USN | | | | | | | | | |

Internal Assessment Test 2 – Nov 2023

Sub:	Artificial Intelligence and Machine Learning					Sub Code:	18CS71	Branch:	CSE	
Date:	04.11.2023	Duration:	90 mins	Max Marks:	50	Sem/Sec:	7 A		OBE	
	Answer any FIVE FULL Questions							MARKS	CO	RBT

1 (a) Explain concept of entropy and information gain in decision tree with graphs and formulae. — Marks: **3**, CO: **CO1**, RBT: **L1**

Entropy is a concept from information theory that characterizes the (im)purity of an arbitrary collection of examples.

Entropy(S)

$$Entropy(S) \equiv -p_{\oplus}\log_2 p_{\oplus} - p_{\ominus}\log_2 p_{\ominus}$$

For c-classifications,

$$Entropy(S) \equiv \sum_{i=1}^{c} -p_i \log_2 p_i$$

Information Gain : expected reduction in entropy caused by partitioning the examples according to this attribute.

Gain(S, A) of ***an*** attribute **A** relative to a collection of examples ***S***, is defined as

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

where ***Values(A)*** is the set of all possible values for attribute A, and ***S***, is the subset of S for which attribute A has value v (i.e., $S_v = \{s \in S \mid A(s) = v\}$ *)*

(b) Derive decision tree for the following dataset — Marks: **7**, CO: **CO3**, RBT: L3

Show the steps in the calculation.

Draw the final tree and write the conjunction of disjunctions.

	PlayTennis	Outlook	Temperature	Humidity	Wind
0	No	Sunny	Hot	High	Weak
1	No	Sunny	Hot	High	Strong
2	Yes	Rain	Cool	Normal	Weak
3	No	Rain	Cool	Normal	Strong
4	Yes	Overcast	Cool	Normal	Strong
5	No	Sunny	Mild	High	Weak

Iteration 1 :

Information gain of Outlook:0.58497

Information gain of Temperature:0.4591

Information gain of Humidity:0.4591

Information gain of Wind:0.0000

Highest Information gain is”

Iteration 2 :

Information gain of Temperature:0.0000
Information gain of Humidity:0.0000
Information gain of Wind:_____

	PlayTennis	Outlook	Temp	Humidity	Wind
1	No	Sunny (1)	Hot (1)	high (1)	weak (1)
2	No	Sunny (2)	Hot (2)	high (2)	Strong
3	Yes	Rain (1)	Cool (1)	Normal (1)	Weak (2)
4	No	Rain. (2)	Cool (2)	Normal (2)	Strong
5	Yes	Overcast	Cool (3)	Normal (3)	Strong
6	No	Sunny (3)	Mild	High (3)	Weak (3)

$E(S) = -\frac{2}{6}\log_2\frac{2}{6} - \frac{4}{6}\log_2\frac{4}{6}$

$= 0.52832 + 0.38998$

$= 0.9183$

Information Gain (Outlook)

$E_{Sunny} = 0$

$E_{Rain} = 1$

$E_{Overcast} = 0$

$IG(E, Outlook) = 0.9183 - (1 \times \frac{2}{6})$

$= 0.58497$

Information Gain (Temp)

$E_{Hot} = 0$

$E_{cool} = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}$

$=$ ~~0.18904~~

$= 0.38998 + 0.5283$

$= 0.9183$

$E_{Mild} = 0.$

$IG(E, Temp) = 0.9183 - (0.9183 \times \frac{3}{6})$

$= 0.4592$

Information Gain (Humidity)

$E_{high} = 0$

$E_{Normal} = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} = 0.9183$

$IG(E, Humidity) = 0.9183 - (0.983 \times \frac{3}{6})$

$= 0.4592.$

Information Gain (Wind)

$E_{weak} =$ ①, $E_{strong} = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}$

$= -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}$ $= 0.9183$

$= 0.9183$

IG(E, Wind) 0.9183 − ((3/6 × 0.9183) + (3/6 × 0.9183))

= 0

Highest Information Gain: Outlook

Outlook — Rain
— Sunny
— Overcast

$E_{Rain} = 1$

PlayTennis	Outlook	Temperature	Humidity	Wind
yes	rain	cool	Normal	Weak
no	rain	cool	Normal	Strong

Information Gain (Temperature)

$E_{cool} = 1$

$IG(E_{rain}, Temp) = 1 - 1 = 0$

Information Gain (Humidity)

Wind — Weak - yes
— Strong - no

$E_{normal} = 1$

$IG(E_{rain}, Humidity) = 1 - 1 = 0$

Information Gain (Wind)

$E_{weak} = 0$ $E_{strong} = 0$

$IG(E_{rain}, wind) = 1 - 0 = 1$

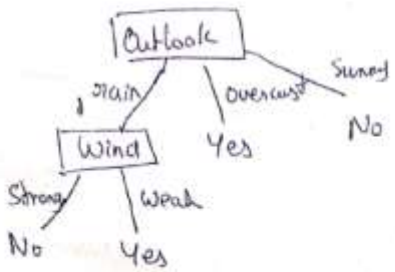
Highest Information gain: Wind

Outlook − Overcast − Yes

Outlook − Sunny −

PlayTennis	Outlook	Temperature	Humidity
No	Sunny		
No	Sunny		
No	Sunny		

Outlook - Sunny - No



2 (a)	Write the backpropagation algorithm. Comment about the intuition behind the error calculations.	6	CO2	L2

Write the backpropagation algorithm. Comment about the intuition behind the error calculations.

BACKPROPAGATION(training_examples, η, n_{in}, n_{out}, n_{hidden})

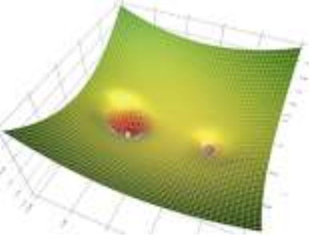
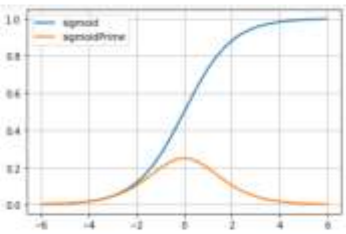
Each training example is a pair of the form $(\vec{x}, \vec{y})$ *where* $\vec{x}$ *is the vector of network input values, and* $\vec{t}$ *is the vector of target network output values.*

η is the learning rate (e.g., ***.O5).*** *n_{in}, is the number of network inputs, n_{hidden} the number of units in the hidden layer, and n_{out} the number of output units.*

*The inputfiom unit i into unit j is denoted x_{ji}, and the weight from unit **i** to unit j is denoted w_{ji}.*

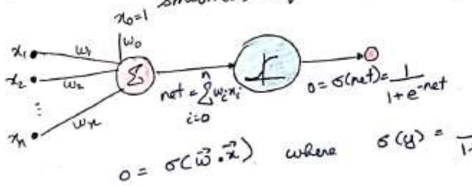
- Create a feed-forward network with n_{in}, inputs, n_{hidden} hidden units, and n_{out} output units.
- Initialize all network weights to small random numbers (e.g., between –0.05 to +0.05)
- Until the termination condition is met, Do
  - For each $(\vec{x}, \vec{y})$ in *training_examples,* **Do**

    Propagate the input forward through the network:

	1. Input the instance $\vec{x}$ to the network and compute the output o_u of every unit ***u*** in the network. *Propagate the errors backward through the network:* 2. For each network output unit k, calculate its error term δ_k $$\delta_k \leftarrow o_k(1-o_k)(t_k-o_k)$$ 3. For each hidden unit *h*, calculate its error term δ_h $$\delta_h \leftarrow o_h(1-o_h)\sum_{k \epsilon outputs} w_{kh}\delta_k$$ 4. Update each network weight w_{ji} where $$\boldsymbol{w_{ji} \leftarrow w_{ji} + \Delta w_{ji}}$$ where $$\boldsymbol{\Delta w_{ji} = \eta \delta_j x_{ji}}$$ **Intuition** $(t_k - o_k)$ – comes from the delta rule $o_k(1 - o_k)$ – comes from the derivative of sigmoid since training examples provide target values t_k for network outputs, no target values are directly available to indicate the error of hidden units' values. The error term for hidden unit *h* is calculated by summing the error terms j_k for each output unit influenced by *h*, weighting each of the δ_k's by w_{kh}, the weight from hidden unit *h* to output unit *k*. This weight characterizes the degree to which hidden unit h is "responsible for" the error in output unit k.			
(b)	 Above is a parabolic space for the hypothesis space for weights with respect to the associated E values. Answer briefly in 1 or 2 lines. i) What does the global minimum represent? The hypothesis that minimizes the error for the given training data ii) Why do we take the negative of this vector? $-\nabla E(\overrightarrow{w})$ To go in the direction of the negative gradient on a descent to the minima iii) What is the impact of learning rate(η) on the training rule for gradient descent? It controls the step size of the descent. A high learning rate risks overstepping the minima. A sufficiently small learning rate ensures convergence. iv) Is backpropagation algorithm for MLP guaranteed to find global minimum? No, in case where there are multiple local minima, it is not guaranteed to find the global minimum.	**4**	**CO2**	L2
3 (a)	With a neat diagram explain the sigmoid function and it's derivative for the differentiable threshold unit. 	**5**	**CO1**	L2

A differentiable threshold unit

- networks should be capable of representing highly nonlinear functions.
- perceptron unit - discontinuous and undifferentiable for gradient descent.
- Sigmoid function: a unit similar to perceptron but with a smoothed, differentiable threshold function.



$o = \sigma(\vec{w} \cdot \vec{x})$ where $\sigma(y) = \frac{1}{1+e^{-y}}$

- σ - sigmoid/ logistic function
- output range is between 0 and 1
- increases monotonically with the input.
- it maps a very large input domain to a small range of outputs - squashing function of the unit.
- the derivative of the sigmoid function can be represented as

$$\frac{d\sigma(y)}{dy} = \sigma(y) \cdot (1-\sigma(y))$$

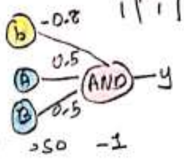
(b) Design a perceptron that implements AND function.
Why is that a single layer perceptron cannot be used to represent XOR function?

5 | CO1 | L2

1 (True) . -1 (False)

A	B	Out
0	0	0
0	1	0
1	0	0
1	1	1

$b = -0.8$, $w_1 = w_2 = 0.5$



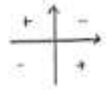
A=0, B=0
$x1(0.5) + x2(0.5) - 0.8$
$0 + 0 - 0.8 = -0.8$, $y<0$, so -1

A=0, B=1
$= x1(0.5) + x2(0.5) - 0.8$
$= 0 \times 0.5 + 1 \times 0.5 - 0.8$
$= 0.5 - 0.8 = -0.3$, $y<0$, so -1

A=1, B=0
$= x1(0.5) + x2(0.5) - 0.8$
$= 1 \times 0.5 + 0 \times 0.5 - 0.8$
$= -0.3$ so -1

A=1, B=1
$= 1 \times 0.5 + 1 \times 0.5 - 0$
$= 0.5 + 0.5 - 0.8$
$= 1 - 0.8$
$= 0.2$, $y>0$, so 1

- NAND ($\neg$ AND) NOR ($\neg$ OR) can be represented.
- XOR whose value is 1 only when $x_1 \neq x_2$ cannot be represented

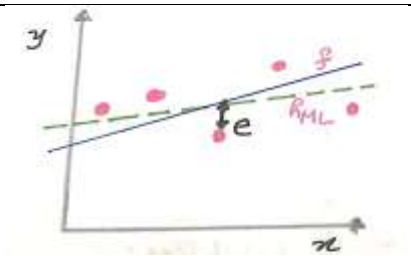
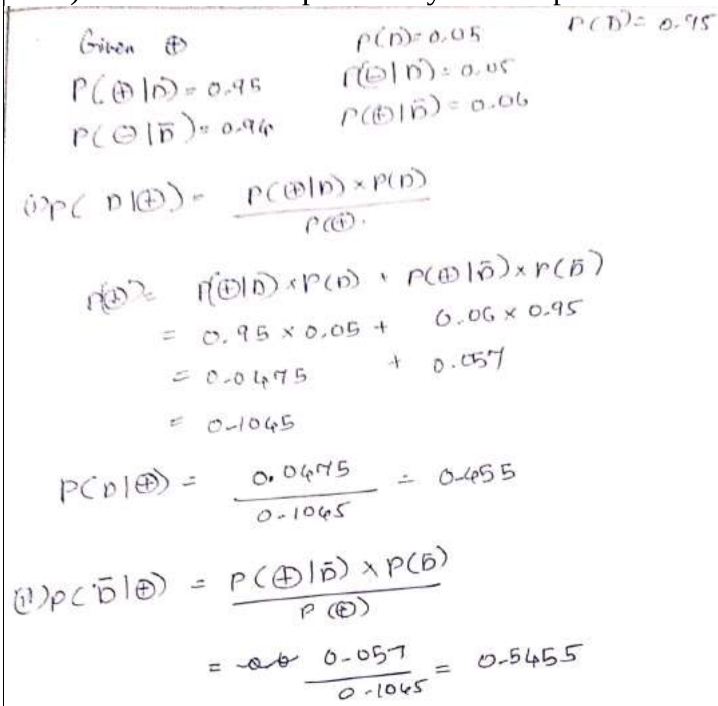


non-linearly separable.

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

$w_0 + w_1 \cdot 0 + w_2 \cdot 0 < 0 \Rightarrow w_0 < 0$
$w_0 + w_1 \cdot 1 + w_2 \cdot 0 \geq 0 \Rightarrow w_1 \geq -w_0$
$w_0 + w_1 \cdot 0 + w_2 \cdot 1 \geq 0 \Rightarrow w_2 \geq -w_0$
$w_0 + w_1 \cdot 1 + w_2 \cdot 1 < 0 \Rightarrow w_1 + w_2 < -w_0$

4th condition contradicts 2 & 3.
- not possible to find a set that satisfies these set of inequalities.

4 (a)	 What does f, e, h_{ML} and the 5 dots represent in the above problem of learning a linear function?	**3**	**CO1**	L1
(b)	Explain EM algorithm and derivation of k means for estimating means of k Normal distributions	**7**	**CO2**	L2
5 (a)	A patient takes a lab test and the result comes back positive. It is known that the test returns a correct positive result in only 95% of the cases and a correct negative result in only 94% of the cases. Furthermore, only 0.05 of the entire population has this disease. i) What is the probability that the patient has the disease? ii) What is the probability that the patient does not have the disease?  	**5**	**CO1**	L2
(b)	Explain Naïve Bayes Classifier using the Bayes theorem and comment about the advantages of the classifier.	**5**	**CO2**	L2

■ Naive Bayes Classifier

- applies to learning tasks where each instance x is described by
  - a conjunction of attribute values
  - target function $f(x)$ can take on any value from some finite set V.
  - attributes are described by tuples $\langle a_1, a_2, \ldots a_n \rangle$

Bayesian approach: for classifying new instance, assign to the most probable target value, v_{MAP} given the attribute values $\langle a_1, a_2, \ldots a_n \rangle$

$$v_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} \; P(v_j \mid a_1, a_2, \ldots a_n)$$

use Bayes theorem to rewrite this expression as

$$v_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} \; \frac{P(a_1, a_2, \ldots a_n \mid v_j)\, P(v_j)}{P(a_1, a_2, \ldots a_n)}$$

← constant, so remove.

$$= \underset{v_j \in V}{\operatorname{argmax}} \; P(a_1, a_2, \ldots a_n \mid v_j)\, P(v_j)$$

- Naive Bayes classifier is based on assumption that attribute values are conditionally independent given the target value.
- So $P(a_1, a_2, \ldots a_n \mid v_j) = \prod_i P(a_i \mid v_j)$

$$v_{NB} = \underset{v_j \in V}{\operatorname{argmax}} \; P(v_j) \prod_i P(a_i \mid v_j)$$

- v_{NB} denotes the target output value of the naive Bayes Classifier.
- In naive Bayes, $P(v_j)$ and $P(a_i \mid v_j)$ terms are estimated based on their frequencies in the training data.
- no search is involved in NB learning method.
- hypothesis is formed simply by counting frequencies of various data combinations within the training examples.

Q. No.	Question	Marks	CO	Level
6 (a)	Apply Naïve-Bayes classifier for the below dataset to classify the new instances h (Color =White, legs=2, Height = Short, Smelly =Yes) Show each step in the calculation.	5	**CO3**	L3

	Color	Legs	Height	Smelly	Species
0	White	3	Short	Yes	M
1	Green	2	Tall	No	M
2	Green	3	Short	Yes	M
3	White	3	Short	Yes	M
4	Green	2	Short	No	H
5	White	2	Tall	No	H
6	White	2	Tall	No	H
7	White	2	Short	Yes	H

<table>
<tr>
<td></td>
<td>
NB Color Leg Height Smelly Species.   Classify
0 White ① 3 Short Yes M ①   Color = white, legs = 2,
1 Green 2 Tall No M ②   Height Short, Smelly = yes.
2 Green 3 Short Yes M ③   classification – H
3 White ② 3 Short Yes M ④
4 Green 2 Short No H ①
5 White ① 2 Tall No H ②
6 White ② 2 Tall No H ③
7 White ③ 2 Short Yes H 4

$v_{NB} = \underset{v_j \in V}{\arg\max}\, P(v_j) \prod_i P(a_i|v_j)$

$P(M) = \frac{4}{8} = 0.5$ $P(H) = 0.5$
$P(white|M) = 2/4 = 0.5$ $P(white|H) = 3/4 = 0.75$
$P(2|M) = 1/4 = 0.25$ $P(2|H) = 4/4 = 1$
$P(short|M) = 3/4 = 0.75$ $P(short|H) = 2/4 = 0.5$
$P(yes|M) = 3/4 = 0.75$ $P(yes|H) = 1/4 = 0.25$
$0.5 \times P(green) \times \prod_i P(a_i|M)$ $P(H) \times \prod_i (P(a_i|H))$
$= 0.5 \times 0.5 \times 0.25 \times 0.75 \times 0.75 = 0.035$ $0.5 \times 0.75 \times 1 \times 0.5 \times 0.25$
max (H) $= 0.0469$.
</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b)</td>
<td>
Explain Gibbs Algorithm
Why is it justified as compared to the Bayes optimal classifier?

Gibbs Algorithm.
- A less optimal method proposed by Opper & Haussler, '91.
- Bayes Optimal classifier is costly to apply
→ because posterior probability needs to be calculated for every hypothesis

Gibbs Algorithm
1. Choose a hypothesis h from H at random, according to the posterior probability distribution over H.
2. Use h to predict the classification of the next instance x.

- Given a new instance, Gibbs algorithm applies a hypothesis drawn at random according to the current posterior probability distribution.

- It can be shown that, under certain conditions, the expected misclassification error for Gibbs algorithm is at most twice the expected error of the Bayes Optimal Classifier.
- Implication for concept learning problem:
If the learner assumes a uniform prior over H and if target concepts are drawn from such a distribution then, classifying based on Gibbs will have expected error at most twice that of Bayes optimal classifier.
</td>
<td>5</td>
<td>CO2</td>
<td>L2</td>
</tr>
</table>

CI **CCI** **HOD**

Course Outcomes		Blooms Level	Modules covered	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4	
CO1	Appraise the theory of Artificial intelligence and Machine Learning.	**L2**	**1,2**	3	3	2	2	0	2	2	0	0	0	0	0	0	2	0	3	
CO2	Illustrate the working of AI and ML Algorithms.	**L3**	**2,3,4**	3	3	3	3	3	3	0	0	0	0	0	0	0	2	0	3	
CO3	Demonstrate the applications of AI and ML.	**L2**	**4,5**	3	3	3	3	3	3	0	0	0	0	0	0	0	2	0	3	

CO PO Mapping

COGNITIVE LEVEL	REVISED BLOOMS TAXONOMY KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend
L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.
L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PROGRAM OUTCOMES (PO), PROGRAM SPECIFIC OUTCOMES (PSO)				CORRELATION LEVELS	
PO1	Engineering knowledge	PO7	Environment and sustainability	0	No Correlation
PO2	Problem analysis	PO8	Ethics	1	Slight/Low
PO3	Design/development of solutions	PO9	Individual and team work	2	Moderate/ Medium
PO4	Conduct investigations of complex problems	PO10	Communication	3	Substantial/ High
PO5	Modern tool usage	PO11	Project management and finance		

PO6	The Engineer and society	PO12	Life-long learning	
PSO1	Develop applications using different stacks of web and programming technologies			
PSO2	Design and develop secure, parallel, distributed, networked, and digital systems			
PSO3	Apply software engineering methods to design, develop, test and manage software systems.			
PSO4	Develop intelligent applications for business and industry			