NATIONAL UNIVERSITY OF SINGAPORE

CS3244 - MACHINE LEARNING

(Semester 2: AY2019/20)

Time Allowed: 1 Hour 30 Minutes

INSTRUCTIONS TO CANDIDATES

1. This assessment paper contains **FOUR (4)** parts and comprises **FOURTEEN (14)** printed pages, including this page.
2. Answer **ALL** questions as indicated.
3. This is a **OPEN BOOK** assessment.
4. You are allowed to use **NUS APPROVED CALCULATORS**.
5. Please write your **Student Number** below. Do not write your name.

STUDENT NUMBER: ______________________________

☐ By ticking this box, I declare that I will adhere to the NUS code of student conduct (`http://nus.edu.sg/osa/resources/codeofstudentconduct`), will not cheat, will comply with all other rules for the assessments, and am aware that failure to comply may result in disciplinary action against them.

EXAMINER'S USE ONLY		
Part	Mark	Score
I	10	
II	8	
III	12	
IV	10	
TOTAL	40	

In Part I, II, III, and IV, you will find a series of structured questions. For each structured question, give your answer in the reserved space in the script.

Part I
Concept Learning 1

(10 points) Structured questions. Answer in the space provided on the script.

1. (10 points) Let $Z = \{0, 1, \ldots, 10\}$. Consider the input instance space $X = \{(x_1, x_2)\}_{x_1, x_2 \in Z}$ consisting of integer points in the x_1, x_2 plane, and the hypothesis space H such that each hypothesis $h \in H$ is defined as

$$h(x_1, x_2) = \begin{cases} 1 & \text{if } a \leq x_1 \leq b \text{ and } c \leq x_2 \leq d, \\ 0 & \text{otherwise;} \end{cases}$$

where $a, b, c, d \in Z$. We represent hypothesis h in the form (a, b, c, d). For example, a typical hypothesis in H is $(3, 5, 2, 9)$. Note that for any $h = (a, b, c, d) \in H$, if $a > b$ or $c > d$, then **no** input instance $(x_1, x_2) \in X$ satisfies h.

Let hypotheses h and h' be in the same hypothesis space H, i.e., $h, h' \in H$. We know that a hypothesis represents a set of input instances in X such that every input instance in this set satisfies this hypothesis.

Let us define a **new hypothesis space H'** that consists of all **differences** of the hypotheses in H. The **difference** $h \setminus h'$ of the hypotheses h and h' is defined as $h \setminus h'(x) = ((h(x) = 1) \wedge (h'(x) = 0))$ for all $x \in X$ and therefore represents the set difference of the sets of input instances represented by h and h'. For example, a typical hypothesis in H' is $(3, 5, 2, 9) \setminus (4, 4, 4, 6)$.

Trace the CANDIDATE-ELIMINATION algorithm (reproduced below in Fig. 1) for the **hypothesis space H'** given the sequence of positive ($c(x_1, x_2) = 1$) and negative ($c(x_1, x_2) = 0$) training examples from Table 1 below (i.e., show the sequence of S and G boundary sets). You only need to show the **semantically distinct**[1] hypotheses in each boundary set.

1. $G \leftarrow$ maximally general hypotheses in H
2. $S \leftarrow$ maximally specific hypotheses in H
3. For each training example d
   - If d is a positive example
     - Remove from G any hypothesis inconsistent with d
     - For each $s \in S$ not consistent with d
       - Remove s from S
       - Add to S all minimal generalizations h of s s.t.
         h is consistent with d, and
         some member of G is more general than h
       - Remove from S any hypothesis that is more general than another hypothesis in S
   - If d is a negative example
     - Remove from S any hypothesis inconsistent with d
     - For each $g \in G$ not consistent with d
       - Remove g from G
       - Add to G all minimal specializations h of g s.t.
         h is consistent with d, and
         some member of S is more specific than h
       - Remove from G any hypothesis that is more specific than another hypothesis in G

Figure 1: CANDIDATE-ELIMINATION algorithm.

[1] The notion of **semantically distinct** hypotheses was mentioned and explained informally on page 10 of the "Concept Learning" lecture slides. For a formal definition, the hypotheses h and h' are **semantically distinct** iff there exists some $x \in X$ satisfying $h \setminus h'$ or $h' \setminus h$, that is, $\exists x \in X \ \ (h \setminus h'(x) = 1) \vee (h' \setminus h(x) = 1)$.

Example	Input Instance		Target Concept
	x_1	x_2	$c(x_1, x_2)$
1	6	3	1
2	8	7	0
3	4	7	1
4	2	1	0
5	3	9	0

Table 1: Positive ($c(x_1, x_2) = 1$) and negative ($c(x_1, x_2) = 0$) training examples for target concept c.

Solution:

$G_0 = \{(0, 10, 0, 10) \setminus (6, 5, 3, 2), (0, 10, 0, 10) \setminus (10, 0, 10, 0), \ldots\} = \{(0, 10, 0, 10)\}$

$S_0 = \{(6, 5, 3, 2) \setminus (0, 10, 0, 10), (10, 0, 10, 0) \setminus (0, 10, 0, 10), \ldots\} = \{(6, 5, 3, 2)\}$

The above hypotheses in G_0 and S_0 are **not semantically distinct.**

$G_1 =$

$S_1 =$

$S_2 =$

$G_2 =$

$G_3 =$

$S_3 =$

$S_4 =$

$G_4 =$

Solution:

$$S_5 =$$

$$G_5 =$$

Part II
Neural Networks 1

(8 points) Structured questions. Answer in the space provided on the script.

1. (3 points) Fig. 2a below shows a network A of perceptron units with a hidden layer of two units, while Fig. 2b below shows a perceptron unit B. They are based on the following structure:

   - Network A of perceptron units and perceptron unit B have one (Boolean) output unit each for producing the output o_A and the output o_B, respectively.
   - There should be two input units (i.e., one input unit for each of the two (Boolean) input attributes x_1, x_2).
   - A Boolean is **–1 if false**, and **1** if true.
   - The activation function of every (non-input) unit is a **–1 to 1 step function** (refer to page 6 of the "Neural Networks" lecture slides), including that of the output unit.
   - The weights (i.e., hypothesis) of network A of perceptron units and the weights (i.e., hypothesis) of perceptron unit B are indicated in Fig. 2a and Fig. 2b, respectively.
   - A bias input is of value 1 and is not considered a hidden unit.

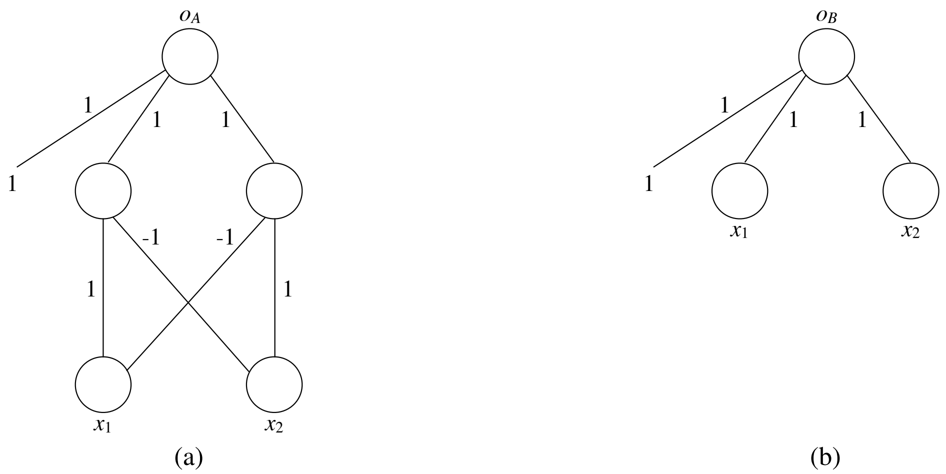


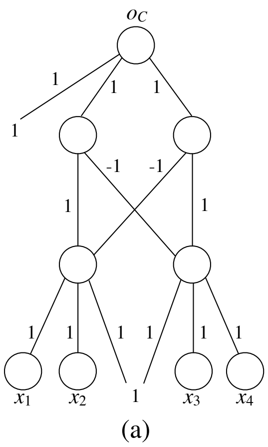
Figure 2: Perceptron networks: (a) network A of perceptron units, and (b) perceptron unit B.

Prove formally or disprove that the network A of perceptron units is *more general than or equal to* perceptron unit B.

Solution:

2. (5 points) Fig. 3a below shows a network C of perceptron units with two hidden layers of two units each, while Fig. 3b below shows a network F of perceptron units with a hidden layer of two units. They are based on the following structure:

   - Networks C and F of perceptron units have one (Boolean) output unit each for producing the output o_C and the output o_F, respectively.
   - There should be four input units (i.e., one input unit for each of the four (Boolean) input attributes x_1, x_2, x_3, x_4).
   - A Boolean is **–1 if false**, and **1** if true.
   - The activation function of every (non-input) unit is a **–1 to 1 step function** (refer to page 6 of the "Neural Networks" lecture slides), including that of the output unit.
   - The weights (i.e., hypothesis) of networks C and F of perceptron units are indicated in Fig. 3a and Fig. 3b, respectively.
   - A bias input is of value 1 and is not considered a hidden unit.



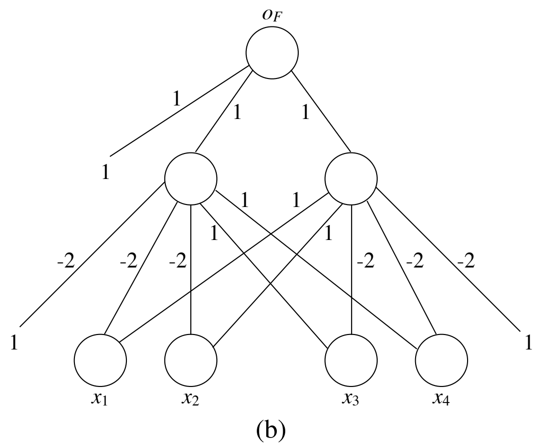


Figure 3: Perceptron networks: (a) network C of perceptron units, and (b) network F of perceptron units.

Prove formally or disprove that the network C of perceptron units is *more general than or equal to* network F of perceptron units.

Solution:

Solution:

Part III
Bayesian Inference

(12 points) Structured questions. Answer in the space provided on the script.

Fig. 4a and Fig. 4b below show perceptron units A and B. Fig. 4c below shows a network C of perceptron units with two hidden layers of two units each (i.e., same as that in Fig. 3a), while Fig. 4d below shows a network F of perceptron units with a hidden layer of two units (i.e., same as that in Fig. 3b). They are based on the following structure:

- Perceptron units A and B have one (Boolean) output unit each for producing the output o_A and the output o_B, respectively. Similarly, networks C and F of perceptron units have one (Boolean) output unit each for producing the output o_C and the output o_F, respectively.
- There should be four input units (i.e., one input unit for each of the four (Boolean) input attributes x_1, x_2, x_3, x_4).
- A Boolean is **–1 if false**, and **1** if true.
- The activation function of every (non-input) unit is a **–1 to 1 step function** (refer to page 6 of the "Neural Networks" lecture slides), including that of the output unit.
- The weights $\mathbf{w}_A$ (i.e., hypothesis) of perceptron unit A and the weights $\mathbf{w}_B$ (i.e., hypothesis) of perceptron unit B are indicated in Fig. 4a and Fig. 4b, respectively. The weights $\mathbf{w}_C$ (i.e., hypothesis) of network C of perceptron units and the weights $\mathbf{w}_F$ (i.e., hypothesis) of network F of perceptron units are indicated in Fig. 4c and Fig. 4d, respectively.
- A bias input is of value 1 and is not considered a hidden unit.

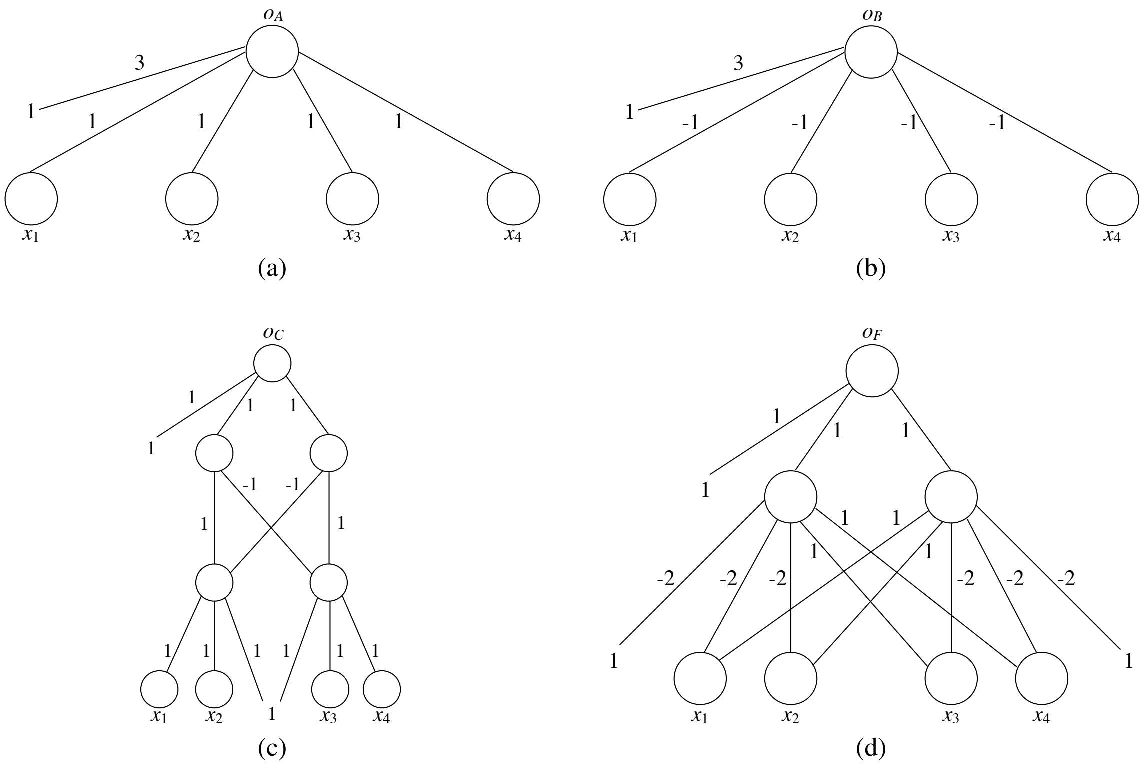


Figure 4: Perceptron networks: (a) perceptron unit A, (b) perceptron unit B, (c) network C of perceptron units, and (d) network F of perceptron units.

1. (6 points) One of the four perceptron networks in Fig. 4 above has been used to generate a dataset of 4 Boolean input attributes x_1, x_2, x_3, x_4 and a Boolean target output t_d with the following 3 noise-free training examples of the form $d = \langle (x_1, x_2, x_3, x_4), t_d \rangle$:

$$D = \{d_1 = \langle (-1, -1, -1, 1), 1 \rangle, d_2 = \langle (1, 1, -1, -1), 1 \rangle, d_3 = \langle (1, 1, 1, 1), -1 \rangle\}.$$

Suppose that the **prior beliefs** of hypotheses/weights $\mathbf{w}_A$, $\mathbf{w}_B$, $\mathbf{w}_C$, and $\mathbf{w}_F$ are equal and they sum to 1.

Using Bayes' Theorem, calculate the posterior beliefs $P(\mathbf{w}_A|D)$, $P(\mathbf{w}_B|D)$, $P(\mathbf{w}_C|D)$, and $P(\mathbf{w}_F|D)$. Show the steps of your derivation. **No marks will be awarded for not doing so.**

We assume that the input instances $\mathbf{x}_d = (x_1, x_2, x_3, x_4)$ for $d \in D$ are fixed. Therefore, in deriving an expression for $P(D|\mathbf{w}_A)$, $P(D|\mathbf{w}_B)$, $P(D|\mathbf{w}_C)$, or $P(D|\mathbf{w}_F)$, we only need to consider the probability of observing the target outputs t_d for $d \in D$ for these fixed input instances $\mathbf{x}_d$ for $d \in D$.

Furthermore, we assume that the training examples are conditionally independent given the hypothesis/weights of any perceptron network in Fig. 4 above.

Solution:

Using the posterior beliefs calculated above, compute the **Bayes-optimal classification** for the new input instance $\mathbf{x}_{d_4} = (-1, -1, -1, -1)$. Show the steps of your derivation. **No marks will be awarded for not doing so.**

Solution:

2. (6 points) One of the four perceptron networks in Fig. 4 above has been used to generate another dataset of 4 Boolean input attributes x_1, x_2, x_3, x_4 and a Boolean target output t_d with the following 3 noise-free training examples of the form $d = \langle (x_1, x_2, x_3, x_4), t_d \rangle$:

$$D' = \{d_1 = \langle (-1, 1, -1, -1), 1 \rangle, d_2 = \langle (1, -1, -1, 1), 1 \rangle, d_3 = \langle (-1, -1, -1, -1), -1 \rangle\}.$$

Suppose that the prior beliefs of hypotheses/weights $\mathbf{w}_A$, $\mathbf{w}_B$, $\mathbf{w}_C$, and $\mathbf{w}_F$ are equal and they sum to 1.

Using Bayes' Theorem, calculate the posterior beliefs $P(\mathbf{w}_A|D')$, $P(\mathbf{w}_B|D')$, $P(\mathbf{w}_C|D')$, and $P(\mathbf{w}_F|D')$. Show the steps of your derivation. **No marks will be awarded for not doing so.**

Similar to question 1, we assume that the input instances $\mathbf{x}_d = (x_1, x_2, x_3, x_4)$ for $d \in D'$ are fixed. We also assume that the training examples are conditionally independent given the hypothesis/weights of any perceptron network in Fig. 4 above.

Solution:

Solution:

Using the posterior beliefs calculated above, compute the **Bayes-optimal classification** for the new input instance $\mathbf{x}_{d_4} = (-1, -1, -1, -1)$. Show the steps of your derivation. **No marks will be awarded for not doing so.**

Solution:

Part IV
Neural Networks 2

(10 points) Structured questions. Answer in the space provided on the script.

1. (10 points) Cara has constructed a dataset of 4 Boolean input attributes x_1, x_2, x_3, x_4 and 4 Boolean target outputs $t_{k_1}, t_{k_2}, t_{k_3}, t_{k_4}$ with the following 4 training examples of the form $d = \langle (x_1, x_2, x_3, x_4), (t_{k_1}, t_{k_2}, t_{k_3}, t_{k_4}) \rangle$:

$$D = \{d_1 = \langle (1,0,0,0), (1,0,0,0) \rangle, d_2 = \langle (0,1,0,0), (0,1,0,0) \rangle,$$
$$d_3 = \langle (0,0,1,0), (0,0,1,0) \rangle, d_4 = \langle (0,0,0,1), (0,0,0,1) \rangle \}.$$

Consider the network of perceptron units in Fig. 5 below with **only a hidden layer of one unit** based on the following **constraints**:

- There should be **only four (Boolean) output units** k_1, k_2, k_3, k_4 and **four input units** (i.e., one input unit for each of the four (Boolean) input attributes x_1, x_2, x_3, x_4).
- A Boolean is **0 if false**, and **1** if true.
- The activation function $\sigma(\cdot)$ of every (non-input) unit is a **0 to 1 step function**, including that of the output unit. That is,

$$\sigma(s) = \begin{cases} 1 & \text{if } s > 0, \\ 0 & \text{otherwise.} \end{cases}$$

  This is **in contrast** to the activation function on page 6 of the "Neural Networks" lecture slides that is a -1 to 1 step function.
- **The weights $w_0, w_1, \ldots, w_{12}$ (i.e., hypothesis) must be consistent with D and must take on one of the following values: –1, 0, 1.**
- A bias input is of value 1 and is not considered a hidden unit.

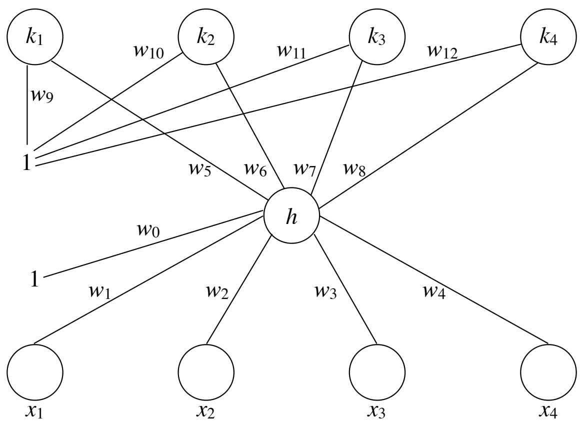


Figure 5: Network of perceptron units.

Prove formally or disprove that no such network of perceptron units in Fig. 5 can be consistent with D.

Solution:

Solution:

Solution:

END OF PAPER