Quiz 1

- Due No due date
- Points 10
- Questions 10
- Available Jan 16 at 7:10pm - Jan 18 at 11:59pm
- Time Limit None
- Allowed Attempts 3

Instructions

Intro and Universal Approximators

This quiz covers lectures 1 and 2. Several of the questions invoke concepts from the hidden slides in the slide deck, which were not covered in class. So please go over the slides before answering the questions.

You will have three attempts for the quiz. Questions will be shuffled and you will not be informed of the correct answers until after the deadline. While you may discuss the concepts underlying the questions with others, you must solve all questions on your own - see course policy.

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 1	99 minutes	7.92 out of 10
LATEST	Attempt 2	49 minutes	7 out of 10
	Attempt 1	99 minutes	7.92 out of 10

Correct answers are hidden.

Score for this attempt: 7 out of 10
Submitted Jan 18 at 10:22pm
This attempt took 49 minutes.

Question 1
1 / 1 pts

Which of your quiz scores will be dropped?

Hint: watch Lecture 0

- Lowest 3 quiz scores
- No scores will be dropped
- Lowest 2 quiz scores
- Lowest 1 quiz scores

Question 2

1 / 1 pts

Is the following statement true or false? Hebbian learning allows reduction in weights and learning is bounded.

Slide: lec 1, "Hebbian Learning" slide 66

- True
- False

If neuron x repeatedly triggers neuron y, the synaptic knob (Weight) connecting x to y gets larger. Hence the weight only increases and a mechanism for weight reduction is not given. Also, the upper bound to which the weight increases to is not defined in the learning, making it unbounded.

Question 3

1 / 1 pts

Match the corresponding terms and definitions introduced in Lecture 1.

Hint: Lecture 1: Slides on 31-81

The McCulloch and Pitts model

[is a Logical Calculus of the Id]

Alexander Bain

[is known for his Connectionis]

Lawrence Kubie

[modeled the memory as a circ]

Hebbian Learning

[is a proposed mechanism to u ▾]

Marvin Minsky and Seymour Papert

[Showed that the simple (singl ▾]

One of David Hartley's Observations

[Our brain represents compou ▾]

Frank Rosenblatt

[made the first algorithmically ▾]

Associationism Theory by Aristotle

[These are his four laws: The l ▾]

Incorrect

Question 4

0 / 1 pts

How does the number of weights (note: not neurons) in an XOR network with **threshold logic** perceptrons with **1 hidden layer** grow with the number of inputs to the network?

Hint: Review Lec 2: Slides on "Caveat 2" (Slide 75)

- ○ Between polynomial and exponential
- ○ Linear
- ◉ Exponential or faster
- ○ Polynomial but faster than linear

A depth-2 threshold (TC) circuit can compute parity/XOR using O(n^2) weights. Since the number of weights grows quadratically with the number of inputs, this growth is polynomial rather than linear or exponential.

Incorrect

Question 5

0 / 1 pts

How does the number of weights (note: not neurons) in an XOR network with *threshold logic* perceptrons with *1 hidden layer* grow with the number of inputs to the network?

See lec 2: Slides on “Optimal depth” and “Network size” 113-123

- (selected) Linear
- Between polynomial and exponential
- Exponential or faster
- Polynomial but faster than linear

In the XOR circuit shown in the slides, the number of perceptrons grows linearly with input, and each perceptron has only two inputs and, hence, a fixed number of weights (2). Thus, the number of weights too grows linearly with input

Question 6
1 / 1 pts

Suppose the data used in a classification task is 10-dimensional and positive in all dimensions. You have two neural networks. The first uses threshold activation functions for hidden layers, and the second uses softplus activation functions for hidden layers. In both networks, there are 2000 neurons in the first hidden layer, 8 neurons in the second hidden layer, and a huge number of neurons for all later layers. It turns out that the first network can never achieve perfect classification. What about the second network?

Hint: A layer with 8 neurons effectively projects the data onto a 8-dimensional surface of the space. The input is 10-dimensional.

Lec 2 slides 138-150

- (selected) It might fail for some data sets, since the 8 neurons in the second hidden layer could bottleneck the flow of information. In that case, the sizes of layers 3 and above don't matter.

Answer: B. The 8-neuron layer reduces the dimensionality of the data from 10 to 8. This results in a loss of dimensionality, and potentially, of information.

- Assuming that layers 3 and above are so expressive that they never bottleneck the flow of information, the second network will be able to achieve perfect classification.
- It might fail for some data sets, not only because a bottleneck can occur, but also because the 2000 neurons in the first hidden layer could bottleneck the flow of information if the classification task is sufficiently complex.
- It too is guaranteed to be unable to achieve perfect classification.

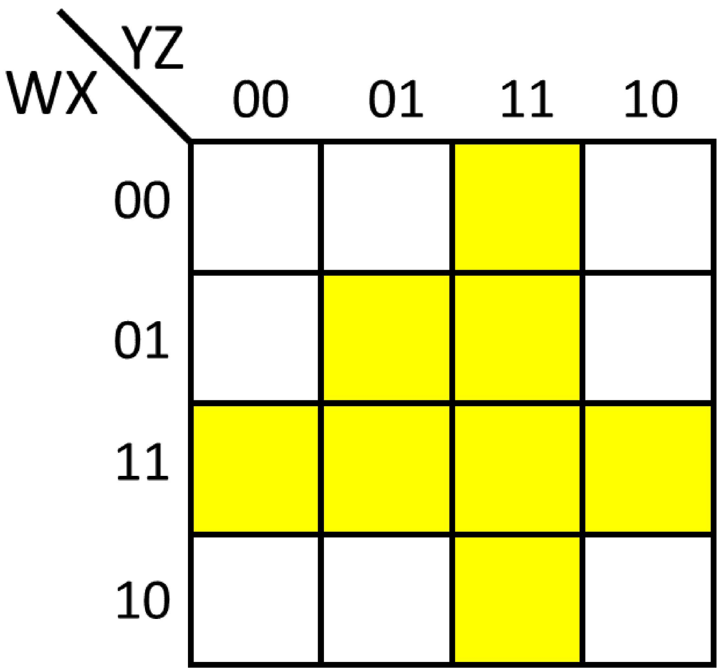
Regardless of the activations, such a loss of dimensionality can result in loss of key information which prevent perfect classification, unless the data themselves actually live in an 8-dimensional surface in the 10-dimensional space. The fact that the threshold activation network fails to achieve perfect accuracy in spite of having a large number of initial feature detectors suggests that the data do not in fact live on an 8-dimensional surface, although we cannot be certain of it.

Incorrect

Question 7

0 / 1 pts

What is the fewest number of neurons needed (including any output layer neurons) for a network to implement the truth table shown by the following Karnaugh map? (numeric answer, int and float are both fine)



Hint: lec 2, “Reducing a Boolean Function”. Slide 47 - 50

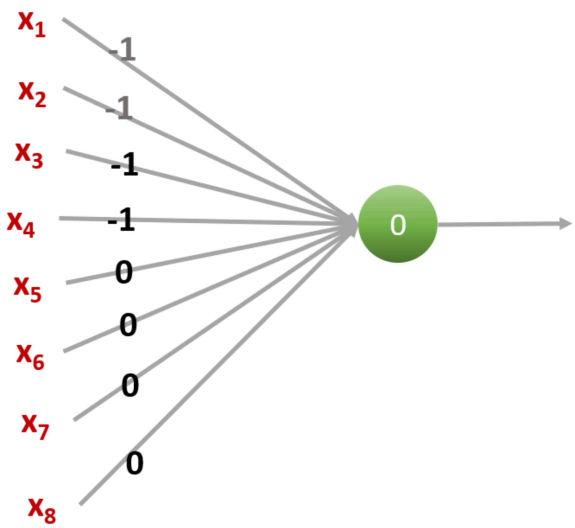
3

The Boolean function is O = WX + YZ + XZ. The reduced DNF has 3 terms, so the Boolean network uses 3 hidden units plus 1 output unit (total 4)

Question 8

1 / 1 pts

Under which conditions will the perceptron graph below fire? Note that ~ is NOT. (select all that apply)



Slide: lec 2, "Perceptron as a Boolean gate", slides 26-30

- [x] fires only if x1, x2, x3, x4 are all 0, regardless of x5 .. x8
- [ ] x1 & x2 & x3 &x4
- [x] ~x1 & ~x2 & ~x3 & ~x4
- [ ] Never fires

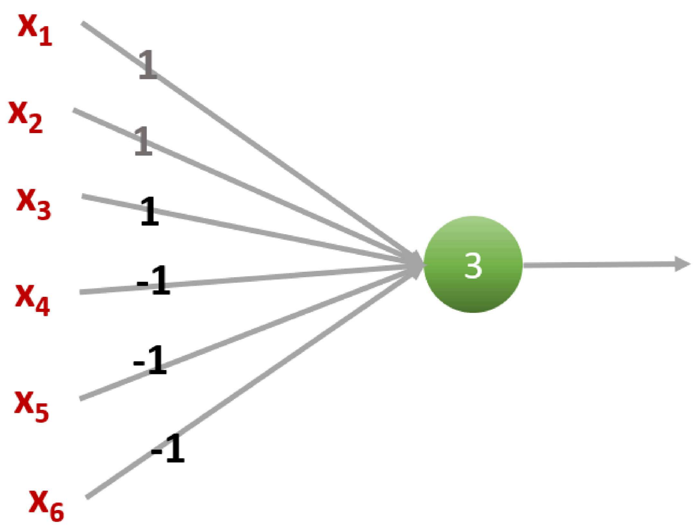
The number above each connection is the connection's weight w_i, which is multiplied to its corresponding X_i input (either 0 or 1). For the perceptron to fire, the sum of all w_i * X_i must be >= to the number in the circle.

As w_5 ~ w_8 are 0, any activations here won't influence firing. But if at least one of [X_1, X_2, X_3, X_4] are 1, the perceptron will not fire, as the total will never be >= the threshold 0.

(~x1 & ~x2…) is only true when x1…x4 are all zero.

Question 9

1 / 1 pts



Under which condition(s) is the perceptron graph above guaranteed to fire? Note that ~ is NOT. (select all that apply)

Slide: lec 2, "Perceptron as a Boolean gate" slides 26-30

- [x] x1 & x2 & x3 & ~x4 & ~x5 &~x6

x1 & x2 & x3 & ~x4 & ~x5 &~x6 = 1(1) + 1(1) +1(1) + 0(-1)+0(-1) +0(-1) =3

x1 &~x2 & x3 & ~x4 & x5 & ~x6= 1(1) + 0(1) +1(1) + 0(-1)+1(-1) +0(-1) =1

~x1 & ~x2 & ~x3 & x4 & x5 & x6 = 0(1) + 0(1) +0(1) + 1(-1)+1(-1) +1(-1) =-3

- [ ] ~x1 & ~x2 & ~x3 & x4 & x5 & x6
- [ ] Never fires
- [ ] x1 &~x2 & x3 & ~x4 & x5 & ~x6

For this perceptron to fire, you need the total to be >= 3. Clearly you'll need all possible positive contributions, with all negative inputs turned off

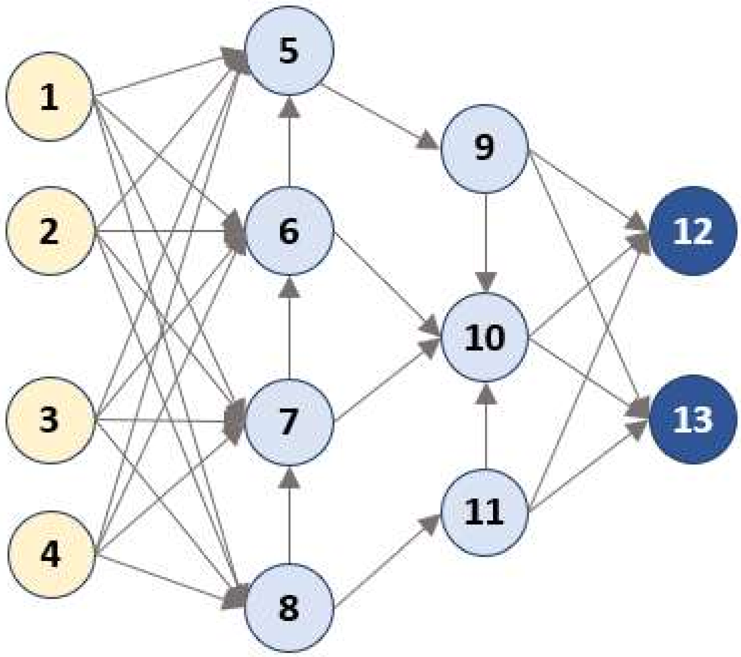
x1 & x2 & x3 & ~x4 & ~x5 &~x6 = 1(1) + 1(1) +1(1) + 0(-1)+0(-1) +0(-1) =3

x1 &~x2 & x3 & ~x4 & x5 & ~x6= 1(1) + 0(1) +1(1) + 0(-1)+1(-1) +0(-1) =1

~x1 & ~x2 & ~x3 & x4 & x5 & x6 = 0(1) + 0(1) +0(1) + 1(-1)+1(-1) +1(-1) =-3

Question 10

1 / 1 pts



If the yellow nodes are inputs (not neurons) and the dark blue nodes are outputs, which neurons are in layer 6?

Hint: lec 2, slides 19-20 on "What is a layer"

- 10, 12
- 10
- 9, 10, 11
- 10, 12, 13

Admittedly this one is tricky. Pay attention to the definitions. Definition of layer #: the neurons reachable (with the same number of edges) along the longest path from source to sink. If a node is reachable multiple times along the longest path, its layer # is the max of those reachable cases (i.e. if some node D was reachable along the longest path 3 times [1,2,3] along the longest path, its layer number is 3).

The longest path would be from any input -> 8 -> 7 -> 6 -> 5 -> 9 -> 10 -> any output. Node 8 would count as layer 1, as you traverse 1 edge along the longest path to get there. While it is possible to visit 5,6,7,8 via one edge, they are visited later along the longest path. Layer 2 would be 7 AND 11, as 11 is not visited later along the longest path, but is still reachable with the same number of edges.

But layer 6 is node 10 only. While 10 is reachable twice along the longest path (once via 3 edges once via 6 edges), remember that we choose the max. Also, 12 and 13 are visited AFTER 10 along the longest path, meaning they are part of layer 7.

So the sixth layer only contains node 10.

Quiz Score: 7 out of 10