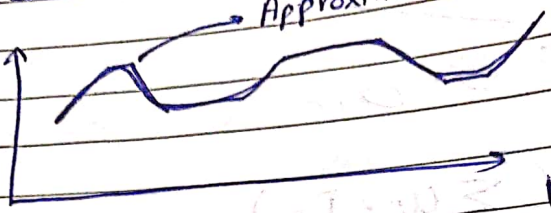
On putting these diff features in a particular order, they are called FEATURE VECTOR



This is LINEAR DECISION BOUNDARY



NON-LINEAR DECISION BOUNDARY



TRAINING PHASE

Training Set

Pre-processing

Removing Noise etc

Feature extraction/selection

FEATURES

Model learning / estimation

MODEL

TEST PHASE

Physical env

Data acquisition/sensing

Pre-processing

Feature extraction

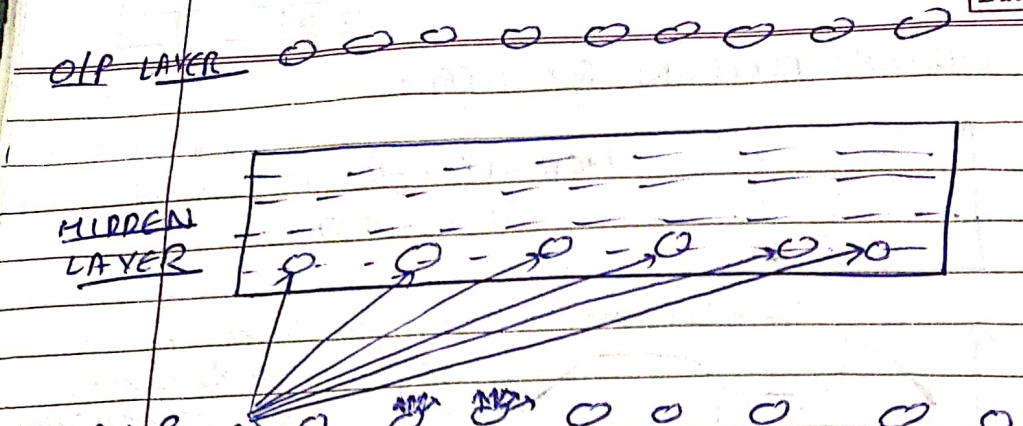
FEATURES

Classification based on Model

Post-processing

DECISION

error finding

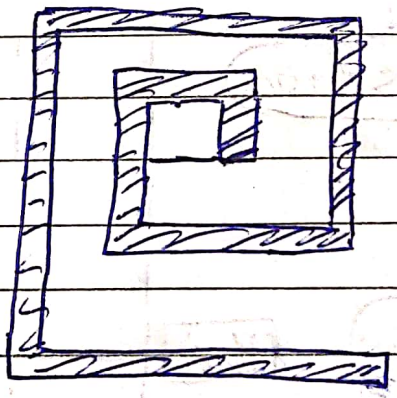


$$O_j = f(\sum w_{ij} I_j)$$

→ OVERLAPPING BOUNDARY.

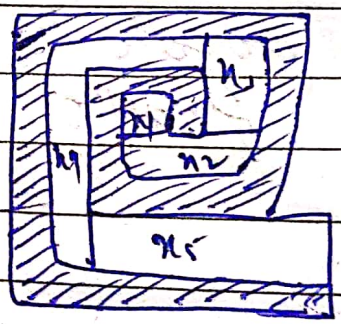
eg → Person sitting of chair

Normal = A
Shaded = B



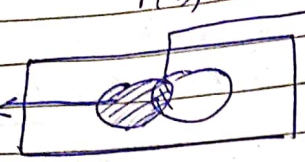
Can not be classified by linear/non-linear boundary

MULTI-LAYER PERCEPTRON PROBLEM



→ CONDITIONAL PROBABILITY :

$P(A|B) = \frac{P(A \cap B)}{P(B)}$ = probability of A when B has already occured



1. % of adult who are man & alcoholic = 2.25. Person given is male find prob of alcoholic.

$P(A|M) = \frac{P(A \cap M)}{P(M)} = \frac{2.25}{100 \times \frac{1}{2}}$

(A: Alcoholic, M: Male)

Assuming no. of male = no. of female

IDEALLY

$= \frac{4.5}{100} = 0.045$

→ BAYES THEOREM

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

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

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

2. What is prob of 2 girls in a couple when given atleast one is a girl.

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

A: atleast one is girl

B: first is girl

~~$P(A \cap B) = 1/4$~~

P(

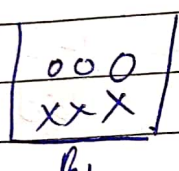
P(2G / atleast 1G)

P(2G) = 1/4 $\quad$ {GG}

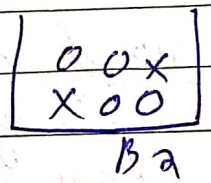
P(1G) = 3/4 $\quad$ {GB, BG, GG} → Always I as if 2G given, atleast 1 will be girl

$$P(2G/1G) = \frac{P(1G/2G)\,P(2G)}{P(1G)} = \frac{1 \times 1/4}{3/4} = \frac{1}{3}$$

3. If I randomly drawn a green ball, Find prob green from first bucket



B_1



B_2

x = green
0 = red

4. $P(G/B_1) = \frac{1}{2}$ $\quad$ $P(G/B_2) = \frac{1}{3}$

$$P(B_1/G) = \frac{P(B_1 \cap G)}{P(G)} = \frac{P(G/B_1)\,P(B_1)}{P(G)}$$

$$= \frac{\frac{1}{2} \times \frac{1}{2}}{\frac{1}{4} + \frac{1}{6}} = \frac{\frac{1}{4}}{\frac{3+2}{12}} = \frac{3}{5}$$

$$P(G) = P(G/B_1)\,P(B_1) + P(G/B_2)\,P(B_2)$$
$$= P(G \cap B_1) + P(G \cap B_2)$$

BAYES DECISION THEORY

(2.1)

(1) CLASSIFICATION BASED ON PRIORI PROBABILITY

w_1 Accept product $P(w_1)$ | w_2 Reject Product $P(w_2)$

* $P(w_1) > P(w_2) \Rightarrow w_1$
  $P(w_1) < P(w_2) \Rightarrow w_2$
  This is called a priori probability

If $P(w_1) = 0.9$ $P(w_2) = 0.1$, then product will always be accepted, and this is not ideal. So, we will add an additional feature x

(2) CLASSIFICATION BASED ON CLASS PDF

* $P(x/w_1)$ $P(x/w_2)$

Class condition PDF

These can be calculated using training set.

Now we know the feature, we need to classify it.

$$P(w_i \cap x) = P(w_i/x)\,P(x)$$

$$= P(x/w_i)\,P(w_i)$$

$$\Rightarrow P(w_i/x).P(x) = P(x/w_i)\,P(w_i)$$

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

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

→ POSTERIOR PROBABILITY

Page No.
Date

⇒ $P(w_1|x) > P(w_2|x) \Rightarrow w_1$

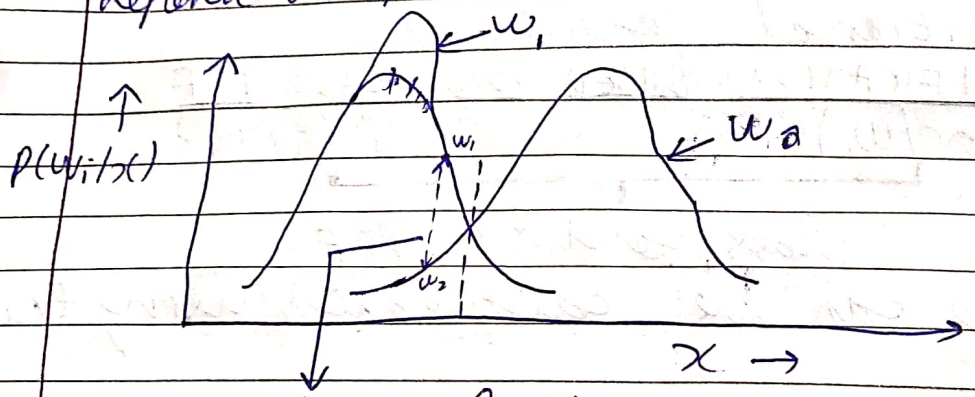
$P(w_1|x) < P(w_2|x) \Rightarrow w_2$

$$\frac{P(x|w_1)\cdot P(w_1)}{P(x)} > \frac{P(x|w_2)\,P(w_2)}{P(x)}$$

$P(x|w_1)$: training set; $P(w_1)$: priori probability; $P(x|w_2)$: training set; $P(w_2)$: priori probability

A POSTERIOR PROB

* if $P(x|w_1) = P(x|w_2)$, then we only depend on priori.



if we select w_1

$P(\text{Error}|x) = P(w_2|x)$

if we select w_2

$P(\text{Error}|x) = P(w_1|x)$

To minimise the error, we select $P(w_1|x)$ if $P(w_1|x) > P(w_2|x)$ and vice versa.

eg → x is colour Intensity. If intensity is low = accept, otherwise reject

AVERAGE ERROR

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

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

(If we minimize $P(\text{error}|x)$ $\forall x$, then avg error is minimised)

DECISION RULE

$P(\text{error}|x) = \min[P(w_1|x), P(w_2|x)]$

and if $P(w_1|x) > P(w_2|x) \rightarrow$ class w_1

else → class w_2

→ $P(x/w_1)$ $P(x/w_2)$

class conditional PDF

$$P(w_1/x) = \frac{P(x/w_1)\,P(w_1)}{P(x)}$$ ← A POSTERIORI PROB

$$P(w_2/x) = \frac{P(x/w_2)\,P(w_2)}{P(x)}$$

$$P(x) = \sum_{i=1}^{2} P(x/w_i)\cdot P(w_i) \quad (2.2)$$

BAYESIAN THEORY FOR CONTINUOUS FEATURES

→ ~~PRIOR PROBABILITY~~

Ques Quality of course: good, ~~fair~~, bad

(ii) Prob Prior: 0.2, 0.4, 0.4

(iii) Class prob $P(x/w_i)$:

	good	fair	bad
Interesting	0.8	0.5	0.10 ~~0.0~~ 0.9
Boring	0.20	0.50	

loss function $P\,d(a_i/w_i)$:

	good	fair	bad
Taking	0	5	10
Not taking	20	5	0

→ ~~GENERATION~~ GENERALIZATION

* 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 (NOT ONLY ACCEPT OR REJECT, INTERMEDIATE ACTIONS)

(3) CLASSIFICATION BASED ON RISK

* Introduce a loss function more general than Prob of error

$c_1 \rightarrow$ states of nature
$w_1, w_2, \ldots, w_c$
$a \rightarrow$ Actions
$\alpha_1, \alpha_2 \ldots \alpha_a$
Loss function $\rightarrow$
$\lambda(\alpha_i / w_j)$ $\rightarrow$ Loss incurred for taking action α_i when true state of nature is w_j
(action, actual state)

$X \rightarrow$ d-dimensional feature vector.

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

$$P(x) = \sum_{j=1}^{c} P(x|w_j)\,P(w_j)$$

ACTION α_i:

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

Risk function / conditional Risk / expected loss

Try all classifying in all possible state w_j

TWO CATEGORY CLASSIFICATION

$\Rightarrow$ we classify it into MIN RISK

eg. w_1 w_2
α_1 α_2
$\lambda(\alpha_i/w_j) = \lambda_{ij}$

$$R(\alpha_1/x) = \lambda(\alpha_1/w_1)\,P(w_1/x) + \lambda(\alpha_1/w_2)\,P(w_2/x)$$
$$= \lambda_{11}\,P(w_1/x) + \lambda_{12}\,P(w_2/x)$$
$$R(\alpha_2/x) = \lambda_{21}P(w_1/x) + \lambda_{22}\,P(w_2/x)$$

IF $R(\alpha_2/x) > R(\alpha_1/x)$ OR
$R(\alpha_2/x) - R(\alpha_1/x) > 0$

$\Rightarrow$ IF $(\lambda_{21} - \lambda_{11})\,P(w_1/x) > (\lambda_{12} - \lambda_{22})\,P(w_2/x)$

ideally $\lambda_{11} = 0$, as it is of class I, and we classified it in class I, so no error

$\therefore \lambda_{21} - \lambda_{11} \geq 0$ and $\lambda_{12} - \lambda_{22} > 0$

$\therefore P(w_1/x) > P(w_2/x) \rightarrow w_1$

SO MIN-RISK CLASS IS CLASSIFIED

(2.3) $\Rightarrow$ MIN ERROR RATE CLASSIFICATION

α_i = Taking state of nature is w_i

$$\lambda(\alpha_i/w_j) = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases}$$ (As no. error)

$$R(\alpha_i/x) = \sum_{j=1}^{c} \lambda(\alpha_i/w_j).\,P(w_j/x)$$
$$= \sum_{i \neq j} P(w_j/x)$$
$\hookrightarrow$ As if $i = j$, $\lambda(\alpha_i/w_j) = 0$
$$= 1 - P(w_i/x)$$

Ideally, we must try to minimize $R(\alpha_i/x)$. In order to do so, we will have to maximize $P(w_i|x)$. So, decision rule: Decide w_i if $P(w_i|x) > P(w_j|x)$

(2.4) DISCRIMINANT FUNC (2.4)

$x \rightarrow$ [$g_1(x)$, $g_2(x)$, ..., $g_c(x)$] $\rightarrow x \in ?$

CLASSIFIER

$g(x) \rightarrow$ DISCRIMINANT FUNC

$w_1, w_2 \dots w_c \rightarrow$ No. of classes

$g_i(x)$: $i = 1, 2, \dots, c$

* $g_i(x) > g_j(x) \quad \forall j \neq i$
$\Rightarrow x \in w_i$

} DISCRIMINANT FUNCTION CLASSIFIER

Discriminant func should be higher. If we classify acc to risk, then risk must be minimum - Thus we use:

Min Risk classifier: (as maximum discriminant will correspond to minimum risk, we use a negation sign)

$g_i(x) = -R(\alpha_i/x)$

$R(\alpha_i/x)$ = Error Rate $\rightarrow \sum_{j \neq i} P(w_j/x) = 1 - P(w_i/x)$

ON BASIS OF CLASS CONDITIONAL PROB

$g_i(x) = P(w_i/x)$ (For min-error rate classification maximum posterior prob = maximum discriminant func)

Applying some monotonically increasing func on $g_i(x)$ for general purpose

① $g_i(x) = \frac{P(x/w_i)\,P(w_i)}{P(x)}$

$\rightarrow P(x) = \sum_{j=1}^{c} P(x/w_j)\,P(w_j)$

~~Taking ln~~

we can remove $P(x)$ as it is cancelled while comparing 2 classes $g_i(x)$ and $g_j(x)$

② $g_i(x) = P(x/w_i)\,P(w_i)$

~~to~~ Taking log ~~and~~ as production is ~~costly~~ calculation/

All ① ② ③ produce same classification, but some can be simpler to understand and compute

③ ~~$\ln(g_i(x))$~~ $= \ln P(x/w_i) + \ln P(w_i)$

TWO-CATEGORY CLASSIFICATION

* If $g_1(x) > g_2(x) \Rightarrow w_1$
$g_1(x) = g_2(x) \Rightarrow$ DECISION BOUNDARY
$g_1(x) < g_2(x) \Rightarrow w_2$

↳ called DICHOTOMIZER

$g(x) = g_1(x) - g_2(x)$

$g(x) = P(w_1/x) - P(w_2/x)$

$= \ln P(x/w_1) + \ln P(w_1)$ → FROM (2)

$= \ln \left(P(x/w_1) \cdot P(w_1) - P(x/w_2) \cdot P(w_2) \right)$

DECISION RULE w_1 if $g(x) > 0$

~~Taking log of both sides~~

$\log g(x) = \log \left(P(x/w_1) \cdot P(w_1) - P(x/w_2) P(w_2) \right)$

$= \log \left(\frac{P(x/w_1) P(w_1)}{P(x/w_2) P(w_2)} \right)$

$= \log \left(\frac{P(x/w_1)}{P(x/w_2)} \cdot \frac{P(w_1)}{P(w_2)} \right)$ USING (3)

$\log(x) = \log \left(\frac{P(x/w_1)}{P(x/w_2)} \right) + \log \left(\frac{P(w_1)}{P(w_2)} \right)$

$\ln g(x) = \ln P(x/w_1) + \ln P(w_1) - \ln P(x/w_2) - \ln P(w_2)$

$= \log \left(\frac{P(x/w_1)}{P(x/w_2)} \right) + \log \frac{P(w_1)}{P(w_2)}$

NAIVE BAYES

Q.

Sno.	color	TYPE ~~Sports/Suv~~	ORIGIN	Stolen
1	R	Sports	~~Imported~~ Domestic	Yes
2	R	Sports	~~Imp~~ "	No
3	R	Sports	~~Imp~~ "	Yes
4	Y	Sports	~~Imp~~ "	No
5	Y	Sports	~~Imp~~ Imported	Yes
6	Y	SUV	~~Imp~~ "	No
7	Y	SUV	~~Imp~~ "	Yes
8	Y	SUV	~~Domestic~~ Domestic	No
9	R	SUV	~~Imp~~ Imported	No
10	R	Sports	Imported	Yes

DATA ↑

Given car Red ∩ Domestic ∩ SUV

tell whether it ~~is Yes~~ is stolen or not

COLOR

ANS → $P(R/Yes) = 3/5$ $\quad P(R/No) = 2/5$

$P(Y/Yes) = 2/5$ $\quad P(Y/No) = 2/5$

TYPE

$P(Sports/Yes) = 4/5$ $\quad P(Sports/No) = 2/5$

$P(SUV/Yes) = 1/5$ $\quad P(SUV/No) = 3/5$

Origin

$P(Domestic/Yes) = 2/5$ $\quad P(Domestic/No) = 3/5$

$P(Imported/Yes) = 3/5$ $\quad P(Imported/No) = 2/5$

~~$P(x/yes)\,P(yes) = P(x/yes)\,P(yes) = P(Red/yes)\,P(SUV/yes)\,P(Domestic/yes)$~~

~~$P(yes/x) = \frac{P(x/yes)\,P(yes)}{P(x)}$~~

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

~~$P(x)$~~ → ignored as $P(Yes) = P(No)$ so cancelled in comparison

$P(Yes/x) = \frac{P(x/yes)\,P(yes)}{P(x)} = P(Red/yes)\,P(SUV/yes)\,P(Domestic/yes)$

ignored as it is compared with NO ← $P(x)$

$P(Yes/x) = \frac{3}{5} \times \frac{1}{5} \times \frac{2}{5} = \frac{6}{125} \times \frac{5}{10} \rightarrow P(yes)$

$P(x/yes) = P(x_1/yes)\,P(x_2/yes)\cdots$

BY NAIVE BAYES ASSUMPTION

$P(Yes/x) = P(yes \cap RED \cap SUV \cap Domestic)$ → can be separated as INDEPENDENT

$P(x)$

$P(No/x) = \frac{2}{5} \times \frac{3}{5} \times \frac{3}{5} = \frac{24/8}{125}$

As $P(Yes/x) < P(No/x)$, it is not stolen.

(2) SELECT optimal course

Quality of course	good	fair	bad
Prob (prior)	0.2	0.4	0.4
class conditionals $Pr(x/w)$	good	fair	bad
Interesting	0.8	0.5	0.1
Boring	0.2	0.5	0.9
loss func $\lambda(a_i/w_j)$	good	fair	bad
Taking the course	0	5	10
Not taking	20	5	0

ANS. $P_r(Interesting) = P(I/good)\,P(good) + P(I/fair)\,P(fair) + P(I/bad)\,P(bad)$

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

$= 0.16 + 0.20 + 0.04 = 0.4$

$P(Boring) = 1 - 0.4 = 0.6$

$P(good/Interesting) = \frac{P(I/g)\,P(g)}{P(I)}$

$= \frac{0.8 \times 0.2}{0.4} = 0.4$

$P(fair/Interesting) = \frac{P(I/f)\,P(f)}{P(I)} = \frac{0.5 \times 0.4}{0.4}$

$= 0.5$

$P(boring/I) = 1 - (0.5 + 0.4) = 1 - 0.9 = 0.1$

$R(\alpha_i/x) = 1 - P(w_i/x)$

$R(\alpha_1|I) = 1 - P(good|I)$
$= 1 - 0.4 = 0.6$
$R(\alpha_2|I) = 1 - P(f|I)$
$= 1 - 0.5 = 0.5$
~~w = class = Taking course, Not Taking~~
~~α = Interesting / Boring~~

~~$R(taking|I) = P(g|I)$~~
x = feature = Interesting / Boring
w = class = good / fail / Boring

$R(taking|I) = P(g|I)\, d(Taking/good) + P(f|I)\, d(Taking/fail) + P(B|I)\, d(Taking/Boring)$
$= 0.4 \times 0 + 0.5 \times 5 + 0.1 \times 10 = 3.5$

$R(not\ taking|I) = 0.4 \times 20 + 0.5 \times 5 + 0.1 \times (0)$
$= 10.5$

Similarly, $R(taking|Boring) + 2.5 =$ and $R(taking|Boring)$

Minimum risk = optimal solution

DISTANCE :

MINKOWSKI DISTANCE :

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

EUCLIDEAN DISTANCE :

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

$h = 2$

MANHATTAN DISTANCE :

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

$h = 1$

Sq - EUCLIDEAN

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

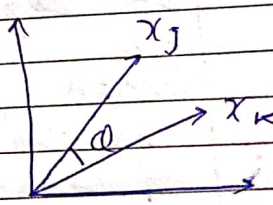
WEIGHTED EUCLIDEAN

Given weightage to each and every feature

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

COSINE SIM

$$\cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}||\vec{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}}$$



$$\sin(\vec{x}, \vec{y}) = \frac{|\vec{x} \times \vec{y}|}{|\vec{x}||\vec{y}|}$$

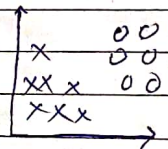
SUPERVISED LEARNING PROBLEM

Can be classified in (i) Regression
(ii) Classification

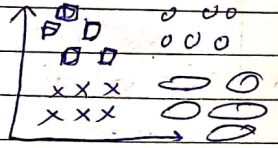
→ **Regression :**
eg → age of person.
A regression problem is when output real or continuous value such as weight of person, salary of person

→ Classification:
problem is when output variable is a category such as person is male or female or diagnosis of tumour is cancer or not. A classical classification model attempts to drawn conclusion from observed values for given one or more inputs, a classification model tries to predict value of one or more outcomes.

In classification data is categorised under different labels acc to some parameters given an input and then labels are predicted from data. It can be demonstrated in form of if-then rules. The classification process deals with problems where data can be divided to binary or discrete value

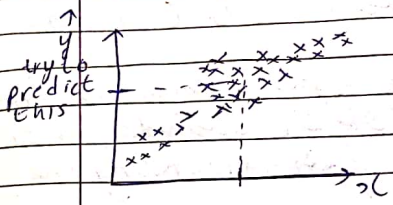


BINARY CLASSIFICATION



MULTI-CLASS CLASSIFICATION

A regression is a process of finding a function for distinguishing data into continuous real value instead of using classes or discrete values. It can also identify then distribution movement depending on historical data because a regression predictive model predict a quantity. Therefore, skill of model must be reported as a endorsed prediction.



CLASSIFICATION	REGRESSION
(i) Mapping func is used for mapping of values to pre-defined classes	(i) Mapping func is used for mapping of values to continuous output.
(ii) is used for prediction of discrete values	(ii) used to predict continuous values
(iii) Nature of predicted data is unordered	(iii) Nature of predicted data is ordered
(iv) we measure accuracy of classification model to calculate performance	(iv) we calculate performance by measuring mean squared error
(v) eg - decision tree, KNN, logistic regression	(v) eg - random forest, linear regression

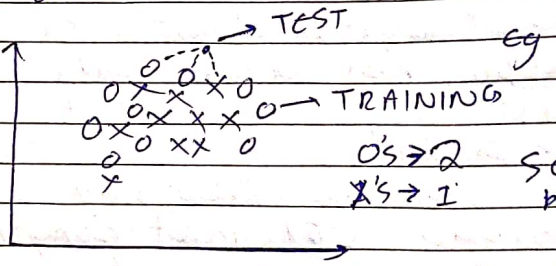
NEAREST-NEIGHBOUR CLASSIFIER

* It is simplest classification algo in supervised learning
* This method of classifying pattern based on class label of closest training patter in feature space
* The common algo used for nearest neighbours is KNN (K-Nearest neighbour) or MKNN (Modified KNN

* The accuracy using nearest neighbour classifier is good. It is guaranteed to yield an error rate no worse than twice of Bayes error rate. (optimal error rate)
* There is no training time required for this classifier. In other words, there is no designing time for training the classifier.
* Every time a test pattern is to be classified it has to be compared with all training pattern to find closest pattern. This classification/computation time can be ~~used~~ high, if training data is large or dimensionality of data is high.

K-NEAREST NEIGHBOUR (KNN)

* An object is classified for majority of votes of class of its neighbours
* The object is assigned to class most common amongst its k nearest neighbours.
* This algo may give more correct classification than basic nearest neighbour where k = 1
* The value of k has to be specified by the user and it should be best choice depending on data
* Larger value of k reduce effect of noise on classification
* The k value can be chosen by using a validation set and choosing the k-value giving best accuracy on validation set.
* Disadvantage of KNN is computation time especially when training data is large.
* Nearest neighbour majorily use Euclidean distance to find nearest neighbour



ALGORITHM:

1. Locate the test data
2. Choose the value of k
3. For each point in test data

→ find the euclidean dist to all training data points

→ store the Euclidean dist in a list & sort it

:

The conditional prob. given each class value

In Naive Bayes model from data is fast because only prob. of each class and prob. given diff input values to be calculated.

class prob.

the conditional class prob.

temp	Humid	windy	play
Hot	High	False	No
Hot	High	True	No
Hot	High	F	Yes
Mild	High	F	Yes
Cool	Normal	F	Yes
Cool	Normal	T	No
Cool	Normal	T	Yes
~~Cool~~ MILD	High	F	No
Cool	Normal	F	Yes
Mild	Normal	F	Yes
Mild	Normal	T	Yes
Mild	High	T	Yes
Hot	Normal	F	Yes
Mild	High	T	No

$P(Yes) = \frac{9}{14}$ $\quad P(No) = \frac{5}{14}$

$P(Sunny/Yes) = \frac{2/14}{9/14} = \frac{2}{9}$ $\quad P(Overcast/Yes) = \frac{4}{9}$

$P(Sunny/Yes) = \frac{3}{9}$ $\quad P(Sunny/No) = \frac{2}{5}$

$P(Rainy/No) = \frac{3}{5}$ $\quad P(Overcast/No) = 0$

$P(Hot/Yes) = \frac{2}{9}$ $\quad P(Hot/No) = \frac{2}{5}$

$P(Mild/Yes) = \frac{4}{9}$ $\quad P(Mild/No) = \frac{2}{5}$

$P(Cool/Yes) = \frac{3}{9}$ $\quad P(Cool/No) = \frac{1}{5}$

$P(High/Yes) = \frac{3}{9}$ $\quad P(High/No) = \frac{4}{5}$

$P(Normal/Yes) = \frac{6}{9}$ $\quad P(Normal/No) = \frac{1}{5}$

$P(False/Yes) = \frac{6}{9}$ $\quad P(False/No) = \frac{2}{5}$

$P(True/Yes) = \frac{3}{9}$ $\quad P(True/No) = \frac{3}{5}$

Ques. Given x = (Sunny, cool, Humid High, Windy True)
find play

Ans. $P(Yes/x) = \frac{P(Sunny \cap cool \cap High \cap True) P(Yes)}{P(x)}$

$= \frac{\frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{9}{14}}{P(x)} = \frac{3^6}{9^4 \times 14 \times P(x)}$

$P(No/x) = \frac{\frac{2}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} \times \frac{5}{14}}{P(x)}$

$= \frac{24}{14 \times 5^3 \times P(x)}$

→ choose first K-points
→ Assign a class to test points based on ~~major~~ majority of classes present in chosen point

PROS AND CONS OF KNN :

PROS :

(i) No assumption about data is useful. For eg, non-linear data (← No assumption about data)
(ii) simple algo to understand and interpret and explain.
(iii) high accuracy (Relatively) : pretty high accuracy but not competative in comparison to many of supervised learning model
(iv) mainly used in classification

CONS :

(i) Computationally expensive because every for new test point, algo need to calculate euclidean distance to all other points.
(ii) High storage because algo stores all training data and eucldean distance of each point with every other point.
(iii) ~~Pre~~ Prediction stage might be slow
(iv) KNN is also sensitive to irrelevant features and scale of data

NAIVE BAYES CLASSIFICATION ALGO FOR BINARY AND MULTI-CLASS CLASSIFICATION PROBLEMS

→ The technique is easiest to understand when describing using binary or categorical input value
→ It is called naive Bayes or idiot Bayes because calculation of probabilities for each hypothesis are simplified to make their calculation tractable ~~rather~~ than
→ Rather than attempting to calculate value of each attribute value ($P(d_1, d_2, d_3 / h)$). They are assumed to be conditionally independent given target value and calculated as

$= P(d_1/h) * P(d_2/h) \ldots$

→ This is very strong assumption that is most unlikely in real data that means attributes do not interact
→ Never the less, the approach perform surprisingly well on data which is not independent

~~A list c~~

NAIVE BAYES REPRESENTATION MODEL

→ It includes
(i) class probabilities : probability of each class in training data set

(ii) conditional prob: The conditional prob for each value given each class value

→ Learning a ~~the~~ Naive Bayes model from training data is fast because only the probability of each class and prob of each class given diff input values needs to be calculated.

→ Calculate class prob.
→ Calculate the conditional class prob.
→

Ques	outlook	temp	Humid	windy	play
1	Rainy	Hot	High	False	No
2	Rainy	Hot	High	True	No
3	overcast	Hot	High	F	Yes
4	Sunny	Mild	High	F	Yes
5	Sunny	Cool	Normal	F	Yes
6	SUNNY	COOL	Normal	T	NO
7	Overcast	Cool	Normal	T	Yes
8	Rainy	MILD	High	F	No
9	Rainy	Cool	Normal	F	Yes
10	Sunny	Mild	Normal	F	Yes
11	Rainy	Mild	Normal	T	Yes
12	overcast	Mild	High	T	Yes
13	overcast	Hot	Normal	F	Yes
14	Sunny	Mild	High	T	No

Ans. $P(Yes) = \frac{9}{14}$ $P(No) = \frac{5}{14}$

$P(Rainy/Yes) = \frac{\frac{2}{14}}{\frac{9}{14}} = \frac{2}{9}$ $P(overcast/Yes) = \frac{4}{9}$

$P(Sunny/Yes) = \frac{3}{9}$ $P(Sunny/No) = \frac{2}{5}$

$P(Rainy/No) = \frac{3}{5}$ $P(overcast/No) = 0$

$P(Hot/Yes) = \frac{2}{9}$ $P(Hot/No) = \frac{2}{5}$

$P(Mild/Yes) = \frac{4}{9}$ $P(Mild/No) = \frac{2}{5}$

$P(Cool/Yes) = \frac{3}{9}$ $P(Cool/No) = \frac{1}{5}$

$P(High/Yes) = \frac{3}{9}$ $P(High/No) = \frac{4}{5}$

$P(Normal/Yes) = \frac{6}{9}$ $P(Normal/No) = \frac{1}{5}$

$P(\text{windy } False/Yes) = \frac{6}{9}$ $P(False/No) = \frac{2}{5}$

$P(True/Yes) = \frac{3}{9}$ $P(True/No) = \frac{3}{5}$

Ques. Given $x = (Sunny, cool, Humid\ \text{High}, High, Windy\ True)$

find play

Ans. $P(Yes/x) = \frac{P(Sunny \cap cool \cap Humid\ High \cap True \cap P(Yes)}{P(x)}$

$= \frac{\frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{9}{14}}{P(x)} = \frac{3^6}{9^4 \times 14 \times P(x)}$

$P(No/x) = \frac{\frac{2}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} \times \frac{5}{14}}{P(x)}$

$= \frac{24}{14 \times 5^3 \times P(x)}$

$$= \frac{P(Yes|x)}{\frac{3^6}{14 \times P(x)}} - \frac{P(No|x)}{\frac{24}{14 \times 5^3 \times P(x)}}$$

$$= \frac{1}{14 \times 9(P(x))} - \frac{12}{7 \times 125 P(x)} = \frac{1}{P(x)}\left(\frac{1}{9 \times 14} - \frac{12}{875}\right)$$

$$= \frac{875 - 1512}{4 \times 875 \times P(x)} < 0$$

$\therefore P(Yes|x) > P(No|x)$

$\rightarrow P(Yes|x) < P(No|x)$

$\therefore$ ANS = NO

If we find $P(x)$

$P(x) = P(\text{Sunny} \cap \text{cool} \cap \text{High} \cap \text{True})$

$= \frac{5}{14} \times P(\text{Sunny})\, P(\text{cool})\, P(\text{High})\, P(\text{True})$

$= \frac{5}{14} \times \frac{4}{14} \times \frac{7}{14} \times \frac{6}{14}$

$= 0.02186$

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

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

$\therefore$ values are now NORMALISED

We find $P(x)$ for normalisation of answer

TEXT CLASSIFICATION

→ It is a process of assigning tags or categories to text according to its content

→ It is one of the fundamental task of NLP with broad application such as sentiment analysis, topic labelling, spam detection etc

→ Text classification is the task of assigning a set of predefined categories to ~~free~~ free text.

→ Text classifier can be used to organise structure & categories pretty much anything for eg:

(i) Spam Detection:

(ii) organise the topics according to their category like sports, movie, defence, politics.

Working on Text Classification

We can do it in 2 ways:

(i) Manual

(ii) Automatic

→ MANUAL CLASSIFICATION usually provide better quality result but it is very expensive and time consuming

→ AUTOMATIC CLASSIFICATION require machine learning NLP (natural language processing) and other technique to automatically classify the text in more effective and faster way.

TEXT — CLASSIFICATION

- RULE BASED
- MACHINE LEARNING BASED
  - Naive Bayes
  - SVM
  - DEEP LEARNING
- HYBRID SYSTEM

RULE BASED APPROACH is classified into:

→ organised group by using set of hand-crafted, linguistic rules

→ These rules the system to use semantically 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 groups; sports and politics.

→ They also a combination of manual and automatic classification

→ They are human comprehensable and improve over time but, it also has some disadvantages for starters, these system require deep knowledge of domain

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

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

MACHINE-LEARNING

→ Instead of relying on normally crafted rules, text classification is ML based in which Text learns to make classification based on past observation

→ It uses pre-labeled (eg. training) data. A machine learning algo can learn different association b/w pieces of text and that a particular text/ class.

→ The first steps of training a classifier using ML is feature extraction (method to represent features into numerical data in form of a vector). The most common approach is "bag of words".

HYBRID SYSTEM

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

→ These hybrid system can be easily find by adding specific rules for these conflicting text that are not being correctly modelled by Bayes classifier.

TEXT-CLASSIFICATION WORK-FLOW

Gathering data → Exploring Data → Prepare data → Build, train and evaluate Model → Tune Hyper parameters → Deploy Model

GATHERING DATA

→ 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 limitation on the rate with which you can make queries.

→ Make sure that the no. of samples for every class and features is not imbalanced, i.e you should have comparable no. of samples per class

→ Make sure, your sample covers spaces of possible inputs not only the common classes.

EXPLORE DATA

→ Building and training data is only one part of workflow. Understanding the characteristics of your data beforehand helps you to build better model.

→ It will help in achieving better efficiency and higher accuracy.

→ Less data for computing and training

Check the data
Collect key matrices
↳ no. of samples, no. of classes, no. of samples per class, no. of words per sample, frequency distribution of words, distribution of sample length

Choose the Model.

PREPARE DATA

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

→ First, 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

→ Second, machine learning algos take numbers as input. So, we need to convert text to numerical vectors by 2 steps:

(i) TOKENISATION:
Divide text into words or smaller sub-text which will enable good generalisation relationship b/w text and numbers

(ii) VECTORIZATION:
Define a numerical measure to characterise these texts

BUILD, TRAIN AND EVALUATE MODEL

→ Building ML model is all about assembling layers together data-processing building blocks. These layers allow us to specify sequence of transformation we want to perform on our input. As our learning algo takes a single text input and

output a classification. we can create linear stacks of layers using sequencial model.

TUNE HYPER PARAMETERS

→ we have to choose number of hyper parameters for defining and training model.

→ we rely on intuitions, examples and best practice recommendation.

→ our first choice of hyper-parameter value however may not yield best result. But it can give us a good starting point for training.

→ Every problem is different and tuning the hyper-parameters will help refine our model to better represent the particular problem

→ Eg. some of hyper parameters are which can be tuned

1. No. of layers in model
2. No. of units per layer
3. Drop-out rate
4. Learning Rate

DEPLOY YOUR MODEL

→ Following key things should be kept in mind while deploying our model:

(i) Make sure your production data follows same distributions as your training and evaluation data

(ii) regularly re-evaluate by collecting more training data

(iii) if your data distribution changes. re-train your model.

QUES.	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

$x = [Y \quad N \quad Mild \quad N]$ flu?

CHILLS: $P(Yes/Yes) = \frac{3}{5}$ $\quad P(Yes/No) = \frac{1}{3}$

RUNNING NOSE: $P(No/Yes) = \frac{1}{5}$ $\quad P(No/No) = \frac{2}{3}$

HEADACHE: $P(Mild/Yes) = \frac{2}{5}$ $\quad P(Mild/No) = \frac{1}{3}$

FEVER: $P(No/Yes) = \frac{1}{5}$ $\quad P(No/No) = \frac{2}{3}$

$$P(Yes/x) = \frac{3}{5} \times \frac{1}{5} \times \frac{2}{5} \times \frac{1}{5} \times \frac{5}{8} = \frac{6}{125 \times 8}$$

$$P(No/x) = \frac{1}{3} \times \frac{2}{3} \times \frac{1}{3} \times \frac{2}{3} \times \frac{3}{8} = \frac{4}{27 \times 8}$$

$P(Yes|x) = \frac{e^3}{125 \times 84}$

$P(No|x) = \frac{+2}{27 \times 84}$

$P(Yes|x) = P(No|x)$

$= \frac{1}{4}\left(\frac{3}{125} - \frac{2}{27}\right) < 0$

$\therefore P(No|x) > P(Yes|x)$

$\therefore$ NO FLU

NAIVE BAYES → TEXT CLASSIFICATION

1. Present each document by vector of words → one attribute per position in document (doc)
2. Learning, use training eg. to estimate $P(+), P(-), P(doc|+), P(doc|-)$

Naive bayes,

$$P(doc|v_j) = \prod_{i=1}^{length(doc)} P(a_j = w_k | v_j)$$

$$= P(a_i = w_k | v_j)$$

↑

probability $P(a_i = w_k | v_j)$ is that word in collision (i) is w_k given v_j.

QUES.	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	+

Test data = "I hated the poor acting"

ANS

(i) convert text to features where no of distinct words in features and there frequency in value in feature vector

class	I	loved	the	movie	hated	a	great	good	poor	acting
+	1	1	1	1	0	0	0	0	0	0
−	1	0	1	1	1	0	0	0	0	0
+	0	0	0	2	0	1	1	1	0	0
−	0	0	0	0	0	0	0	0	1	1
+	0	0	0	1	0	1	1	1	0	1

(ii) $P(+) = \frac{3}{5}$ $\quad P(-) = \frac{2}{5}$ PRIORI PROB

~~(iii) $P(I|+) =$~~

(iii) CLASS CONDITIONAL PROBABILITY

$$P(w_k | x_i) = \frac{n_k + 1}{|voc| + |n|}$$

word ↵ ↓ class

where n_k = number of times w_k occurs in class x_i

n = total words in class w_i x_i
|vocl = number of distinct words in content

$P(I|+) = \frac{1+1}{10+14} = \frac{2}{24}$ $P(I|-) = \frac{1+1}{10+6} = \frac{2}{16}$

$P(\text{loved}|+) = \frac{2}{24}$ $P(\text{loved}|-) = \frac{1}{16}$

$P(\text{the}|+) = \frac{2}{24}$ $P(\text{the}|-) = \frac{2}{16}$

$P(\text{movie}|+) = \frac{5}{24}$ $P(\text{movie}|-) = \frac{2}{16}$

$P(\text{hated}|+) = \frac{1}{24}$ $P(\text{hated}|-) = \frac{2}{16}$

$P(\text{a}|+) = \frac{3}{24}$ $P(\text{a}|-) = \frac{1}{16}$

$P(\text{great}|+) = \frac{3}{24}$ $P(\text{great}|-) = \frac{1}{16}$

$P(\text{good}|+) = \frac{3}{24}$ $P(\text{good}|-) = \frac{1}{16}$

$P(\text{poor}|+) = \frac{1}{24}$ $P(\text{poor}|-) = \frac{2}{16}$

$P(\text{acting}|+) = \frac{2}{24}$ $P(\text{acting}|-) = \frac{2}{16}$

(iv) CALCULATION OF POSTERIORI BASED ON NAIVE BAYES ASSUMPTION

$P(+|\text{text}) = \frac{P(\text{text}|+)\,P(+)}{P(\text{text})}$

↳ Ignoring

$= P(I|+)\,P(\text{hated}|+)\,P(\text{the}|+)\,P(\text{poor}|+)\,P(\text{acting}|+)\,P(+)$

$= \frac{2}{24} \times \frac{1}{24} \times \frac{2}{24} \times \frac{1}{24} \times \frac{2}{24} \times \frac{3}{5}$

$= \frac{24}{24^5 \times 5} = \frac{1}{5 \times 24^4}$

$P(-|\text{text}) = \frac{2}{16} \times \frac{2}{16} \times \frac{2}{16} \times \frac{2}{16} \times \frac{2}{16} \times \frac{2}{5}$

$= \frac{1}{5 \times 16^3 \times 4}$

$P(-|\text{text}) > P(+|\text{text})$

∴ Text belongs to negative class

SYLABBUS

- → Basic structure of pattern
- → Naive Bayes ✓
- → Bayesian classification ✓
- → Intro to bayesian decision theory for 2 category classification, continuous case ✓
- → Estimation of post-priori prob ✓
- → Conditional risk (R) ✓
- → Min-risk classification ✓
- → Min-error classification ✓
- → Text classification ~~the~~ ✓
- → Text classification through Naive Bayes ✓
- → Maximum likelihood estimation
- → KNN ✓ → Distance measure ✓
- → PDF ✓

Page No.
Date

QUES.1.

TEXT	TAG
A great game	Sports
The election was over	Not sports
very clean match	Sports
A clean but forgettable game	Sports
It was a close election	Not sports

TEST Text:
~~A VERY CLOSE GAME~~

QUES2.

TEXT	CLASS
Chinese Beijing Chinese	C
Chinese Chinese Sanghai	C
Chinese Macao	C
Tokyo Japan Chinese	J

TEST TEXT:
Chinese Chinese Chinese Tokyo Japan

ANS.1.

A	great	game	The	election	was	over	very	clean	match	but	forgettable	It	close	close
1	1	1	0	0	0	0	0	0	0	0	0	0	0	S
0	0	0	1	1	1	1	0	0	0	0	0	0	0	NS
0	0	0	0	0	0	0	1	1	1	0	0	0	0	S
1	0	1	0	0	0	0	0	1	0	1	1	0	0	S
1	0	0	0	1	1	0	0	0	0	0	0	1	1	NS

$P(A|S) = \frac{2+1}{14+11} = \frac{3}{25}$ $\quad P(A|NS) = \frac{1+1}{14+9} = \frac{2}{23}$

$P(very|S) = \frac{1+1}{14+11} = \frac{2}{25}$ $\quad P(very|NS) = \frac{1}{23}$

$P(close|S) = \frac{0+1}{14+11} = \frac{1}{25}$ $\quad P(close|NS) = \frac{2}{23}$

$P(game|S) = \frac{2+1}{14+11} = \frac{3}{25}$ $\quad P(game|NS) = \frac{1}{23}$

Page No.
Date

$P(S) = \frac{3}{5}$ $\quad P(NS) = \frac{2}{5}$

$P(Text|S) = \frac{3}{5} \times \frac{3}{25} \times \frac{2}{25} \times \frac{1}{25} \times \frac{3}{25}$

$= \frac{54}{5 \times 25^4}$

$P(Text|NS) = \frac{2}{5} \times \frac{2}{23} \times \frac{1}{23} \times \frac{2}{23} \times \frac{1}{23}$

$= \frac{8}{5 \times 23^4}$

∴ Sports Text
↳ as $P(Text|S) > P(Text|NS)$

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

Page No.
Date

NS. 2

Chinese	Beijing	Sanghai	Macao	Tokyo	Japan	class
2	1	0	0	0	0	C
2	0	1	0	0	0	C
1	0	0	1	0	0	C
1	0	0	0	1	1	j

$P(\text{Chinese}|C) = \frac{5+1}{8+6} = \frac{6}{14}$

$P(\text{Tokyo}|C) = \frac{0+1}{8+6} = \frac{1}{14}$

$P(\text{Japan}|C) = \frac{0+1}{8+6} = \frac{1}{14}$

$P(\text{Chinese}|j) = \frac{1+1}{3+6} = \frac{2}{9}$

$P(\text{Tokyo}|j) = \frac{1+1}{9} = \frac{2}{9}$

$P(\text{Japan}|j) = \frac{2}{9}$

$P(\text{Test}|C) = \frac{3}{4} \times \left(\frac{6}{14}\right)^3 \times \frac{1}{14} \times \frac{1}{14} = \frac{18}{4 \times 14^5} = \frac{0.00161}{0.000301122}$

$P(\text{Test}|j) = \frac{1}{4} \times \left(\frac{2}{9}\right)^3 = \frac{8.2}{4 \times 9^5} = \frac{0.0027}{0.000133}$

~~Chinese~~ ~~Japanese~~

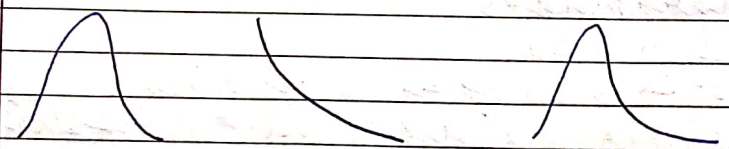
∴ Chinese.

Page No.
Date

MAXIMUM ~~LIKEHO~~ LIKELIHOOD ESTIMATION

→ Is a method of statistical model for given observation. ~~The method obtains~~ the parameter estimates by finding parameter value that maximise the likelihood func.

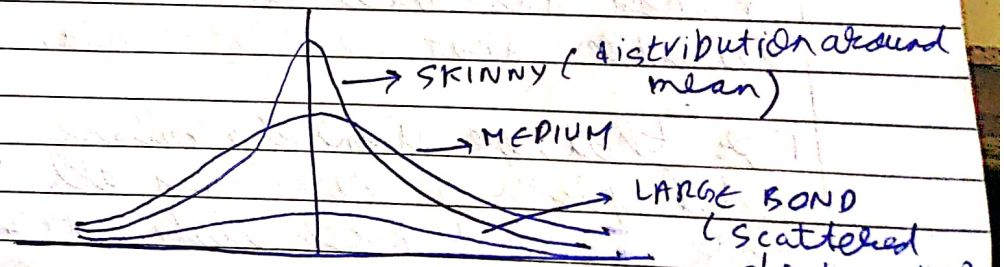
→ It is method that determines values for parameter of a model The parameter value are found s.t. they maximise the likelihood that process described by the model produce the data that ~~was~~ was actually obtained



NORMALLY DISTRIBUTED

(i) We expect most of the measurements to be close to mean

(ii) We expect measurement to be relatably symmetric about mean



(fig)

So for any given/observed data, first we decide which model we think best describes process of generating data. In this example we assume data generation process can be adequately described by gaussian (Normal) distribution.

Visual inspection of figure suggest that gaussian distribution is most suitable because most of 70 points are clustered in middle with few points scattered in left and right.

(fig) ↑

We know gaussian distribution has 2 parameters
(i) $\mu \rightarrow$ mean
(ii) $\sigma \rightarrow$ Standard deviation
Different values of μ, σ result in diff. curves.

Now, we want to calculate which curve is most reasonable for creating data points that we are observing.

Maximum likelihood estimation is a method that will find mean and standard deviation that best fit data.

For that, we want to calculate is the total prob of observing all the data i.e. joint prob distribution of all points which requires calculation of conditional prob. So we make assumption:
(i) Each data point is generated independently of others for easy computation

The prob density of observing a single data point x that is generated from gaussian distribution is given by :-

$$P(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

* We can figure out the values of mean & std. deviation that results in giving the maximum value of above expression by using calculus (maxima of funcs).

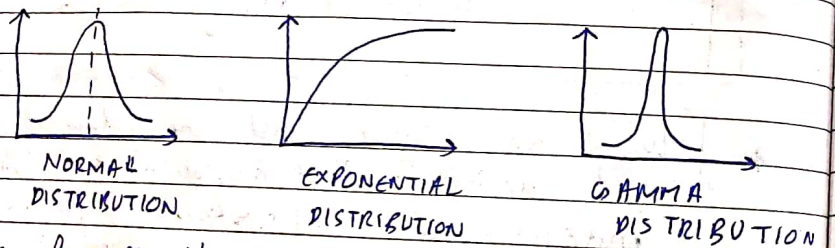
$e^{-\frac{(x-\mu)^2}{2\sigma^2}} \cdot \frac{2(x-\mu)}{2\sigma^2}$

MAXIMUM LIKELIHOOD ESTIMATION

→ Maximum likelihood estimation is a process of finding the curve and the parameters of the curve that best matches the data set given to us.

→ We try to find the parameters of the curve in such a way that it maximises the maximum likelihood function.

→ The curve we choose is based on data points available to us.

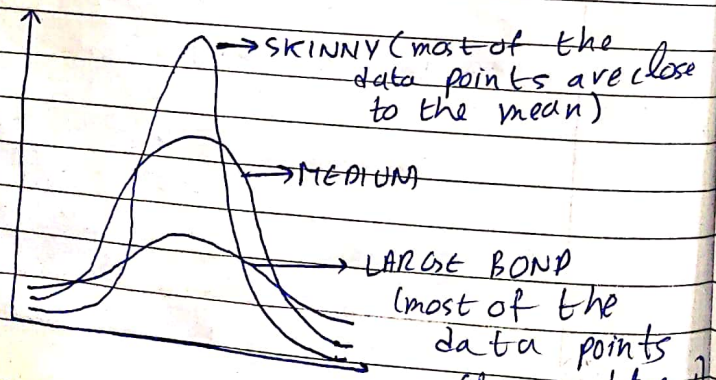


→ We choose the gaussian distribution when

(i) data set is symmetric about mean

(ii) data is close to the mean

→



MAXIMUM LIKELIHOOD ESTIMATION FOR GAUSSIAN DISTRIBUTION

(a) Gaussian distribution has 2 parameters

μ → mean

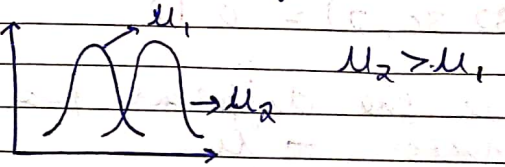
σ → standard deviation.

Based on μ, σ, shape of curve changes

EFFECT OF μ

$\mu \uparrow$, graph shifts right

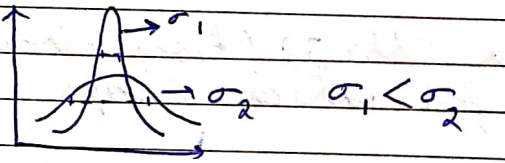
$\mu \downarrow$, graph shifts left.



EFFECT OF σ

$\sigma \uparrow$, graph is broader

$\sigma \downarrow$, graph is narrower about mean



(b) Maximum likelihood involves calculation of μ, σ, that maximises the probability density function.

$$P(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$

For eg:

(i) Let us assume data point $x = 32$

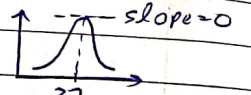
(ii) Fixing $\sigma = 2$, and finding μ for which slope $= 0$, or $\frac{df(x)}{dx} = 0$

$$P(32, 28, 2) = \frac{1}{\sqrt{2\pi(4)}} \exp\left(\frac{-(32-28)^2}{2 \times 4}\right)$$

$$= 0.03$$

$P(32, 30, 2) = 0.12$

(ii) Plot the different points with
x-axis → μ
y-axis ⟹ $P(32, \mu, 2)$



The point with slope $= 0$ is the actual μ required.

(iii) Fix $\mu = 32$ (from above) and finding σ for which slope $= 0$

(2) If more than one data points are given, say x_1, x_2 is given

$P(x_1, \mu, \sigma) = L_1$
$P(x_2, \mu, \sigma) = L_2$

$L = L_1 \times L_2$

Maximum likelihood estimation assumes that measurements of x_1 and x_2 are independent.

$$L(x_1 = 32, x_2 = 34; 28; 2)$$

$$= \frac{1}{\sqrt{2\pi(4)}} \exp\left(\frac{-(32-28)^2}{2(4)}\right) \frac{1}{\sqrt{2\pi(4)}} \exp\left(\frac{-(34-28)^2}{2(4)}\right)$$

(3) GENERAL CASE OF n points in data set

$L = L_1 \times L_2 \times \dots \times L_n$

PROBLEM: we want to maximise derivative of L which is computationally expensive

SOLUTION: take log of above log is a monotonically ↑ function, so we can take log of above

$\log L = \log L_1 + \log L_2 + \dots + \log L_n$

FINDING $\log L_n$

$$\log L_n = \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x_n-\mu)^2}{2\sigma^2}\right)\right)$$

$$= \log \frac{1}{\sqrt{2\pi}} + \log \frac{1}{\sqrt{\sigma^2}} + \log_e e^{\frac{-(x_n-\mu)^2}{2\sigma^2}}$$

$\log(L_n)$

$= \frac{-1}{2}\log 2\pi - n\log\sigma - \frac{(x_n-\mu)^2}{2\sigma^2}\log_e e$

$\log(L_n) = \frac{-1}{2}\log(2\pi) - \log\sigma - \frac{(x_n-\mu)^2}{2\sigma^2}$

USING THIS :

$\# \log L) = \frac{-n}{2}\log 2\pi - n\log\sigma - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_{n,i}-\mu)^2$

(a) Fixing σ = const and ~~maximising L~~ finding μ for which slope = 0

Differentiating wrt μ

$\frac{d(\log L)}{d\mu} = 0 + 0 + \frac{-1}{2\sigma^2}\Sigma(2(x_i-\mu)(-1))$

$= \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i-\mu)$

Equating to 0

$\rightarrow \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i-\mu) = 0$

$x_1 + x_2 + \dots + x_n - n\mu = 0$

$\mu = \frac{x_1+x_2+\dots+x_n}{n}$

(b) fixing $\mu = \frac{x_1+x_2+\dots+x_n}{n}$, and finding σ for which slope = 0

Differentiating wrt σ

$\rightarrow 0 - \frac{n}{\sigma} - \sum_{i=1}^{n}(x_i-\mu)^2 \cdot \frac{(-2)}{\sigma^3} \times \frac{1}{2}$

$= \frac{-n}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{n}(x_i-\mu)^2$

Equating to 0

$\frac{n}{\sigma} = \frac{1}{\sigma^3}\sum_{i=1}^{n}(x_i-\mu)^2$

$\sigma^2 = \frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n}$

$\sigma = \sqrt{\frac{(x_1-\mu)^2 + (x_2-\mu)^2 + \dots + (x_n-\mu)^2}{n}}$

→ This is used in text classification with m = |vocabl $\pi_{tail} = \frac{1}{|vocab|}$

→ This is done so that Prob ≠ 0

(1) M-ESTIMATE

$$P_{tail} = \frac{N_{tail} + m\,\pi_{tail}}{N_{all} + m}$$

N_{tail} → occurences of tail; π_{tail} → prior expectation; N_{all} → total outcomes; m → confidence

→ m-estimate helps us to improve probability analysis for less samples

→ If $N_{all} = N_{tail} = 0$ → degenerates to prior expectation

(2)

Toss No.	1	2	3	4	5
Outcome	tail	tail	head	tail	head

$\pi_{tail} = 0.5$ (i) $m = 2$ (ii) $m = 100$ (iii) $m = 1$

Ans. (i) m-estimate

Toss No.	1	2	3	4	5
N_{tail}	1.0	2.0	2	3	3
m-estimate	$\frac{1+1}{1+2}$	$\frac{2+1}{2+2}$	$\frac{2+1}{3+2}$	$\frac{3+1}{4+2}$	$\frac{3+1}{5+2}$
	$= \frac{2}{3}$	$= \frac{3}{4}$	$= \frac{3}{5}$	$= \frac{4}{6}$	$= \frac{4}{7}$

(ii)

Toss No.	1	2	3	4	5
N_{tail}	1	2	2	3	3
m-estimate	$\frac{51}{101}$	$\frac{52}{102}$	$\frac{52}{103}$	$\frac{53}{104}$	$\frac{53}{105}$

(iii)

Toss No	1	2	3	4	5
N_{tail}	1	2	2	3	3
m-estimate	$\frac{3}{4}$	$\frac{5}{6}$	$\frac{5}{8}$	$\frac{7}{10}$	$\frac{7}{12}$

(2) PDF (PROBABILITY DISTRIBUTION FUNCTION) (FOR CONTINUOUS FEATURES)

→ Standard Gaussian PDF

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

→ In case of multiple features

$$P(x) = \frac{1}{(2\pi)^{n/2}\sigma^n} \sum_{i=1}^{n} \exp\left(-\frac{(x_i-\mu_i)^2}{2\sigma^2}\right)$$

→

	Attr 1	Attr 2	Attr 3	Class	$\sigma = 1$
ex 1	3.1	2.1	2.3	+	
ex 2	4.2	6.2	7.6	+	
ex 3	7.8	1.3	0.5	+	
ex 4	2.3	5.2	2.4	−	
ex 5	6.4	3.2	4.3	−	
ex 6	1.3	5.8	3.3	−	

$$P(x_1 \mid +) = \frac{1}{(2\pi)^{3/2}} \left[\exp\left(-\frac{(x-3.1)^2}{2}\right) + \exp\left(-\frac{(x-4.2)^2}{2}\right) + \exp\left(-\frac{(x-7.8)^2}{2}\right)\right]$$

$$P(x_1 \mid -) = \frac{1}{(2\pi)^{3/2}} \left[\exp\left(-\frac{(x-2.3)^2}{2}\right) + \exp\left(-\frac{(x-6.4)^2}{2}\right) + \exp\left(-\frac{(x-1.3)^2}{2}\right)\right]$$

$$P(x_2|+) = \frac{1}{2\pi^{3/2}}\left[\exp\left(-\frac{(x-2.1)^2}{2}\right) + \exp\left(-\frac{(x-6.2)^2}{2}\right) + \exp\left(-\frac{(x-1.3)^2}{2}\right)\right]$$

$$P(x_2|-) = \frac{1}{2\pi^{3/2}}\left[\exp\left(-\frac{(x-5.2)^2}{2}\right) + \exp\left(-\frac{(x-3.2)^2}{2}\right) + \exp\left(-\frac{(x-5.8)^2}{2}\right)\right]$$

$$P(x_3|+) = \frac{1}{2\pi^{3/2}}\left[\exp\left(-\frac{(x-2.3)^2}{2}\right) + \exp\left(-\frac{(x-7.6)^2}{2}\right) + \exp\left(-\frac{(x-6.5)^2}{2}\right)\right]$$

$$P(x_3|-) = \frac{1}{2\pi^{3/2}}\left[\exp\left(-\frac{(x-2.4)^2}{2}\right) + \exp\left(-\frac{(x-4.3)^2}{2}\right) + \exp\left(-\frac{(x-3.3)^2}{2}\right)\right]$$

→ TESTING VARIABLE : x (9, 2.5, 3.2)

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

$$= P(x_1|+)\,P(x_2|+)\,P(x_3|+)\,P(+)$$

$$= \frac{1}{2\pi^{3/2}}\left[\exp\left(-\frac{(9-3.1)^2}{2}\right) + \exp\left(-\frac{(9-4.2)^2}{2}\right) + \exp\left(-\frac{(9-7.8)^2}{2}\right)\right] \times$$

$$\frac{1}{2\pi^{3/2}}\left[\exp\left(-\frac{(2.5-2.1)^2}{2}\right) + \exp\left(-\frac{(2.5-6.2)^2}{2}\right) + \exp\left(-\frac{(2.5-1.3)^2}{2}\right)\right] \times$$

$$\cdots \times \frac{3}{6}$$

$P(-|x)$ = similarly

→ BASIC-STRUCTURE OF PATTERN RECOGNISATION

(a) PATTERN

* an object, process or event that is given a name
* an image is not a pattern till it is given a name

(b) PATTERN RECOGNISATION

* The process by which machines can
  (i) observe the environment
  (ii) identify the patterns of interest
  (iii) classify the pattern

(c) CLASSES

Set of pattern having common attributes and common source/origin in most cases

(d) FEATURE

The attributes / values that are common in objects of same class, and different in pattern of different classes is called feature

→ DATA-ACQUISTION AND SENSING

The process of acquiring and extracting the relevant and useful data from the environment

eg : extracting address from envelop

$$P(x_2|+) = \frac{1}{2\pi^{3/2}}\left[\exp\left(\frac{-(x-2.1)^2}{2}\right) + \exp\left(\frac{-(x-6.2)^2}{2}\right) + \exp\left(\frac{-(x-1.3)^2}{2}\right)\right]$$

$$P(x_2|-) = \frac{1}{2\pi^{3/2}}\left[\exp\left(\frac{-(x-5.2)^2}{2}\right) + \exp\left(\frac{-(x-3.2)^2}{2}\right) + \exp\left(\frac{-(x-5.8)^2}{2}\right)\right]$$

$$P(x_3|+) = \frac{1}{2\pi^{3/2}}\left[\exp\left(\frac{-(x-2.3)^2}{2}\right) + \exp\left(\frac{-(x-7.6)^2}{2}\right) + \exp\left(\frac{-(x-6.5)^2}{2}\right)\right]$$

$$P(x_3|-) = \frac{1}{2\pi^{3/2}}\left[\exp\left(\frac{-(x-2.4)^2}{2}\right) + \exp\left(\frac{-(x-4.3)^2}{2}\right) + \exp\left(\frac{-(x-3.3)^2}{2}\right)\right]$$

→ TESTING VARIABLE : x (9, 2.5, 3.2)

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

$$= P(x_1|+)\,P(x_2|+)\,P(x_3|+)\,P(+)$$

$$= \frac{1}{2\pi^{3/2}}\left[\exp\left(\frac{-(9-3.1)^2}{2}\right) + \exp\left(\frac{-(9-4.2)^2}{2}\right) + \exp\left(\frac{-(9-7.8)^2}{2}\right)\right] \times \frac{1}{2\pi^{3/2}}\left[\exp\left(\frac{-(2.5-2.1)^2}{2}\right) + \exp\left(\frac{-(2.5-6.2)^2}{2}\right) + \exp\left(\frac{-(2.5-1.3)^2}{2}\right)\right] \times \dots \times \frac{3}{6}$$

$P(-|x)$ = similarly

→ BASIC-STRUCTURE OF PATTERN RECOGNISATION

(a) PATTERN
* an object, process or event that is given a name
* an image is not a pattern till it is given a name

(b) PATTERN RECOGNISATION
* The process by which machines can
(i) observe the environment
(ii) identify the patterns of interest
(iii) classify the pattern

(c) CLASSES
Set of pattern having common attributes and common source/origin in most cases

(d) FEATURE
The attributes / values that are common in objects of same class, and different in pattern of different classes is called feature

→ DATA-ACQUISTION AND SENSING
The process of acquiring and extracting the relevant and useful data from the environment
eg : extracting address from envelop

→ FEATURE EXTRACTION

The process of extracting relevant features about the data and preparation of feature vector containing numerical values based upon which we can classify the patterns.

→ PRE-PROCESSING

This step deals with removal of noise from the pattern. Noise can be defined as property of sensed pattern that is due to randomness of world and not due to model.

→ POST-PROCESSING

Involves calculation of accuracy, performance, evaluation of confidence etc.

* NORMALISATION OF FEATURE POINTS IN KNN

(a) To bring each feature in 0–1,

x_{ij} is replaced by $\frac{x_{ij} - \min(\text{column } j)}{\max(\text{col } j) - \min(\text{col } j)}$

(b) eg.

x_1	x_2	x_3	Class
10	15	20	1
25	30	1	2
2	5	3	1

Changes to

x_1	x_2	x_3	Class
$\frac{10-2}{25-2}$	$\frac{15-5}{30-5}$	$\frac{20-1}{20-1}$	1
⋮	⋮	⋮	2
⋮	⋮	⋮	1

(c) Feature points: YYYNNNNN

Find P(Y) that maximises the likelihood

ANS. $P(y) = \begin{cases} x & y = Y \\ 1-x & y = N \end{cases}$

For given feature points

$P(x) = x^3 (1-x)^5$

$\log P(x) = 3 \log x + 5 \log(1-x)$

Diff wrt x

$$\frac{d \log P(x)}{dx} = 0 = \frac{3}{x} + \frac{5(-1)}{1-x}$$

$$\frac{3}{x} = \frac{5}{1-x}$$

$$3 - 3x = 5x$$

$$8x = 3 \quad x = \frac{3}{8}$$

$\therefore P(y = \text{Yes}) = 3/8$

$P(y = \text{No}) = 5/8$

WEIGHTED-KNN ALGORITHM

→ In weighted-KNN algorithm, higher preference or weights are assigned to closer neighbours and less weights are assigned to far away neighbours.

→ $w_i = \frac{1}{d(x_i, y)}$ where y = testing point, x_i = training point

ALGORITHM:

(i) Find dist b/w test point and all training points

(ii) Find the k - nearest points.

(iii) $w_i = \frac{1}{d(x_i, y)}$ = weight assigned to each point such that closer points have a higher weight. This w_i can be varied based on the application.

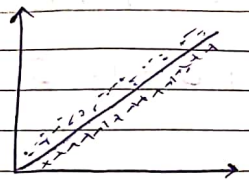
(iv) Compute m_j for each class

$$m_j = \sum_{i=1}^{K} \delta(x_i, j) \times w_i$$

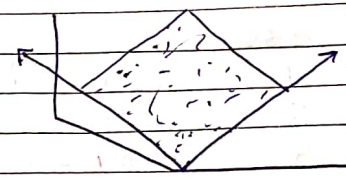
where $\delta(x_i, j) = 1$ if class$(x_i) = j$
0 else

(v) class with max m_j is the required class of test point.

DIMENSIONALITY REDUCTION



Data is in 2D, but it is about a line only. So instead of representing in 2D wrt (0,0) origin, we can take eq^n of line and represent in 1D



Data is in 3D, can be represented in 2D

RANK OF MATRIX

number of linearly independent column of matrix is rank of matrix

eg1

	M	T	W	Th	Fr	~~Sat~~	~~Sun~~
A	1	1	1	0	0		
B	2	2	2	0	0		
C	1	1	1	0	0		
D	5	5	5	0	0		
E	0	0	0	2	2		
F	0	0	0	3	3		
G	0	0	0	1	1		

eg2 $A = \begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix}$

Row 3 = Row 1 − Row 2

Basic axis [0 0 0]

change it to [1 2 1]

[−2 −3 1]

[1 2 1] + 0[−2 −3 1] ← first row [0 1 0]

0[1 2 1] + 1[−2 −3 1] ← second row [0 1]

third row [1 1]

actually [1 −1]

called vectors

for eg 1, base axis [1 1 1 0 0]
[0 0 0 1 1]

row	
1	[1 0] = [1 1 1 0 0] + 0 × [0 0 0 1 1]
2	[2 0]
3	[3 0]
4	[4 0]
5	[0 2]
6	[0 3]
7	[0 1]

→ called POINTS

WHY DIMENSIONITY REDUCTION?

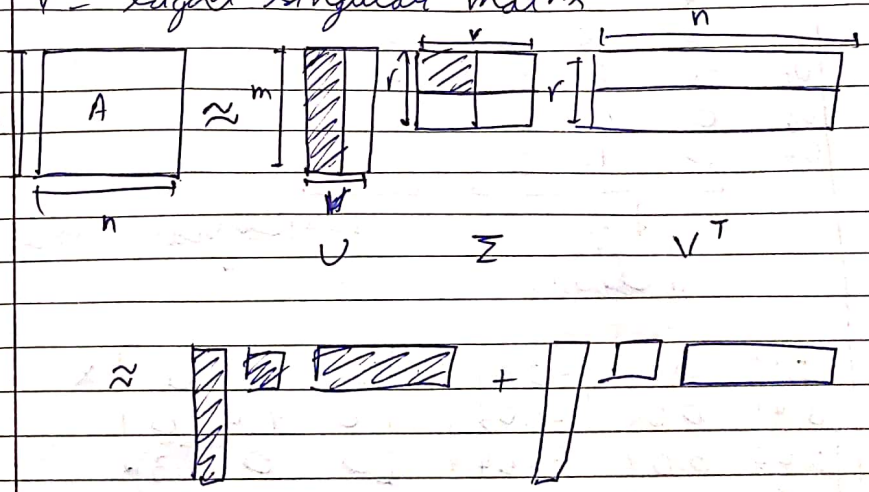
(i) Discovers hidden correlation
(ii) Remove noisy and redundant features
(iii) interpretation and visualization
(iv) easier storage and processing of data

SVD - SINGULAR VALUE DECOMPOSITION

$$A_{m \times n} = U_{m \times r} \Sigma_{r \times r} (V_{n \times r})^T$$

It is always possible to decompose a real matrix A into $U \Sigma V^T$ where
(i) U, Σ, and V are unique
(ii) U and V are column orthonormal
meaning → $U^T U = I$
$V^T V = I$
(iii) Σ is diagonal matrix where each entry is positive and sorted in decreasing order

A = input data matrix (m × n)
U = left singular matrix → determinant ≠ 0
Σ = singular matrix
V = right singular matrix



As Σ is decreasing matrix, most of the values at the end are 0. So if we know which values to neglect, our task will be reduced

$$A = U \Sigma V^T$$
$$= U \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 \\ 0 & 0 & 0 & \sigma_4 \end{bmatrix} V^T$$

→ as σ_3 and σ_4 are close to 0, we can neglect this part

$$\|M_p\| = \sqrt{\Sigma_{ij} (A_{ij} - B_{ij})}$$

eg→ Following is data of movies

U_1	1	1	1	0	0
U_2	3	3	3	0	0
U_3	4	4	4	0	0
U_4	5	5	5	0	0
U_5	0	2	0	4	4
U_6	0	0	0	5	5
U_7	0	1	0	2	2

sci-fic movies

Romantic movies

$$\begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.02 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & 0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times$$

This is the min value amongst all 3. So as this is closest to 0, we neglect it. As we neglect last value, we also neglect last column of U and last row of V^T

$$\begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.12 & 0.12 & -0.69 & 0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

V- PRODUCT

It gives the coordinates of point in credential axis.

$$U \Sigma V^T = \begin{bmatrix} 1.61 & 0.19 & -0.01 \\ 5.08 & 0.66 & -0.03 \\ 6.82 & 0.85 & -0.05 \\ 8.43 & 1.04 & -0.06 \\ 1.86 & -5.60 & -0.84 \\ 0.86 & -6.93 & - \\ 0.86 & -7.75 & - \end{bmatrix}$$

eg→ Test $q = [5\ 0\ 0\ 0\ 0]$

~~[4 5 0 0 0]~~

$[0\ 4\ 5\ 0\ 0]$

amount movie belongs to romantic

$[0\ 4\ 5\ 0\ 0] \times V$

$$[0\ 4\ 5\ 0\ 0] \times \begin{bmatrix} 0.56 & 0.12 \\ 0.59 & -0.12 \\ 0.56 & 0.12 \\ 0.09 & -0.69 \\ 0.09 & 0.69 \end{bmatrix} = [5.2 \quad 0.4]$$

Amount the movie belongs to sci-fi

PRINCIPLE COMPONENT ANALYSIS

→ Mean
→ Std deviation } 1D
→ Variance

→ Co-variance } 2D

$$Var = \frac{\sum_{i=1}^{n} (x_i - x)(x_i - x)}{n-1}$$ → ALWAYS

$$Cov(X,Y) = \frac{\sum_{i=1}^{n} (x_i - x)(y_i - y)}{n-1}$$ (USE n in MEAN)

eg→

Hours	Marks
9	39
15	56
25	93
14	61
10	50
18	75
0	32
16	85
5	42
19	70
16	66
20	80

Find covariance

ANS. $\mu_x = \frac{\sum x_i}{n}$

$= \frac{}{} 13.92$

$\mu_y = \frac{\sum y_i}{n}$

$= \frac{749}{12} = 62.42$

~~Cov(x,y)~~
~~$= (-39.92)(-30.42)$~~
$\neq$
~~$= (-13.92)(-30.42)$~~

$M_i - \bar{M}$	$M_i - \bar{M}$	$M_i - \bar{M}$	$M_i - \bar{M}$
-4.92	-23.42	2.08	22.58
1.08	-6.42	-8.92	-20.42
11.08	30.58	5.08	7.58
0.08	-1.42	2.08	3.58
-3.92	-12.42	6.08	7.58
4.08	12.58		
-13.92	-30.42		

$(x_i-\bar{x})(y_i-\bar{y})$

115.23
-6.93
338.83
-0.11
48.69
51.33
423.45
46.97
182.15
38.15
7.45
106.89

$cov(x,y) = 104.59$

$C = MATRIX = \begin{bmatrix} Cov(x,x) & Cov(x,y) \\ Cov(y,x) & Cov(y,y) \end{bmatrix}$

COVARIANCE MATRIX IN 3D

$$C = \begin{bmatrix} Cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & Cov(z,z) \end{bmatrix}$$

eg

X	Y	$X_i - \mu_x$	$Y_i - \mu_y$	$(x_i-\mu_x)(y_i-\mu_y)$
2.5	2.4	0.69	+0.49	0.3381
0.5	0.7	-1.31	-1.21	1.5851
2.2	2.9	0.39	0.99	0.3861
1.9	2.2	0.09	0.29	0.0261
3.1	3.0	1.29	1.09	1.4061
2.3	2.7	0.49	0.79	0.3871
2.0	1.6	0.19	-0.31	-0.0589
1.0	1.1	-0.81	-0.81	0.6561
1.5	1.6	-0.31	-0.31	0.0961
1.1	0.9	-0.71	-1.01	0.7171

$\mu_x = \frac{18.1}{10} = 1.81$ $\quad \mu_y = \frac{1.91}{10}$

$Cov(x,y) = \frac{5.539}{10} = 0.5539$

$Var(x) = 0.4761 + 1.7161 + 0.1521 + 0.0081 + 1.6641 + 0.2401 + 0.0361 + 0.6561 + 0.0961 + 0.7171 + 0.5041$

$= \frac{5.549}{10} = 0.5549$

EIGEN VECTOR

$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}$ $\quad |A - \lambda I| = 0$

Matrix, const, Identity Matrix

$$\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = 0$$

$$\begin{bmatrix} -\lambda & 1 \\ -2 & -3-\lambda \end{bmatrix} = 0$$

$\lambda^2 + 3\lambda + 2 = 0$

$\lambda_1 = -1 \quad \lambda_2 = -2 \Rightarrow$ EIGEN VALUE

$(A - \lambda I)V = 0$

↳ EIGEN VECTOR

$\lambda = -1$

$$\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -2 & -2 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ -2 & -2 \end{bmatrix} \begin{bmatrix} V_{11} \\ V_{12} \end{bmatrix} = 0$$

$V_{11} + V_{12} = 0 \qquad V_{11} = -V_{12}$

$-2V_{11} - 2V_{12} = 0 \qquad V_{11} = -V_{12}$

$$V_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

for λ_2

$$\begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix} \begin{bmatrix} V_{21} \\ V_{22} \end{bmatrix} = 0$$

$2V_{21} + V_{22} = 0 \qquad -2V_{21} - V_{22} = 0$

$V_{21} = \frac{-V_{22}}{2} \qquad V_{21} = 1 \quad V_{22} = -2$

$$V_2 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

PROPERTIES:

(i) Can only be found for square matrix
(ii) Not every square matrix has eigen vector
(iii) For given $N \times N$ matrix, there are max of N eigen vectors
(iv) If you scale the vectors by some amount before you multiply it, you will get same multiple of it as result

Q. Find eigen vector and eigen values.

$$A = \begin{bmatrix} 8 & -8 & -2 \\ 4 & -3 & -2 \\ 3 & -4 & 1 \end{bmatrix}$$

$-3-8 = -11$
$4+6 = 10$
$-16+9 = -7$
$-11-7+10$

ANS. $|A - \lambda I| = 0$

$$\begin{vmatrix} 8-\lambda & -8 & -2 \\ 4 & -3-\lambda & -2 \\ 3 & -4 & 1-\lambda \end{vmatrix} = 0$$

$(8-\lambda)(-3+\lambda^2+2\lambda) + 8(4-4\lambda+6)$
$-2(-16+9+3\lambda) = 0$

$(8-\lambda)(\lambda^2+2\lambda-11) + 8(10-4\lambda)$
$-2(3\lambda-7) = 0$

$-\lambda^3 - 2\lambda^2 + 11\lambda + 8\lambda^2 + 16\lambda - 88 + 80 - 32\lambda$
$-6\lambda + 14 = 0$

$-\lambda^3 + 6\lambda^2 - 11\lambda + 6 = 0$
$-\lambda^3 + \lambda^2 + 5\lambda^2 - 5\lambda - 6\lambda + 6 = 0$
$-\lambda^2(\lambda-1) + 5\lambda(\lambda-1) - 6(\lambda-1) = 0$
$(\lambda-1)(-\lambda^2+5\lambda-6) = 0$
$(\lambda-1)(-\lambda^2+3\lambda+2\lambda-6) = 0$
$(\lambda-1)(-\lambda(\lambda-3)+2(\lambda-3)) = 0$
$\lambda = 1, 2, 3.$

$\lambda_1 = 1$

$$\begin{bmatrix} 7 & -8 & -2 \\ 4 & -4 & -2 \\ 3 & -4 & 0 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = 0$$

$7V_1 - 8V_2 - 2V_3 = 0$
$4V_1 - 4V_2 - 2V_3 = 0$
$3V_1 - 4V_2 = 0$

$3V_1 - 4V_2 = 0$
$V_1 = 4 \quad V_2 = 3$

$V_1 = \begin{bmatrix} 4 \\ 3 \\ 2 \end{bmatrix}$

$16 - 12 - 2V_3 = 0$
$2V_3 = 4 \quad V_3 = 2$

$\lambda_1 = 2$

$$\begin{bmatrix} 6 & -8 & -2 \\ 4 & -5 & -2 \\ 3 & -4 & -1 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = 0$$

$4V_1 - 5V_2 - 2V_3 = 0$
$3V_1 - 4V_2 - V_3 = 0$
$\rightarrow 6V_1 - 8V_2 - 2V_3 = 0$
$- \quad 4V_1 - 5V_2 - 2V_3 = 0$
$2V_1 - 3V_2 = 0$
$2V_1 = 3V_2$
$V_1 = 3 \quad V_2 = 2$

$V_2 = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$

$3(3) - 4(2) - V_3 = 0$
$V_3 = 9 - 8 = 1$

$\lambda_1 = 3$

$$\begin{bmatrix} 5 & -8 & -2 \\ 4 & -6 & -2 \\ 3 & -4 & -2 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = 0$$

$5V_1 - 8V_2 - 2V_3 = 0$
$4V_2 - 6V_2 - 2V_3 = 0$

$V_1 - 2V_2 = 0 \qquad V_1 = 2V_2$
$V_1 = 2 \quad V_2 = 1$
$5(2) - 8(1) - 2V_3 = 0$
$10 - 8 = 2V_3 \qquad V_3 = 1$

$3V_1 - 4V_2 - 2V_3 = 0$
$3(2) - 4(1) - 2(1) = 0$ ✓

$V_3 = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix}$

FINDING EIGEN EQ^n

$\lambda^3 - [\text{Sum of diagonal elements}]\lambda^2 + [\text{Sum of diag minor}]\lambda - |A| = 0$

minor of A_{11}
+ minor of A_{22}
+ minor of A_{33}

$$= \begin{vmatrix} -3 & -2 \\ -4 & 1 \end{vmatrix} + \begin{vmatrix} 8 & -2 \\ 3 & 1 \end{vmatrix} + \begin{vmatrix} 8 & -8 \\ 4 & -3 \end{vmatrix}$$

$= 11$

→ PCA is statistical procedure that uses an orthogonal transformation to convert a set of observation of possibly correlated variables into a set of values of linearly uncorrelated variables called PRINCIPLE COMPONENTS

→ It is a way of identifying pattern in data and expressing data in such a way as to highlight their similarity and differences.

→ Since pattern in data can be hard to find in data of high dimensions where luxury of graphical representation is not available, PCA is powerful tool to analysis that data.

→ ~~another~~ once you have found this pattern in data, then you can compress the data by reducing no. of dimensions without much loss of info

PROCEDURE:

(i) collect the data
(ii) subtract the mean
(iii) Calculate co-variance matrix
(iv) Calculate eigen vectors and corresponding eigen values of co-variance matrix
(v) Choose first k-eigen vectors (sorted in decreasing order)
(vi) that will be ~~fit~~ new k-dimensions.
(vii) transform the original and dimensional data points into k-dimensions.

→ This final step in PCA where we have to choose component ~~for~~ OR eigen vector in such a way that loss of info is min we wish to keep in our data and form a feature vector.

→ Final data = Row feature vector x Row data adjusted
(matrix with eigen vector in column transpose so that eigen vector are now in row with most significant eigen vector at top) (mean adjusted data transposed.)

QUES. $x = [2.5, 0.5, 9.2 -]]$ before
$y = [2.4, 0.7, -]$

$cov(x,y) = 0.61544$
$cov(x,x) = 0.61655$

STEP (i) (ii) (iii) $cov = \begin{bmatrix} 0.616 & 0.615 \\ 0.615 & 0.716 \end{bmatrix} = C$.

STEP (iv) $|C - \lambda I| = 0$

$\begin{vmatrix} 0.616-\lambda & 0.615 \\ 0.615 & 0.716-\lambda \end{vmatrix} = \begin{vmatrix} 0.385 & 0.615 \\ 0.615 & -0.284 \end{vmatrix}$

$= 0.109 - 0.378 = -0.269$

$\begin{vmatrix} 0.616-\lambda & 0.615 \\ 0.615 & 0.716-\lambda \end{vmatrix}$

$= (0.616-\lambda)(0.716-\lambda) - 0.378$

$= 0.441 + \lambda^2 - 0.441 - 1.332\lambda - 0.378 = 0$

$\lambda^2 - 1.332\lambda + 0.063 = 0$

$D = 1.774274 - 0.252$

$= 1.522$

$\sqrt{D} = 1.234$

$$\lambda = \frac{1.332 \pm 1.234}{2}$$

$$= 1.283,\ 0.049$$

$\lambda = 1.283$

$$\begin{bmatrix} 0.616 - 1.283 & 0.615 \\ 0.615 & 0.716 - 1.283 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} -0.667 & 0.615 \\ 0.615 & -0.567 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = 0$$

$$-0.667 V_1 + 0.615 V_2 = 0$$

$$0.615 V_1 - 0.567 V_2 = 0$$

~~$-0.615 \times 0.667 V_1 + (0.615)^2 V_2 = 0$~~

~~$0.615 \times 0.667 V_1 - (0.567)(0.667) V_2 = 0$~~

$$V_1 = \frac{0.615}{0.667} V_2 = 0.922 V_2$$

→ Raw adjusted data = Raw feature vector inverse × final data

= Raw feature vector transpose × final data

(~~Raw~~ as orthogonal, inverse = transpose)

→ original data = (Raw feature vector transpose × final data) + Mean

eg → $a^2 + b^2 = 1$

$V_1^2 + V_2^2 = 1$ V_1, V_2 are vectors

$$\begin{bmatrix} -9.9453 & 7.9876 \\ 7.9876 & -6.4153 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$-9.9453 a + 7.9876 b = 0$$

$$-9.9453 a + 7.9876 \sqrt{1 - a^2} = 0$$

$$\frac{\sqrt{1-a^2}}{a} = \frac{9.9453}{7.9876} = ~~1.550~~ 1.2451$$

$$\frac{1 - a^2}{a^2} = ~~2.4025~~ 1.550$$

$$1 - a^2 = ~~2.4025 a^2~~ 1.550 a^2$$

$$1 = ~~3.4025 a^2~~ 2.550 a^2$$

$$\frac{1}{a^2} = 2.550$$

$a^2 = 0.3921$ or divide final ans by $\sqrt{a^2 + b^2}$

$a = 0.6262$

$b = 0.7796$

$V_1 = 0.922 V_2$ $V_1 = 0.922$, $V_2 = ~~0.9~~ 1$

An normalisation:

$$V_1 = \frac{0.922}{\sqrt{1 + (0.922)^2}} = 0.677$$

$V_2 = 0.79$

Page No.
Date

DECISION TREE

→ It is a decision support tool that uses a tree like model of decision and their possible consequence including chance event outcomes, resource cost and utility

→ we can also say that decision tree is flowchart like structure in which each internal node represent test on an attribute. each branch represent outcome of test and each leaf node represent a class label

→ A path from root to leaf represent classification rules.

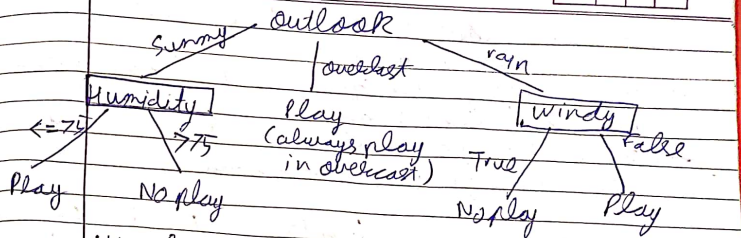
eg:

outlook	Temp	Humidity	Windy	Class
Sunny	79	90	True	No play
Sunny	56	70	False	Play
Sunny	79	75	True	Play
Sunny	60	90	True	No play
overcast	88	88	False	No play
overcast	63	75	True	play
overcast	88	95	False	play
rain	78	60	False	play
rain	6866	70	False	play
rain	68	60	True	No play

→ which feature to be at which node is decided using formula

→ ~~last data part is anomaly~~

Page No.
Date



We found 5 leaf node i.e it generate 5 rules

RULES

(i) If it is sunny and humidity <=75 then play

(ii) If it is sunny and humidity >75 then do not play

(iii) If it is overcast, then play

(iv) If it is rainy and windy then no play

(v) If it is rainy and not windy then play

CLASSIFICATION-STEPS

→ The classification of unknown input vector is done by traversing decision tree from root node to leaf node

→ record enteries tree at root node

→ at root, a test is applied to determine which child node the record will encounter next

→ This process is repeated until record arrives at a leaf node

→ All the record that end up at

a leaf node of the tree are classified in the same way.

→ There is a unique path from root to each leaf node

→ The path is a route which is used to classify the record.

	outlook	Temp	Humid	Wind	Decision
	Sunny	H	High	Weak	No
	Sunny	H	High	Strong	No
	overcast	H	High	Weak	Yes
	Rainy	Mild	High	Weak	Yes
	Rainy	C	Normal	Weak	Yes
	Rainy	C	Normal	Strong	No
	overcast	C	Normal	Strong	Yes
	Sunny	Mild	High	Weak	No
	Sunny	C	Normal	Weak	Yes
	Rainy	Mild	Normal	Weak	Yes
	Sunny	Mild	Normal	Strong	Yes
	overcast	Mild	High	strong	Yes
	overcast	H	Normal	weak	Yes
	Rainy	Mild	High	Strong	No

Ans Entropy $= -\sum P(I) \log_2 P(I)$

Gain(S, A) = Entropy(S) − $\sum P(S|A)$ Entropy(S/A) → MAXIMISE THIS TO MINIMISE SCATTERING

Entropy (Decision) $= \frac{5}{14}\log_2\frac{5}{14} + \frac{9}{14}\log_2\frac{9}{14}$

$P(NO) = 0.940$

Gain (Decision, Wind)

= Entropy (Decision) $- \sum P(\text{Wind})$ Entropy (D/Wind)

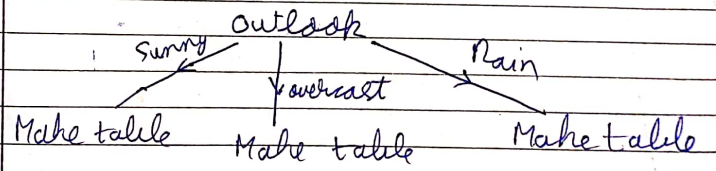
= Entropy $(D) - P(\text{Weak})$ Entropy $(D/\text{Weak}) - P(D/\text{Strong})$ Entropy (D/Strong)

= Entropy $(D) - \frac{8}{14}\left[\frac{2}{8}\log\frac{2}{8} + \frac{6}{8}\log\frac{6}{8}\right] - \frac{6}{14}\left[\frac{3}{6}\log\frac{3}{6} + \frac{3}{6}\log\frac{3}{6}\right]$

$= 0.940 - \frac{8}{14}\left[\frac{2}{8}\log\frac{2}{8} + \frac{6}{8}\log\frac{6}{8}\right] - \frac{6}{14}\log\frac{3}{6}$

→ Gain (Decision, outlook), Gain (Decision, Humid), Gain (Decision, Temp) is calculated similarly

→ Gain (Decision, outlook) is maximum



Page No.
Date

Table for Sunny

Temp	Humid	Wind	Decision
H	High	Weak	No
H	High	Strong	No
Mild	High	Weak	No
C	Normal	Weak	Yes
Mild	Normal	Strong	Yes