NATIONAL UNIVERSITY OF SINGAPORE

IT5005 – ARTIFICIAL INTELLIGENCE

Final Assessment

(Semester 1: AY2024/25)

Time Allowed: 2 Hours

INSTRUCTIONS

1. This assessment paper contains **SIX (06)** questions comprises **SIX (06)** printed pages.
2. This is a **CLOSED BOOK** assessment.
3. Only an A4 cheat sheet is allowed. Apart from calculators, electronic devices are not allowed.
4. Answer **ALL** questions and write your answers only on the **ANSWER SHEET** provided.
5. Do **not** write your name on the ANSWER SHEET.
6. The maximum mark of this assessment is 100.
7. Pseudocodes are provided in the Appendix (Page 6) at the end.

Question	**Max. mark**
Q1	15
Q2	15
Q3	15
Q4	15
Q5	20
Q6	20
Total	**100**

——— **END OF INSTRUCTIONS** ———

There are 6 questions [Total: 100 marks]

1. [Total: 15 marks] Propositional Logic.

Using logic, you need to determine whether the following statement is true: “The vase was broken”. The following clues are given:

R1. Charlie was outside.

R2. The vase was broken if and only if the cat was in the house or Bob was playing indoors.

R3. If Bob was playing indoors, then Charlie was outside.

R4. If Charlie was outside, then cat was in the house.

Use the following propositional symbols to represent the sentences in the puzzle.

O: Charlie was outside.	**B**: The vase was broken.
H: The cat was in the house.	**I**: Bob was playing indoors.

Answer the following questions:

a) Translate the sentences into propositional form. [**4 marks**]

b) Convert the sentences to CNF. [**4 marks**]

c) Check whether the statement “The vase was broken” is true or not using resolution-refutation algorithm. [**7 marks**]

2. [Total: 15 marks] Bayesian Networks

a) Consider the Bayesian network shown in Figure 1. Identify whether the following statements are true or not? [**5 marks**]

i. $C \perp G$ [2.5 marks]

ii. $C \perp E$. [2.5 marks]

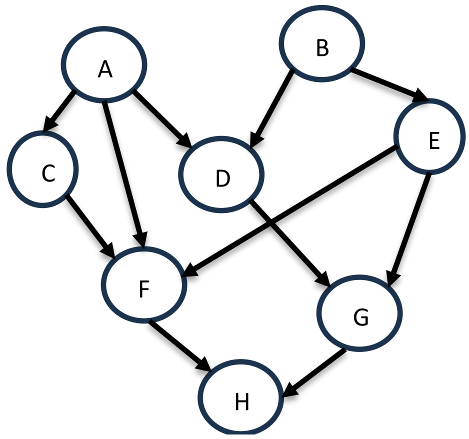
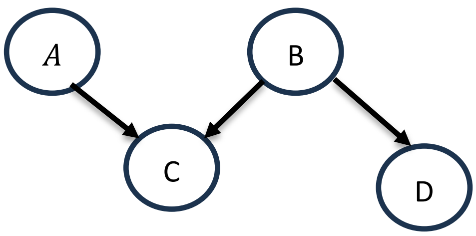


Figure 1. Bayes Network 1

b) A Bayesian network with Boolean random variables is shown in Figure 2. The variable A takes both values in its domain with equal probability, while the variable B is true only 25% of time. The conditional probabilities of the variables C and D are shown in the figure. [**10 marks**]

A	B	P(C=c|A,B)
t	t	0.6
t	f	0.1
f	t	0.8
f	f	0.3



B	P(D=d|B)
t	0.8
f	0.4

Figure 2 Bayesian Network 2

i. Factorize the joint probability distribution of the Bayesian network? [2 marks]

ii. What is the probability that all variables are true? [4 marks]

iii. Evaluate $P(a, b)$? [4 marks]

3. [Total: 15 marks] Hidden Markov Models

Aiken monitors a restricted area to prevent unauthorized entry. If Aiken was on high alert (H) yesterday, he is less likely to remain on high alert today due to fatigue. Consequently, the probability that he is on high alert today drops to 0.4. Conversely, if Aiken was not on high alert yesterday, he is on high alert today with a probability of 0.1. The probability of detecting an intrusion (I) given that Aiken is on high alert is 0.8. On the other hand, the probability of detecting an intrusion given that Aiken is not on high alert is 0.1. You learn that on Monday and Tuesday, Aiken detects an intrusion. Moreover, on Sundays Aiken is on high alert.

Assume that the variable A_t indicate the level of alertness on day t and the variable D_t indicates that an intruder is detected on day t. Both A_t and D_t are binary random variables.

a. Identify the hidden and observation variables and their values. [**2 marks**]
b. Provide the state transition matrix for the hidden variables. [**4 marks**]
c. Provide the observation matrices for this problem. [**4 marks**]
d. What is the probability distribution of hidden variable on Tuesday? No need to evaluate the expression. Closed-form formula with relevant values assigned to variables is enough. [**5 marks**]

4. [Total: 15 marks] Markov Decision Process

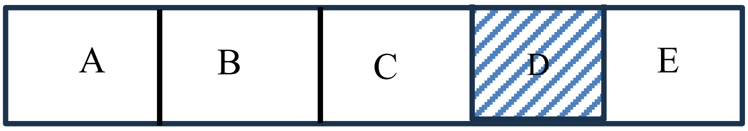


Figure 3 Grid World

Table 1: State and Actions

State (s)	$Actions(s)$
A	$right$
B	$left, right$
C	$left, jump$
D	$right$

Consider the grid world shown in Figure 3. State E is the terminal state, and the shaded region corresponds to state D. The set of actions that agent can take in each state are shown in the Table 1. Action $right$ at state A takes the agent to state B with probability 0.7 and the agent stays in the same state with probability 0.3. Action $left$ at state B takes the agent to state A with probability 1 and action $right$ at state B takes the agent to state C with probability 1. At state C, the action left takes the agent to state B with probability 1 and action jump takes the agent to state E with probability 1. At state D, the action right takes the agent to state E with probability 1. The agent collects a reward of +1 for transition to state E. For all other transitions, the agent collects a reward of -1. Answer the following questions for this MDP. Assume that the discount factor is 1.

a. Write down the Bellman update equations. [**8 marks**]

b. Assuming that utility of terminal state is 1, find the utilities of the states for the policy π, where

$$\pi(A) = right,\ \pi(B) = right,\ \pi(C) = jump \text{ and } \pi(D) = right.$$ [**7 marks**]

5. **[Total: 20 marks] Introduction to Learning.**

Table 2: Dataset 1

Data Index	Feature A	Feature B	Label
1	1	0	-1
2	1	1	-1
3	1	2	1
4	2	1	1

a. Consider the dataset shown in Table 2. Mr. Aiken would like to build a linear classification model. To this end, he initialized the weight vector as $\boldsymbol{w} = [0, 1, 0]^T$. He is keen on solving this problem by hand. Assume that Aiken is using perceptron learning algorithm with a learning rate of 0.1. You need to help him in getting started by answering the following questions: [**12 marks**]

   i. Using the initial weight vector, find the predicted label for each data point. [4 marks]
   ii. Do the weights need update? If yes, perform one iteration of the weight update. [5 marks]
   iii. Could Rosenblat's PLA converge to a solution? Provide the rationale. [3 marks]

Table 3: Dataset 2

Data Index	House Size (square meters)	Price ($K)
1	50	150
2	60	180
3	70	209
4	80	241

b. Consider the dataset shown in Table 3. You are asked to design a model to predict the price of the house using linear regression. [**8 marks**]

   i. Which loss function would you prefer: mean square error or mean absolute error? Provide the rationale. [2 marks]
   ii. Without explicit learning, identify the linear regression model for this dataset? [3 marks]
   iii. Using the model from b(ii), predict the price of house with a size of 75 square meters. [3 marks]

6 **[Total: 20 marks] Neural Networks and Deep Learning**

a. Explain in one sentence the difference between stochastic gradient descent and mini-batch gradient descent algorithms. [**2 marks**]

b. Aiken has designed a neural network to perform regression. Answer the following queries to help him in designing the network. [**4 marks**]

   i. Which activation should Aiken use: sigmoid function or ReLU, given that the target is a continuous variable in the range of 0 to 100? [2 marks]
   ii. Aiken also working on optimizing the weights of the neural network with a custom loss function. Aiken notices that his loss function includes plateaus—regions where the gradient is zero, but these areas do not represent the minimum loss. Aiken is considering variants of the gradient descent algorithm to overcome this issue. Which variant of the gradient descent algorithm would Aiken need to use for this scenario: gradient descent, RMSProp, or the Adam algorithm? Justify your recommendation. [2 marks]

c. Consider a two-layer neural network with single perceptron in each layer and assume that bias term $b = 0$. The weights for the first and second layer are initialized as 2 and 3, respectively, i.e., $w^{[1]} = 2$ and $w^{[2]} = 3$. Furthermore, an exponential linear unit (ELU) is used as the activation function at each perceptron. The ELU is defined as:

$$g(x) = \begin{cases} x & if\ x \geq 0 \\ \alpha(\exp(x) - 1) & else \end{cases}$$

The derivative of ELU is defined as:

$$g'(x) = \begin{cases} 1 & if\ x \geq 0 \\ \alpha \exp(x) & else \end{cases}$$

For ELU, assume that the parameter $\alpha = 1$. The input to the neural network is $x = -1$ and the expected output is $y = -1.4255$. [**14 marks**]

i. Calculate $a^{[1]}$ and $a^{[2]}$. [4 marks]

ii. Calculate the mean square error, i.e., $(\hat{y} - y)^2$, where $\hat{y}$ is the output of neural network. [2 marks]

iii. Calculate $\frac{\partial L}{\partial w^{[2]}}$ and $\frac{\partial L}{\partial w^{[1]}}$. [8 marks]

Appendix:

Logical Equivalences

$$
\begin{aligned}
(\alpha \wedge \beta) &\equiv (\beta \wedge \alpha) && \text{commutativity of } \wedge \\
(\alpha \vee \beta) &\equiv (\beta \vee \alpha) && \text{commutativity of } \vee \\
((\alpha \wedge \beta) \wedge \gamma) &\equiv (\alpha \wedge (\beta \wedge \gamma)) && \text{associativity of } \wedge \\
((\alpha \vee \beta) \vee \gamma) &\equiv (\alpha \vee (\beta \vee \gamma)) && \text{associativity of } \vee \\
\neg(\neg\alpha) &\equiv \alpha && \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha) && \text{contraposition} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \vee \beta) && \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) && \text{biconditional elimination} \\
\neg(\alpha \wedge \beta) &\equiv (\neg\alpha \vee \neg\beta) && \text{De Morgan} \\
\neg(\alpha \vee \beta) &\equiv (\neg\alpha \wedge \neg\beta) && \text{De Morgan} \\
(\alpha \wedge (\beta \vee \gamma)) &\equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) && \text{distributivity of } \wedge \text{ over } \vee \\
(\alpha \vee (\beta \wedge \gamma)) &\equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) && \text{distributivity of } \vee \text{ over } \wedge
\end{aligned}
$$

Resolution-Refutation Algorithm:

function PL-RESOLUTION(KB, α) **returns** *true* or *false*
  inputs: KB, the knowledge base, a sentence in propositional logic
    α, the query, a sentence in propositional logic

  $clauses \leftarrow$ the set of clauses in the CNF representation of $KB \wedge \neg\alpha$
  $new \leftarrow \{\}$
  while *true* **do**
    for each pair of clauses C_i, C_j **in** *clauses* **do**
      $resolvents \leftarrow$ PL-RESOLVE(C_i, C_j)
      if *resolvents* contains the empty clause **then return** *true*
      $new \leftarrow new \cup resolvents$
    if $new \subseteq clauses$ **then return** *false*
    $clauses \leftarrow clauses \cup new$

Perceptron Learning Algorithm:

1. Initialize weights w_i
2. For each instance i with features $x^{(i)}$, classify $\hat{y}^{(i)} = sgn(w^T x^{(i)})$
3. Select one misclassified instance $x^{(j)}$ and update weights as follows:
$$w \leftarrow w + \alpha(y^{(j)} - \hat{y}^{(j)})x^{(j)}$$
4. Iterate steps 2 to 3 until
    - Convergence (classification error < threshold), or
    - Maximum number of iterations

=== END OF PAPER ===