Indian Institute of Technology Kharagpur
Department of Computer Science and Engineering

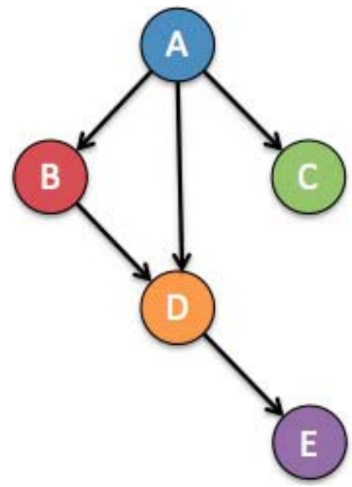
CS60050 : Machine Learning

Spring 2021 | Long Test 1 (ALL Q&A) | Marks : 50

Date : 22-Feb-2021 (Monday) | Time : 8:30pm-9:45pm (75 min)

Question-1: [Bayesian Network] [Marks: 2 + 4 + 4 = 10]

You are given with the following Bayesian Network with 5 nodes (marked using circle) and 5 directed edges (marked using arrows), as shown in the following figure.



The Probability table is given for each node as follows:

P(A=ON) = 0.60,
P(B=ON | A=OFF) = 0.10 , P(B=ON | A=ON) = 0.95,
P(C=ON | A=OFF) = 0.80 , P(C=ON | A=ON) = 0.50,
P(D=ON | A=OFF, B=OFF) = 0.10 , P(D=ON | A=ON, B=OFF) = 0.90,
P(D=ON | A=OFF, B=ON) = 0.30 , P(D=ON | A=ON, B=ON) = 0.95,
P(E=ON | D=OFF) = 0.80 , P(E=ON | D=ON) = 0.10,

Calculate the Probability of the following:
(Write only the numeric values as answer.)

(i) P(A=ON, B=ON, C=ON, D=ON, E=ON) = ? [2 marks]

(ii) P(E=ON | A=ON) = ? [4 marks]

(iii) P(A=ON | E=ON) = ? [4 marks]

Answers:
(i) 0.027075
(ii) 0.13675
(iii) 0.22269

Explanations:

(i) P(A=ON, B=ON, C=ON, D=ON, E=ON) = P(A=ON) x P(B=ON | A=ON) x P(C=ON | A=ON) x P(D=ON | A=ON, B=ON) x P (E=ON | D=ON)
= 0.6 x 0.95 x 0.5 x 0.95 x 0.1 = 0.027075

(ii) As B,D,E is conditionally independent of C, given A -- hence C drops out from calculation. Therefore, we sum over the 4 {B,D} possibilities:

P(E=ON | A=ON)
= Σ P(E=ON, B,D | A=ON) [i.e. sum over all values of B,D = {ON,OFF}]
= Σ P(E=ON | D) x P(D | A=ON, B) x P(B | A=ON)

B	D	P(B | A=ON)	P(D | A=ON, B)	P(E=ON | D)	P(E=ON, B,D | A=ON)
ON	ON	0.95	0.95	0.10	0.09025
ON	OFF	0.95	0.05	0.80	0.03800
OFF	ON	0.05	0.90	0.10	0.00450
OFF	OFF	0.05	0.10	0.80	0.00400

Summing over the last column, we obtain P(E=ON | A=ON) = 0.13675

(iii) Applying Bayes' Rule, we get --

$$P(A=ON \mid E=ON) = \frac{P(E=ON \mid A=ON) \times P(A=ON)}{P(E=ON)}$$

$$= \frac{P(E=ON \mid A=ON) \times P(A=ON)}{P(E=ON \mid A=ON) \times P(A=ON) + P(E=ON \mid A=OFF) \times P(A=OFF)}$$

We already have [from (ii)], P(E=ON | A=ON) = 0.13675
Similarly, we compute P(E=ON | A=OFF) as follows:

B	D	P(B |A=OFF)	P(D |A=OFF, B)	P(E=ON | D)	P(E=ON, B,D |A=OFF)
ON	ON	0.10	0.30	0.10	0.003
ON	OFF	0.10	0.70	0.80	0.056
OFF	ON	0.90	0.10	0.10	0.009
OFF	OFF	0.90	0.90	0.80	0.648

Summing over the last column, we obtain P(E=ON | A=OFF) = 0.716.

$$\text{Therefore, } P(A = ON|E = ON) = \frac{0.13675 \times 0.6}{0.13675 \times 0.6 + 0.716 \times 0.4} = 0.22269$$

===

===
Question-2: [Linear Regression] [Marks: 2 + 2 + 1 = 5]
===
Suppose you are given with the following three points in 2-D a (x,y)-plane, with respect to attribute x and outcome/value y:
Point-1: (0,2) ; Point-2: (1,2) ; Point-3: (1,8).

Answer the following questions:

(i) If you use linear regression to fit the best line/hypothesis with respect to x, i.e. h(x) = Mx + C, using the process of sum-squared error (SSE) minimization, what will be the values of M and C? Write only the numeric values as answer. [2 marks]

(ii) What is the total sum of squared errors (SSE)? Write only a numeric value as answer. [2 marks]

(iii) For a new point (3,10), what is the absolute value of the error/deviation in prediction with the obtained best-fit line? Write only a numeric value as answer. [1 mark]
~~~~~~
Answers:
(i) M = 3 and C = 2
(ii) 18
(iii) 1
~~~~~~
Explanations:
(i) h(x) = 3x + 2 is the best fit line/hypothesis as it minimized the SSE.

(ii) The SSE is computed by summing the squared errors between the actual values and our predictions. For each value of the independent variable (x), our best-fit line makes the following predictions:
-- If x = 0, then y = 3 x 0 + 2 = 2,
-- If x = 1, then y = 3 x 1 + 2 = 5.
Thus we make an error of 0 for the data point (0,2), an error of 3 for the data point (1,2), and an error of 3 for the data point (1,8). So we have SSE = $0^2 + 3^2 + 3^2 = 18$.

(iii) If x = 3, then y = 3 x 3 + 2 = 11 according to the best-fit line. So, the absolute value of the error/deviation is = 1 for the point (3,10).

==

==

Question-3: [Logistic Regression] [Marks: 2 + 2 + 1 = 5]

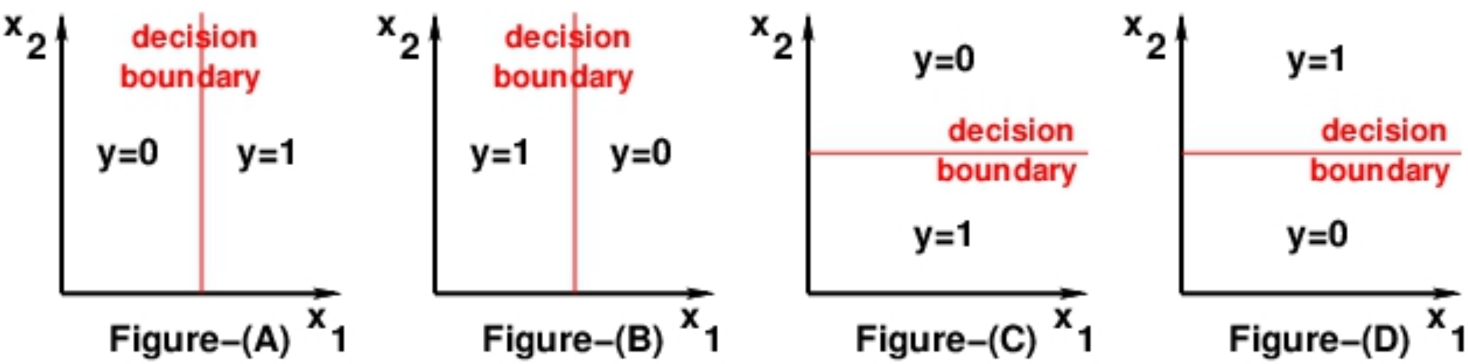
==

In an analytical problem with two attributes, x1 and x2, suppose you have used 'logistic regression' where the hypothesis (output) will be modelled as, h(x1,x2) = y = θ(w0 + w1.x1 + w2.x2), where θ(s) is the logistic (sigmoid) function applied over the summation, s = w0 + w1.x1 + w2.x2.

Answer the following questions:

(i) Let, after a rigorous training phase, you found that your model converges with the following parameters: w0 = -1.5, w1 = 3, and w2 = -0.5
For the independent variables, x1 and x2, if we observe our trained model with the following test-data values, x1 = 1 and x2 = 5, what will be the value of Prob(y=1) with this observed test-data? Write only a numeric value as answer. [2 marks]

(ii) Alternatively (suppose for a separate training data-set), had your model converges with the following parameters: w0 = 6, w1 = 0, and w2 = -1, then which one among the following figures will represent the decision boundary as given by your classifier/model? [2 marks]



(iii) State whether the following statement is True or False:
"Logistic regression can only form linear decision surface and hence can be applied only to linearly separable data." [1 mark]

~~~~~~

Answers:
(i) 0.2689414
(ii) Figure-(C)
(iii) True

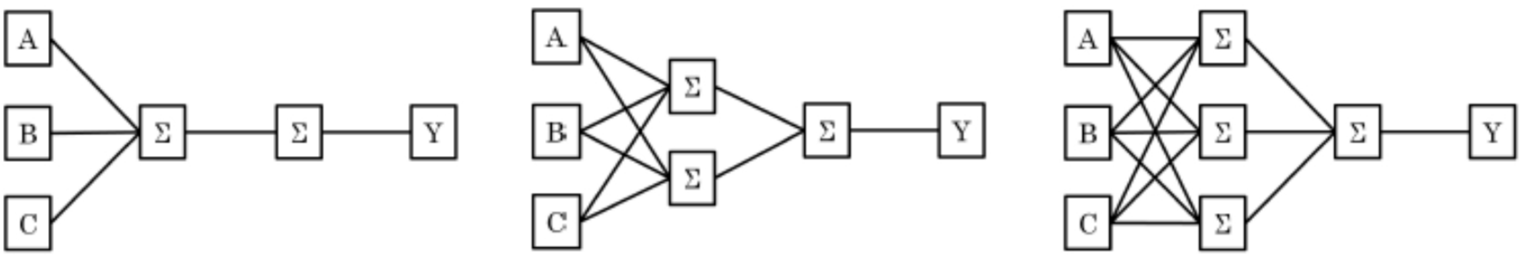
~~~~~~

Explanations:
(i) For x1 = 1 and x2 = 5, sum = -1.5 + 3 x 1 + (-0.5) x 5 = -1
So, using the Logistic Response Function, we can compute that Prob(y = 1) = 1/(1 + e^(-sum)) = 1/(1 + e^(1)) = 0.2689414.

(ii) The decision boundary line will be represented by y = θ(6 - x2) which is shown in the Figure-(C) and Figure-(D). But, Figure-(C) is the right answer because when we put the value x2 = 6 in the equation then y = θ(0) we will get y = 0.5 on the logistic line, and if we increase the value of x2 greater than 6, we will get negative values as sum and hence the output will be the region y = 0.

==

==
Question-4: [Perceptrons]   [Marks: 1 x 5 = 5]
==
Suppose you are given with the following three multi-layer perceptrons, namely PT1, PT2, PT3 (refer to the following figures from left-to-right). Booleans (A, B and C) will take values 0 and 1, and each perceptron will output values 0 and 1. The perceptron unit is denoted using [Σ] which outputs, h(X) = sign(Σ Wi.Xi), as usual. You may assume that each perceptron also has an mandatory input feature (threshold or bias) that always takes the value 1. Connection weights (Wi's) are allowed to take on any values.



In the following questions, select all the neural networks (PT1, PT2, PT3) that can compute the same function as the Boolean expression mentioned. It may help to write out the truth table for each expression.

(i) Boolean Expression = A (BOOLEAN) [1 mark]

(ii) Boolean Expression = A ∧ B (AND) [1 mark]

(iii) Boolean Expression = B ⊕ C (XOR) [1 mark]

(iv) Boolean Expression = A → B (IMPLY) [1 mark]

(v) Boolean Expression = (A ⊕ B) ⊕ C [1 mark]

Choose the correct answer for each of these above five from the following list.
(I) only PT1, but neither PT2, nor PT3
(II) only PT2, but neither PT1, nor PT3
(III) only PT3, but neither PT1, nor PT2
(IV) both PT1 and PT2, but not PT3
(V) both PT1 and PT3, but not PT2
(VI) both PT2 and PT3, but not PT1
(VII) ALL of the PT1, PT2, and PT3
(VIII) NONE of the PT1, PT2, or PT3
~~~~~~
Answers:
(i) (VII)
(ii) (VII)
(iii) (VI)
(iv) (VII)
(v) (III)
~~~~~~
Explanations:
(i) The following weights can realize (A):
PT1: Wt1 = Wt2 = -0.5, Wa = 1, Wb = 0, Wc = 0, Ws1 = 1
PT2: Wt11 = Wt12 = Wt2 = -0.5, Ws1 = Ws2 = 1
      Wa1 = Wa2 = 1, Wb1 = Wb2 = 0, Wc1 = Wc2 = 0
PT3: Wt11 = Wt12 = Wt13 = Wt2 = -0.5, Ws1 = Ws2 = Ws3 = 1,

      Wa1 = Wa2 = Wa3 = 1, Wb1 = Wb2 = Wb3 = 0, Wc1 = Wc2 = Wc3 = 0

(ii) The following weights can realize (A ∧ B):
PT1: Wt1 = -1.5, Wt2 = -0.5, Wa = 1, Wb = 1, Wc = 0, Ws1 = 1
PT2: Wt11 = Wt12 = -1.5, Wt2 = -0.5, Ws1 = Ws2 = 1
      Wa1 = Wa2 = 1, Wb1 = Wb2 = 1, Wc1 = Wc2 = 0
PT3: Wt11 = Wt12 = Wt13 = -1.5, Wt2 = -0.5, Ws1 = Ws2 = Ws3 = 1,
      Wa1 = Wa2 = Wa3 = 1, Wb1 = Wb2 = Wb3 = 1, Wc1 = Wc2 = Wc3 = 0

(iii) The following weights can realize (B ⊕ C):
Here, B ⊕ C = (B ∧ ¬C) V (¬B ∧ C)
PT1: Cannot realize this function!
PT2: Wt11 = Wt12 = Wt2 = -0.5, Ws1 = Ws2 = 1
      Wa1 = Wa2 = 0, Wb1 = 1, Wb2 = -1, Wc1 = -1, Wc2 = 1
PT3: Wt11 = 0, Wt12 = Wt13 = Wt2 = -0.5, Ws1 = 0, Ws2 = Ws3 = 1,
      Wa1 = Wa2 = Wa3 = 0, Wb1 = 0, Wb2 = 1, Wb3 = -1, Wc1 = 0, Wc2 = -1, Wc3 = 1

(iv) The following weights can realize (A → B):
Here, A → B = ¬A V B
PT1: Wt1 = 0.5, Wt2 = -0.5, Wa = -1, Wb = 1, Wc = 0, Ws1 = 1
PT2: Wt11 = Wt12 = 0.5, Wt2 = -0.5, Ws1 = Ws2 = 1
      Wa1 = Wa2 = -1, Wb1 = Wb2 = 1, Wc1 = Wc2 = 0
PT3: Wt11 = Wt12 = Wt13 = 0.5, Wt2 = -0.5, Ws1 = Ws2 = Ws3 = 1,
      Wa1 = Wa2 = Wa3 = -1, Wb1 = Wb2 = Wb3 = 1, Wc1 = Wc2 = Wc3 = 0

(v) The following weights can realize ((A ⊕ B) ⊕ C):
Here, (A ⊕ B) ⊕ C = (¬A ∧ (B ⊕ C)) V (¬B ∧ (A ⊕ C)) V (¬C ∧ (A ⊕ B))
((A ⊕ B) ∧ ¬C) V (¬(A ⊕ B) ∧ C)
    = (((A ∧ ¬B) V (¬A ∧ B)) ∧ ¬C) V (¬((A ∧ ¬B) V (¬A ∧ B)) ∧ C)
    = (A ∧ ¬B ∧ ¬C) V (¬A ∧ B ∧ ¬C) V (¬A ∧ ¬B ∧ C) V (A ∧ B ∧ C)
PT1: Cannot realize this function!
PT2: Cannot realize this function!
PT3: CAN realize this function. HOW ? -- Left as an exercise!

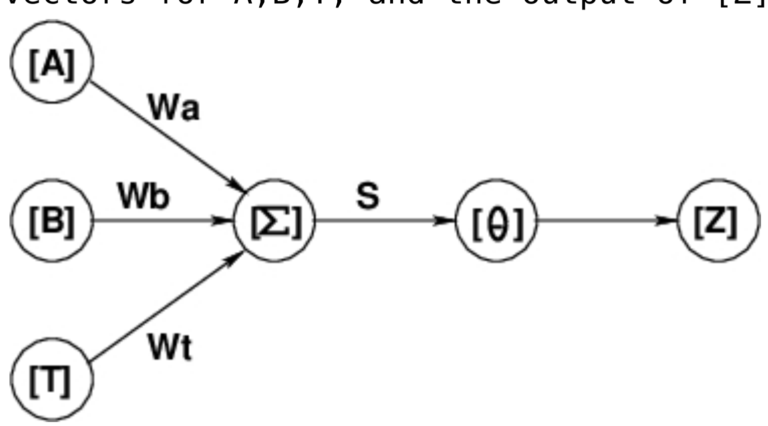
==

==

Question-5: [Artificial Neural Networks]   [Marks: 1 x 15 = 15]

==

Suppose you are given with one-layer artificial neural network (ANN) as shown in the following figure, where [A],[B] are the inputs, [T] is the mandatory threshold input (bias), [Σ] denotes the summation unit, [θ] denotes the logistic (sigmoid) function block, and [Z] is the output. Assume that, T = -1 (always), W = [Wa,Wb,Wt] are the corresponding weight vectors for A,B,T; and the output of [Σ] unit is S, the output of [θ] unit is Z.



You will be given with the following 3 training examples in the form of <A,B,D> tuple, where D being the target output for (A,B): <0,0,1> ; <0,1,1> ; <1,0,1>.

Moreover, the initial weights are given as, Wa = 0, Wb = 0, Wt = 1.

With these training examples, you will propagate forward and need to use back-propagation algorithm to update the weights in an Stochastic/Incremental Gradient Descent (SGD) manner (that is, by taking one training example at a time and propagating values forward and updating the weights while back propagation after each forward pass).

In this example, you will be going 3 times in forward pass and 3 times in backward pass. In forward steps, your goal is to compute S and Z. In backward steps, your goal is to compute the weight updates ΔW = [ΔWa,ΔWb,ΔWt] and find the new weight W = [Wa,Wb,Wt] values. Assume that, the learning rate (step-size), η = 1. So, fill in the 15 vacant entries, indicated using (i)-(xv), of the following table. Write only the numeric values as answer (Calculations of Real values MUST be presented upto TWO decimal places (approximated) after the point).

[Hint: Try computing the weight update factor (partial derivative) δ using (D-Z).]

Pass	A	B	T	Wa	Wb	Wt	S	Z	D	
Forward	0	0	-1	0	0	1	(i)	(ii)	1	[2 marks]
Backward	-	-	-	(iii)	(iv)	(v)	-	-	-	[3 marks]
Forward	0	1	-1	-do-	-do-	-do-	(vi)	(vii)	1	[2 marks]
Backward	-	-	-	(viii)	(ix)	(x)	-	-	-	[3 marks]
Forward	1	0	-1	-do-	-do-	-do-	(xi)	(xii)	1	[2 marks]
Backward	-	-	-	(xiii)	(xiv)	(xv)	-	-	-	[3 marks]

Note:

(I) In all the fields mentioned as "-do-", the previous values (calculated/mentioned in just above cell) are used (obvious!).

(II) The values of all the fields mentioned as "-" are not used.

(III) Calculations of Real values must be presented upto two decimal places after the point (approximated).

~~~~~~

Answers:

(i) -1 (ii) 0.27

(iii) 0 (iv) 0 (v) 0.86

(vi) -0.86 (vii) 0.30

(viii) 0 (ix) 0.15 (x) 0.71

(xi) -0.71 (xii) 0.33

(viii) 0.15 (ix) 0.15 (x) 0.56

~~~~~~

Explanations:

(i)+(ii)

S = A x Wa + B x Wb + T x Wt = 0 x 0 + 0 x 0 + (-1) x 1 = -1

Z = sigmoid(S) = sigmoid(-1) = 1/(1+e^-(-1)) = 0.27

(iii)+(iv)+(v)

δ = (D-Z) x Z x (1-Z) ... Because, it is the last (and only) layer

= (1-0.27) x 0.27 x (1-0.27) = 0.14

ΔWa = η x δ x A = 1 x 0.14 x 0 = 0 and Wa = 0 + ΔWa = 0 + 0 = 0

ΔWb = η x δ x B = 1 x 0.14 x 0 = 0 and Wb = 0 + ΔWb = 0 + 0 = 0

ΔWt = η x δ x T = 1 x 0.14 x (-1) = -0.14 and Wt = 1 + ΔWt = 1 - 0.14 = 0.86

(vi)+(vii)

S = A x Wa + B x Wb + T x Wt = 0 x 0 + 1 x 0 + (-1) x 0.86 = -0.86

Z = sigmoid(-0.86) = 1/(1+e^-(-0.86)) = 0.30

(viii)+(ix)+(x)
δ = (D-Z) x Z x (1-Z) ... Because, it is the last (and only) layer
= (1-0.30) x 0.30 x (1-0.30) = 0.15
ΔWa = η x δ x A = 1 x 0.15 x 0 = 0 and Wa = 0 + ΔWa = 0 + 0 = 0
ΔWb = η x δ x B = 1 x 0.15 x 1 = 0.15 and Wb = 0 + ΔWb = 0 + 0.15 = 0.15
ΔWt = η x δ x T = 1 x 0.15 x (-1) = -0.15 and Wt = 0.86 + ΔWt = 0.86 + (-0.15) = 0.71

(xi)+(xii)
S = A x Wa + B x Wb + T x Wt = 1 x 0 + 0 x 0.15 + (-1) x 0.71 = -0.71
Z = sigmoid(S) = sigmoid(-0.71) = 1/(1+e^-(-0.71)) = 0.33

(xiii)+(xiv)+(xv)
δ = (D-Z) x Z x (1-Z) ... Because, it is the last (and only) layer
= (1-0.33) x 0.33 x (1-0.33) = 0.15
ΔWa = η x δ x A = 1 x 0.15 x 1 = 0.15 and Wa = 0 + ΔWa = 0 + 0.15 = 0.15
ΔWb = η x δ x B = 1 x 0.15 x 0 = 0 and Wb = 0.15 + ΔWb = 0.15 + 0 = 0.15
ΔWt = η x δ x T = 1 x 0.15 x (-1) = -0.15 and Wt = 0.71 + ΔWt = 0.71 + (-0.15) = 0.56
==

==
Question-6: [Support Vector Machines] [Marks: 3 + 3 + 2 + 2 = 10]
==
Hard-Margin Support Vector Machines (SVMs) can learn a decision boundary leading to the largest margin from both classes. You are training SVM on a tiny data-set with 4 points in a 2-D plane (x1,x2) with two classes, A and B, as follows:

Point (x1,x2)	Class
(1,4)	A
(2,3)	A
(4,5)	B
(5,6)	B

This data-set consists of two examples with class label 'A' and two examples with class label 'B'.

Answer the following questions:

(i) Let the equation corresponding to the decision boundary (separating class A from class B) be P.x1 + Q.x2 + R = 0. What will be the vales of P, Q and R? Write only the numeric values as answer. [3 marks]
Note: In your answer, try to make P, Q, R as integers such that the equation of the decision boundary become (or looks) simplest, i.e. all of P, Q, R should NOT be further divisible by a common integer factor greater than 1.

(ii) Let the weight vector be W = [w1,w2] and bias be B. What will be the values of w1, w2 and B, which will be learnt using Hard-margin SVM training algorithm with the mentioned 4 training points? Write only the numeric values as answer. [3 marks]

(iii) What are all the support vectors in this case? Choose an option from the following: [2 marks]

(I) (2,3) and (4,5)
(II) (2,3) and (5,6)
(III) (1,4) and (4,5)
(IV) (1,4) and (5,6)
(V) (1,4) ; (2,3) and (4,5)
(VI) (1,4) ; (2,3) and (5,6)
(VII) (4,5) ; (5,6) and (1,4)
(VIII) (4,5) ; (5,6) and (2,3)
(IX) (1,4) ; (2,3) and (4,5) ; (5,6)
(X) Cannot be found using Hard-Margin SVM

(iv) State whether the following statement is True or False:
  (a) "For two dimensional data points, the separating hyperplane learnt by a li
near Hard-Margin SVM may not always be a straight line." [1 mark]
  (b) "A linear Hard-Margin SVM model can only classify linearly separable data.
" [1 mark]
~~~~~~
Answers:
(i) P = Q = 1 and R = -7
(ii) w1 = w2 = -0.5 and B = 3.5
(iii) (V)
(iv) <False , True>
~~~~~~
Explanations:
(i) SVM tries to maximize the margin between two classes. Therefore, the optimal
 decision boundary is diagonal and it crosses the point (3,4). It is perpendicul
ar to the line between support vectors (4,5) and (2,3), hence it is slope is m =
 -1.
Thus the line equation is (x2 - 4) = (-1).(x1 - 3) => x1 + x2 - 7 = 0.

(ii) From the above equation, we can deduce that the weight vector of the form (
w1,w2) has w1 = w2. It also has to satisfy the following equations:
2w1 + 3w2 + b = 1 , and
4w1 + 5w2 + b = -1 .
Hence, w1 = w2 = -1/2 and b = 7/2

Alternative Logic:
From three support vectors (1,4), (2,3) and (4,5) (i.e. closest points from the
separating line), we get,
 w1 + 4w2 + b = 1 ,
2w1 + 3w2 + b = 1 , and
4w1 + 5w2 + b = -1 .
First two equations produces, w1 = w2.
Now, from the last two equations, we derive as earlier, w1 = w2 = -1/2 and b = 7
/2.

[Note: w1 = w2 = 1/2 and b = -7/2 is also correct as it essentially mean the sa
me (just swap -1 and +1 class value)]

(iii) Perpendicular distance from (1,4), (2,3) and (4,5) to the line x1 + x2 - 7
 = 0 is equal and minimum.
==