21/21 Questions Answered
Saved at 3:38 AM

HW 7

Q1 Multiclass Perceptron

32 Points

In this problem, we will train a Multi-class perceptron on data of the form $(f(X) \in \mathbb{R}^2, Y \in \{A, B, C\})$. In particular, we will use training data to update three weight vectors, $W_y \in \mathbb{R}^2, y = A, B, C$.

We begin with the following set of randomly-initialized weight vectors:

y	$W_{y,1}$	$W_{y,2}$
A	-0.82	-0.02
B	-1.63	-0.88
C	0.39	0.65

Q1.1

6 Points

We will now incorporate the training data point $f(X) = (-1.06, 0.95); Y = C$.
First fill in the resulting weight-feature dot products.

$W_A \cdot f(X)$

.8502

$W_B \cdot f(X)$

.8918

$W_C \cdot f(X)$

.2041

EXPLANATION

$$W_A \cdot f(X) = [-0.82 \quad -0.02] \begin{bmatrix} -1.06 \\ 0.95 \end{bmatrix} = 0.8502$$

$$W_B \cdot f(X) = [-1.63 \quad -0.88] \begin{bmatrix} -1.06 \\ 0.95 \end{bmatrix} = 0.8918$$

$$W_C \cdot f(X) = [0.39 \quad 0.65] \begin{bmatrix} -1.06 \\ 0.95 \end{bmatrix} = 0.2041$$

✔ Correct

Save Answer Last saved on **Aug 05 at 3:32 AM**

Q1.2

6 Points

Now update the weight values as necessary for the training point from part 1.

Note:

Note:
For all of these questions, if a weight vector doesn't get updated, make sure to still write its value in the blank provided.

new $W_{A,1}$

-.82

new $W_{A,2}$

-.02

new $W_{B,1}$

-.57

new $W_{B,2}$

-1.83

new $W_{C,1}$

-.67

new $W_{C,2}$

1.6

EXPLANATION

$Y = argmax_y W_y \cdot f(X) = B$
$Y^* = C$

W_A does not get updated.

$$\begin{aligned}\text{new } W_B &= W_B - f(X)^\top \\ &= [-1.63 \quad -0.88] - [-1.06 \quad 0.95] \\ &= [-0.57 \quad -1.83]\end{aligned}$$

$$\begin{aligned}\text{new } W_C &= W_C + f(X)^\top \\ &= [0.39 \quad 0.65] + [-1.06 \quad 0.95] \\ &= [-0.67 \quad 1.6]\end{aligned}$$

✔ Correct

Save Answer Last saved on **Aug 05 at 3:34 AM**

Q1.3

6 Points

We will now incorporate the training data point $f(X) = (0.09, 1.48); Y = A.$ Fill in the resulting weight-feature dot product, and update the weight values as necessary.

$W_A \cdot f(X)$

-.1034

$W_B \cdot f(X)$

-2.7597

$W_C \cdot f(X)$

2.3077

EXPLANATION

Note that we are using the updated weights.

$$W_A \cdot f(X) = [-0.82 \quad -0.02] \begin{bmatrix} 0.09 \\ 1.48 \end{bmatrix} = -0.1034$$

$$W_B \cdot f(X) = [-0.57 \quad -1.83] \begin{bmatrix} 0.09 \\ 1.48 \end{bmatrix} = -2.7597$$

$$W_C \cdot f(X) = [-0.67 \quad 1.6] \begin{bmatrix} 0.09 \\ 1.48 \end{bmatrix} = 2.3077$$

✔ Correct

Save Answer Last saved on **Aug 05 at 3:34 AM**

Q1.4

6 Points

new $W_{A,1}$

-.73

new $W_{A,2}$

1.46

new $W_{B,1}$

-.57

new $W_{B,2}$

-1.83

new $W_{C,1}$

$W_{C,1}$

-.76

new $W_{C,2}$

.12

EXPLANATION

$Y = argmax_y W_y \cdot f(X) = C$
$Y^* = A$

new $W_A = W_A + f(X)^\top$
$= [-0.82 \quad -0.02] + [0.09 \quad 1.48]$
$= [-0.73 \quad 1.46]$

W_B does not get updated.

new $W_C = W_C - f(X)^\top$
$= [-0.67 \quad 1.6] - [0.09 \quad 1.48]$
$= [-0.76 \quad 0.12]$

✔ Correct

Save Answer Last saved on **Aug 05 at 3:34 AM**

Q1.5

8 Points

We took over from here and ran the perceptron algorithm till convergence. In case you're curious, this data set consisted of 50 data points, and the perceptron algorithm converged after 722 steps. Of these steps, 103 changed the weight vector.

At convergence, we have the following weight vectors:

y	$W_{y,1}$	$W_{y,2}$
A	3.12	0.96
B	3.11	-0.97
C	-8.29	-0.24

Use the converged perceptron to classify the new data point $f(X) = (-1.35, 0.42)$. Fill in the weight-feature dot product for each value of y.

$W_A \cdot f(X)$

-3.8088

$W_B \cdot f(X)$

-4.606

$W_C \cdot f(X)$

11.09

What is the predicted label?

- ○ A
- ○ B
- ◉ C

EXPLANATION

For a multi-class perceptron, the label, Y, is chosen for a data point, X, as

$$Y = argmax_y W_y \cdot f(X).$$

When training a perceptron, if the label chosen by the perceptron matches the label provided with the training data, the weights do not change.

However, if the label differs, say the perceptron classified the data point as Y, but it should have been some other label, Y^*, then the weights must be updated. This update is performed as:

$$W_y = W_y - f(X)$$

and

$$W_{y^*} = W_{y^*} + f(X)$$

✔ Correct

Save Answer Last saved on **Aug 05 at 3:35 AM**

Q2 A Variant on the Perceptron Algorithm

10 Points

You were recently promoted to the Vice President of Recruiting Science at Pacapalooza Technologies. Pacapalooza is expanding rapidly, and you decide to use machine learning to hire the best and brightest. To do so, you have the following available to you for each candidate i in the pool of candidates I: (i) their GPA, (ii) whether they took CS 164 with Hilfinger and received an A, (iii) whether they took CS 188 and received an A, (iv) whether they have a job offer from Pactronic LLC, (v) whether they have a job offer

from Pacmania Corp., and (vi) the number of misspelled words on their resume. You decide to represent each candidate $i \in I$ by a corresponding 6-dimensional feature vector $f(x^{(i)})$. You believe that if you just knew the right weight vector $w \in \mathbb{R}^6$ you could reliably predict the quality of a candidate i by computing $w^T f(x^{(i)})$. To determine w, you sample pairs of candidates from the pool. For a pair of candidates (k, l) you can have them face off in a "Pacapalooza-fight." The result is $\text{score}(k > l)$, which tells you that a candidate k is at least $\text{score}(k > l)$ better than candidate l. Note that the score will be negative when l is a better candidate than k. Assume you collected scores for a set of pairs of candidates P, that $\text{score}(k > l) = -\text{score}(k < l)$, and that $\text{score}(k > l) \neq 0$ for any pair $(k, l) \in P$.

Q2.1

6 Points

You decide to employ a perceptron-like algorithm to determine w, where your dataset is P.

Suppose that we encounter a pair $(k, l) \in P$ for which $\text{score}(k > l) > 0$. How do we update w?

- (●) If $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$, do nothing. Otherwise update $w \leftarrow w + f(x^{(k)}) - f(x^{(l)})$.
- ( ) Update $w \leftarrow w + f(x^{(k)}) - f(x^{(l)})$.
- ( ) If $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$, do nothing. Otherwise update $w \leftarrow w - f(x^{(k)}) + f(x^{(l)})$.
- ( ) If $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$, do nothing. Otherwise update $w \leftarrow w + w^T(f(x^{(k)}) - f(x^{(l)}))$.
- ( ) Update $w \leftarrow w - f(x^{(k)}) + f(x^{(l)})$.

Suppose that we encounter a pair $(k, l) \in P$ for which $\text{score}(k > l) < 0$. How do we update w?

- If $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$, do nothing. Otherwise update $w \leftarrow w - f(x^{(l)}) - f(x^{(k)})$.
- Update $w \leftarrow w - f(x^{(k)}) + f(x^{(l)})$.
- If $w^T f(x^{(l)}) \geq w^T f(x^{(k)}) - \text{score}(k > l)$, do nothing. Otherwise update $w \leftarrow w + f(x^{(l)}) - f(x^{(k)})$.
- Update $w \leftarrow w + f(x^{(k)}) - f(x^{(l)})$.
- If $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$, do nothing. Otherwise update $w \leftarrow w - w^T(f(x^{(k)}) - f(x^{(l)}))$.

EXPLANATION

The idea behind the solution is to use a margin-based perceptron. Given a pair of candidates k, l, our goal is to have $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$ if $\text{score}(k > l) > 0$. Otherwise, we want the opposite inequality, that is, $w^T f(x^{(l)}) \geq w^T f(x^{(k)}) - \text{score}(k > l) = w^T f(x^{(k)}) + \text{score}(l > k)$.

So, our perceptron-like algorithm is as follows: we repeat

1. Choose a random pair $(k, l) \in P$ (you can also sequentially run through P)
2. If $\text{score}(k > l) > 0$: if $w^T f(x^{(k)}) \geq w^T f(x^{(l)}) + \text{score}(k > l)$, do nothing; else, update $w \leftarrow w + f(x^{(k)}) - f(x^{(l)})$.
3. If $\text{score}(k > l) \leq 0$: if $w^T f(x^{(l)}) \geq w^T f(x^{(k)}) - \text{score}(k > l)$, do nothing; else, update $w \leftarrow w + f(x^{(l)}) - f(x^{(k)})$.

✔ **Correct**

Save Answer Last saved on **Aug 05 at 3:36 AM**

Q2.2

4 Points

Your perceptron-like algorithm is unable to reach zero errors on your training data. Which of the following techniques would help improve performance on the training data?

- [ ] Running the perceptron algorithm for a longer period of time; since the perceptron algorithm is guaranteed to keep equal or reduce the number of errors at each time step, we are guaranteed to eventually reach zero errors.
- [x] Add higher-order features to our list of six features, e.g., pairwise products, and run our perceptron algorithm with this newly constructed dataset. New features increase the dimensionality of the space, and improve the chances that the data is separable.
- [ ] Removing some of the features from the training data, and training the perceptron on this subset of data. Too many features increases the chance of overfitting on the training data, which would decrease performance on the training data.
- [ ] Collect a larger set of data, so the perceptron algorithm does a better job of fitting to the data distribution; with a small training set, the perceptron cannot fully learn w, causing it to produce errors on the training data.

EXPLANATION

Running the algorithm for a longer period of time does not necessarily help improve performance, since the perceptron algorithm doesn't necessarily decrease the number of errors at each time step.

each time step.

Adding higher-order features could help lower the error rate, since the data could be linearly separable in this higher dimensional space.

Removing features doesn't help, as this reduces dimensionality. Reducing dimensionality will keep equal or increase the training error.

Training on a larger set of data will not help reduce the number of errors on the current training set, but will rather likely increase the number of errors. The introduction of more samples will cause the perceptron to have to focus on properly classifying the new samples.

✔ **Correct**

Save Answer Last saved on **Aug 05 at 3:36 AM**

Q3 Separability

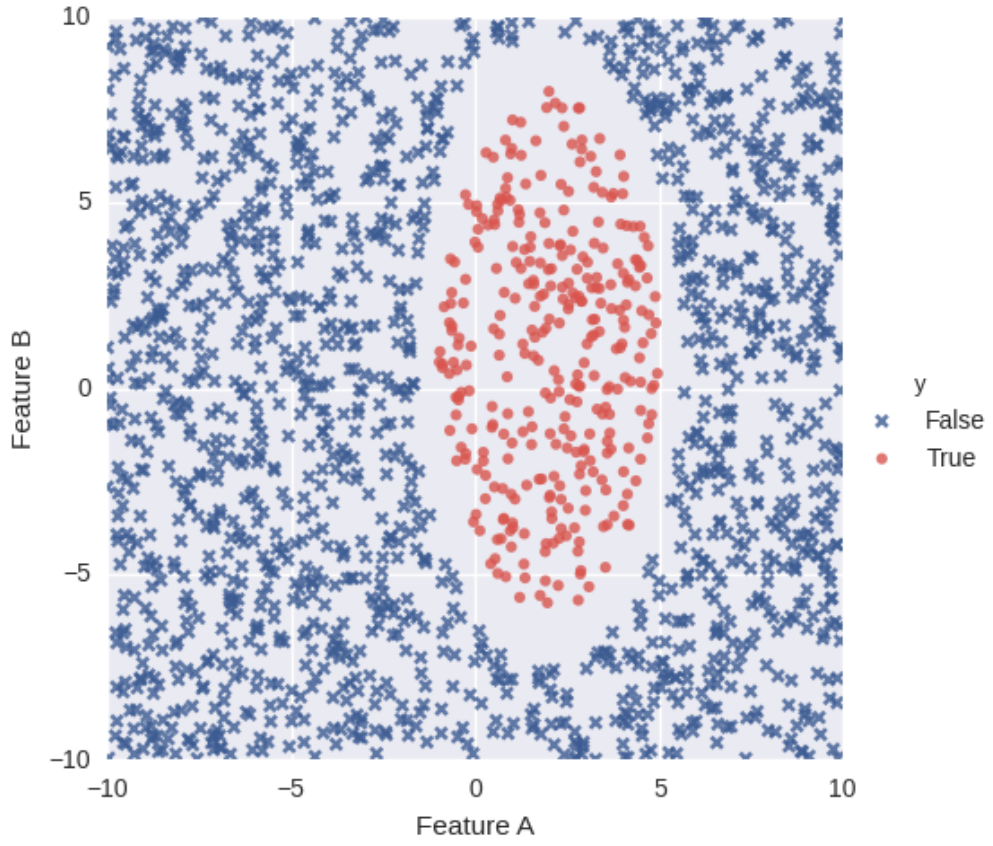
10 Points

Q3.1

5 Points

It is well known that Pactronic LLC is the premier manufacturer of Pacmen. At Pactronic LLC, quality control is currently done manually -- a group of scientists decide whether a Pacman is ready to be released into the wild based on (Feature (A)) a Pacman's intelligence score and (Feature (B)) a Pacman's empathy score. Here are many examples of Pacmen that have been released and withheld in the past. Each dot corresponds to a Pacman, and responds to the following question as true or false: this Pacman is

ready to be released.



As the Vice President of Science, you would like to automate the decision making process, and decide to use the perceptron algorithm. Which of the following subsets of features would allow you to perfectly classify whether or not a Pacman can be released in the wild?

- [ ] (A, B)
- [x] (A^2, AB, B^2, A, B)
- [ ] (A, B, X), where $(X = (A \geq C_1) \wedge (B \geq C_2)$ for some fixed (C_1, C_2) that you are allowed to pick.
- [ ] (A)
- [ ] (B)

(B)

EXPLANATION

(A, B) doesn't work, since the data is not linearly separable. (A^2, AB, B^2, A, B) works, since taking a linear combination of these features gives us a set of classification boundaries equivalent to the full set of conics. Ellipses fall under the set of conics, which means that this set of features works.

The features (A, B, X) where $X = (A \geq C_1) \wedge (B \geq C_2)$ for some fixed (C_1, C_2) that you are allowed to pick doesn't work, since this feature only allows us to model ellipses.

Neither (A) nor (B) alone doesn't allow us to perfectly classify the data points, since the data is not linearly separable in the features.

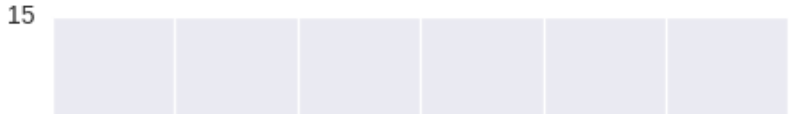
✔ **Correct**

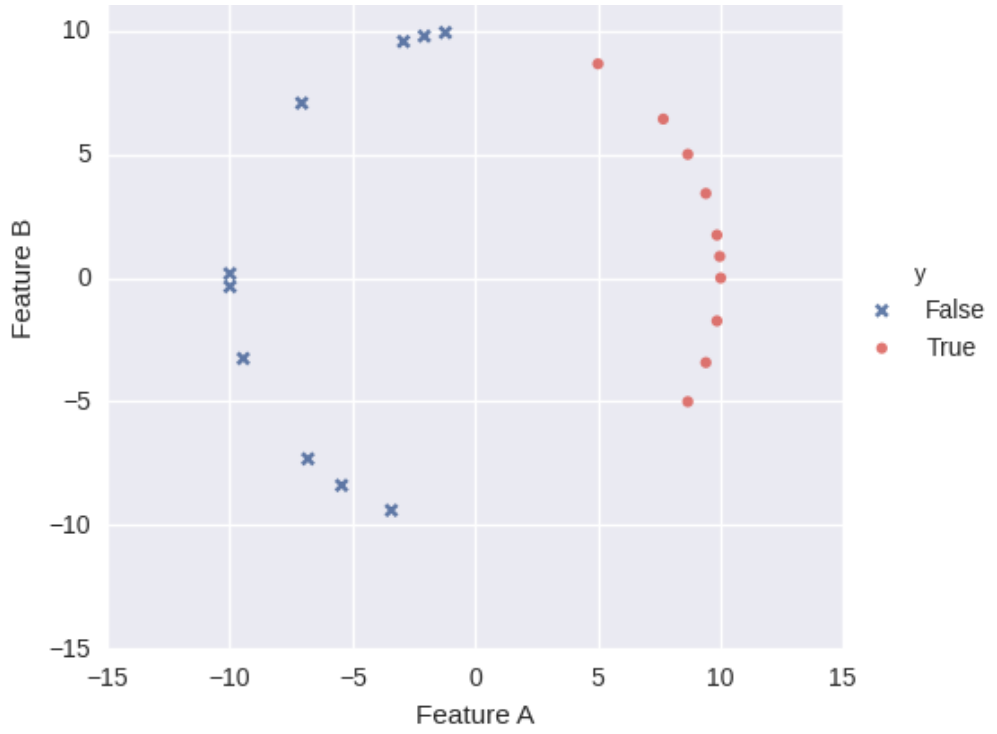
Save Answer Last saved on **Aug 05 at 3:37 AM**

Q3.2

5 Points

The CEO of Pactronic soon decides that the company will be focusing on creating fewer, but much better Pacmen. This calls for an entire re-design of the Pacman. Accordingly, the scientists come up with the latest and greatest generation of Pacmen, and once again seek your advice in quality control. Here are the newest Pacman, and their respective features:





Which of the following subsets of features would allow you to perfectly classify whether or not a Pacman can be released in the wild?

- ✔ (A, B)
- ✔ (A^2, AB, B^2, A, B)
- ✔ (A, B, X), where $(X = (A \geq C_1) \wedge (B \geq C_2)$ for some fixed (C_1, C_2) that you are allowed to pick.
- ✔ (A)
- ☐ (B)

EXPLANATION

Notice that the data is linearly separable in (A), so any list of features that includes (A) works. The data is not separable in (B), so the last option doesn't work.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:37 AM**

Q4 Local Optima and Gradient Descent

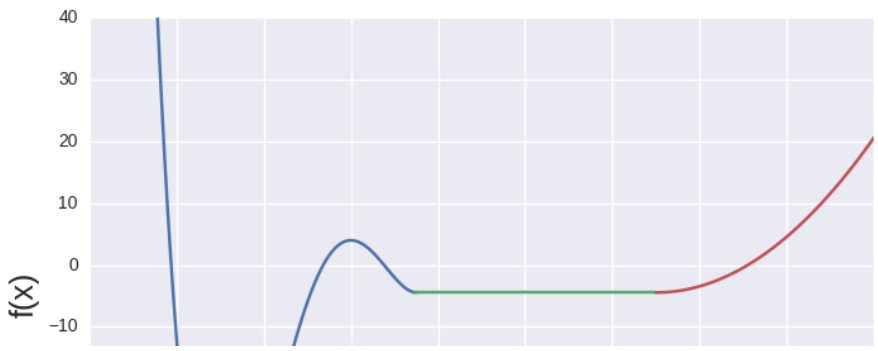
23 Points

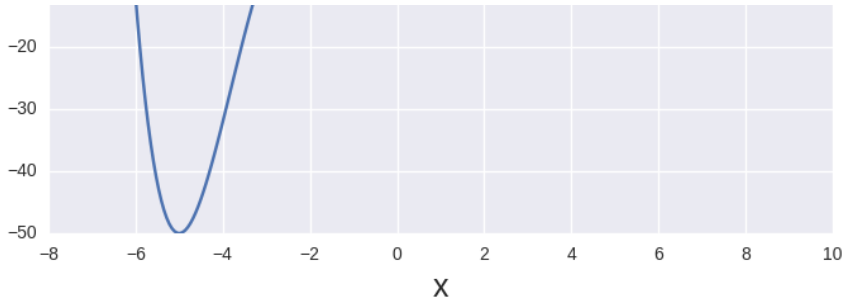
After a busy year of chasing ghosts, Pacman and Paclady are planning to visit the Kakslauttanen Arctic Resort for their winter vacation. Paclady who is particularly fond of skiing, excitedly begins planning ahead. Pacman, who is apprehensive of skiing (when asked why, he rambles on about the Aspen Red Ghost Chase of 2012, but we won't get into that), reluctantly agreed to go skiing, but under one condition: Paclady must tell Pacman how steep the slopes are at several points of interest.

Paclady asks the resort for terrain details, and receives the following graph. The resort says at any given location x, $f(x)$ models the terrain height. Specifically:

- when $x \leq -\frac{1}{2}$, $f(x) = \frac{1}{2}x^4 + 5x^3 + \frac{27}{2}x^2 + 10x$,
- when $-\frac{1}{2} \leq x \leq 5$, $f(x) = -\frac{71}{16}$,
- and when $x \geq 5$, $f(x) = x^2 - 10x + \frac{329}{16}$.

See below for a plot:





The local optima for f lie at $x = -5$ and $x = -2$, with a plateau in the region $-1/2 \leq x \leq 5$.

Q4.1

5 Points

Paclady decides to compute derivatives to measure how steep slopes are.
Evaluate $f'(-6)$.

-44

EXPLANATION

At $x = -6$, we have that $f(x) = \frac{1}{2}x^4 + 5x^3 + \frac{27}{2}x^2 + 10x$ and that $f'(x) = 2x^3 + 15x^2 + 27x + 10$. Substituting, we have that $f'(-6) = -44$.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q4.2

5 Points

Evaluate $f'(0)$.

0

EXPLANATION

At $x = 0$, we have that $f(x) = 71/16$, implying that $f'(x) = 0$. So, $f'(0) = 0$.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q4.3

5 Points

Evaluate $f'(8)$.

6

EXPLANATION

At $x = 8$, we have that $f(x) = x^2 - 10x + 329/16$, implying that $f'(x) = 2x - 10$. Substituting, we have that $f'(8) = 6$.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q4.4

4 Points

Pacman and Paclady get to the resort, and have a fantastic time skiing, but get lost. Unfortunately, a blizzard kicks in right then,

reducing visibility. As Pacman panics and brings up the Aspen Red Ghost Chase of 2012, Paclady remembers that their glass igloo cabin is located at the global minimum elevation point of the resort ($x = -5$). The blizzard complicates things, since they can't ski due to the reduced visibility for safety.

After thinking for a minute, Pacman says, "Aha! We can get home in that case by following gradient descent, as long as we employ a small step size -- once we hit a gradient of 0, we know we're home!" Paclady pauses and says, "Your algorithm almost works, but it depends on where in the resort we currently are."

Check all regions where Pacman and Paclady can be, and still find their igloo, assuming that they employ gradient descent with a small step size and stop walking when they encounter a gradient of 0.

- [x] $x < -5$
- [x] $-5 < x < -2$
- [ ] $-2 < x < -1/2$
- [ ] $-1/2 < x < 3$
- [ ] $3 < x < 5$
- [ ] $x > 5$

Note: Make sure you select all of the correct options--there may be more than one!

EXPLANATION

Notice that for $x <; 2$, running gradient descent with small steps leads to the global minimum, attained at $x = -5$. For $x > 2$, running gradient descent with small steps leads to a local optimum, namely the plateau.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:37 AM**

Q4.5

4 Points

While slowly trudging to their igloo via gradient descent, Pacman and Paclady get into an argument. Pacman complains that trudging down a hill is tiresome, and that they instead should have gotten an igloo closer to $x = 3$. Paclady says that Pacman's previous gradient descent algorithm wouldn't lead them to the igloo in this case, unless they were already at the igloo. Why is this the case?

- ○ Gradient descent would cause Pacman and Paclady to reach $x = -2$ rather than $x = 3$, since it is at a local maximum.
- ◉ Gradient descent terminates when it reaches a gradient of 0, and neighboring regions around $x = 3$ all have a gradient of 0, so Pacman and Paclady would stop searching outside of $x = 3$, within the plateau.
- ○ When gradient descent is stuck in a plateau, it searches for regions with negative rather than zero gradient.
- ○ When gradient descent is stuck in a plateau, it searches for regions with positive rather than zero gradient.
- ○ Gradient descent seeks to maximize a function, which would lead Pacman and Paclady either to $-\infty$ or to ∞.

EXPLANATION

Regions around the plateau have gradient 0, and gradient descent terminates when reaching a gradient of 0. This would cause Pacman and Paclady to get stuck in the plateau.

✔ Correct

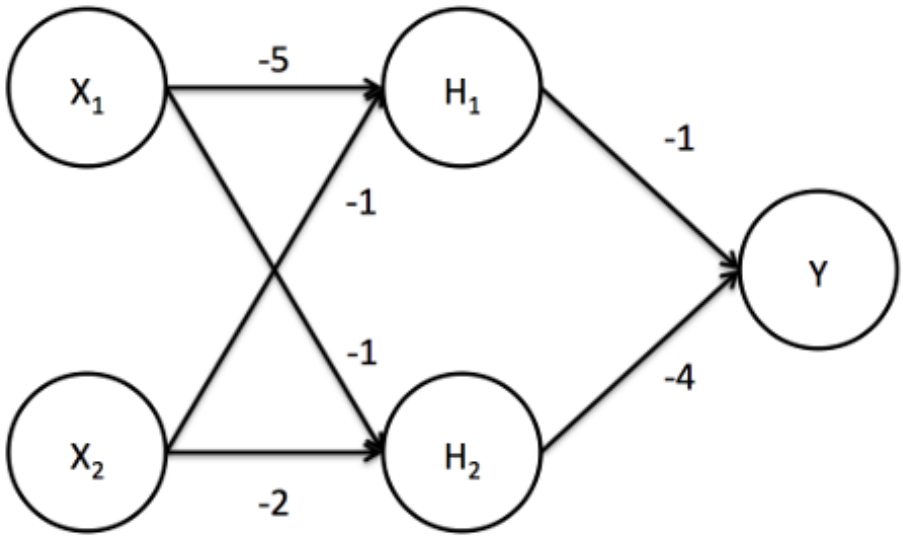
Save Answer Last saved on **Aug 05 at 3:37 AM**

Q5 Neural Networks and Logic Gates

16 Points

As you probably know, Pacumus Maximus Corporation (PMC) is the most well known company that manufactures the gears that let Pacman open and close its mouth. Recently, the Vice President of Science in PMC decided to replace all of its low-level NAND, AND, and NOR gates with mini neural networks. Unfortunately, there was a Pacman uprising, where the Pacmen took over the factory, eating all of the neural network documentation. The scientists were able to salvage the following neural networks weights, but don't remember which gates these neural networks corresponded to. They've hired you to help them recover this information.

Here is the first network that you're given:



Above, the nodes H_1, H_2, and Y have sigmoid activations and

Above, the nodes H_1, H_2, and Y have sigmoid activations and each have biases of 0.5. The inputs are placed in X_1 and X_2. To convert the output Y into a boolean value, we round Y. This will be the case for all problems in this section.

Concretely, we have the following, where w_{AB} denotes the weight between nodes A and B:

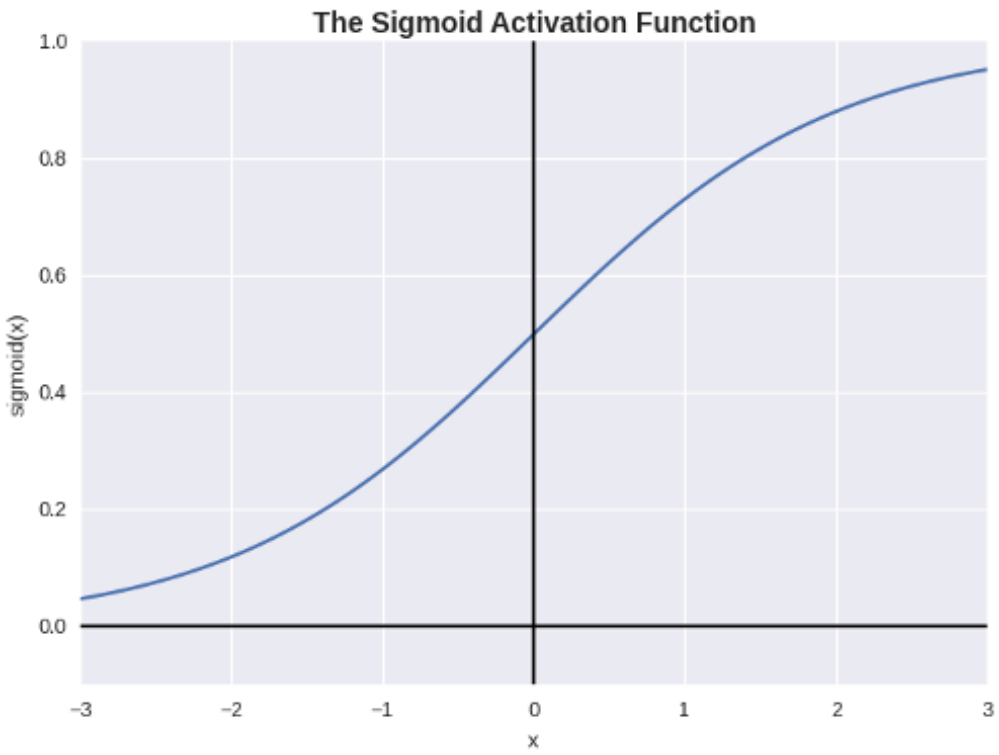
$$H_1(x) = \sigma(w_{H_1X_1} \cdot X_1 + w_{H_1X_2} \cdot X_2 + 0.5)$$

$$H_2(x) = \sigma(w_{H_2X_1} \cdot X_1 + w_{H_2X_2} \cdot X_2 + 0.5)$$

$$Y(x) = \text{round}\{\sigma(w_{YH_1} \cdot H_1 + w_{YH_2} \cdot H_2 + 0.5)\}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

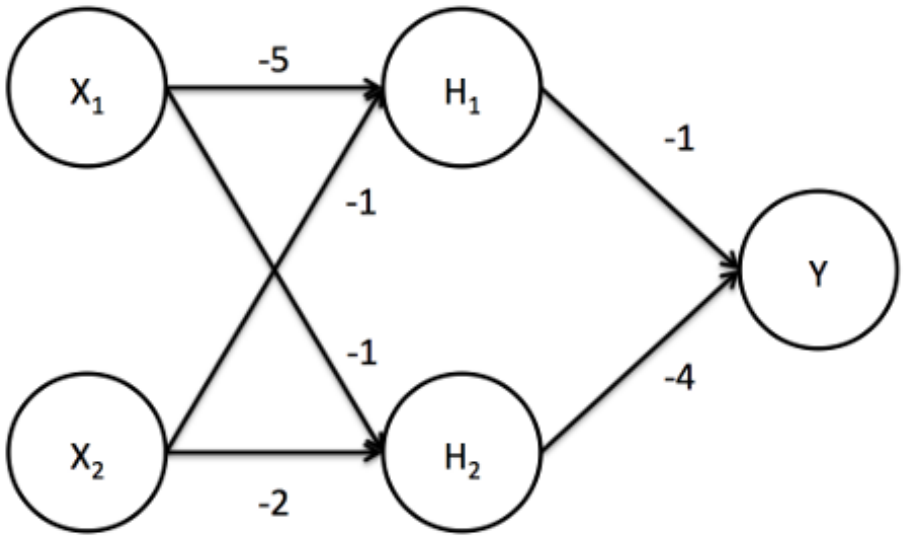
Recall that the sigmoid function, $\sigma(x)$, looks like this:



Q5.1

4 Points

Here's the first network again, for convenience:



What does this first network correspond to?

- ○ NAND
- ◉ AND
- ○ NOR

EXPLANATION

Forward propagate the following pairs of (X_1, X_2) into the network, according to the equations specified above: $(0,0), (0,1), (1,0), (1,1)$. If the values of Y are $1, 1, 1, 0$, respectively, then the network corresponds to a NAND gate. If the values of Y are $0, 0, 0, 1$, respectively, then the network corresponds to an AND gate. If the values of Y are $1, 0, 0, 0$, then the network corresponds to a NOR gate.

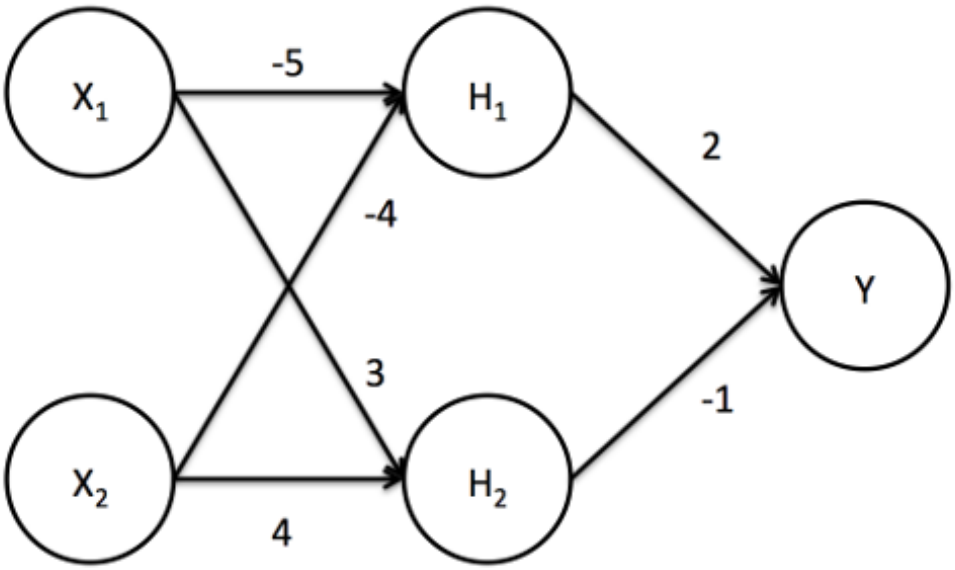
✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q5.2

4 Points

Here is the second network that you're given:



What does this network correspond to?

- NAND
- AND
- NOR

EXPLANATION

Forward propagate the following pairs of (X_1, X_2) into the network, according to the equations specified above: $(0,0), (0,1), (1,0), (1,1)$. If the values of Y are $1, 1, 1, 0$, respectively, then the network corresponds to a NAND gate. If the values of Y are $0, 0, 0, 1$, respectively, then the network corresponds to an AND gate. If the values of Y are $1, 0, 0, 0$,

then the network corresponds to a NOR gate.

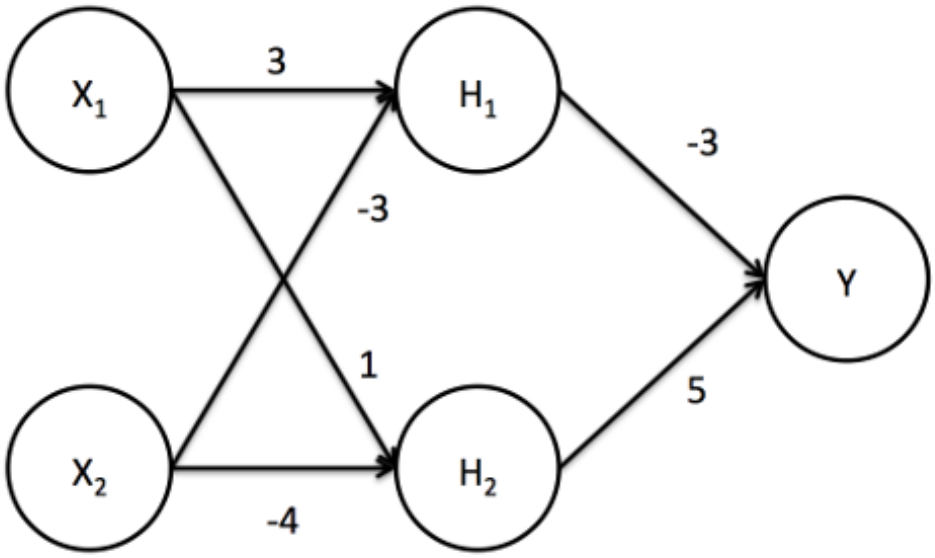
✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q5.3

4 Points

Here is the third network that you're given:



What does this network correspond to?

- (●) NAND
- ( ) AND
- ( ) NOR

EXPLANATION

Forward propagate the following pairs of (X_1, X_2) into the network, according to the equations specified above:

$(0,0), (0,1), (1,0), (1,1)$. If the values of Y are $1, 1, 1, 0$, respectively, then the network corresponds to a NAND gate. If the values of Y are $0, 0, 0, 1$, respectively, then the network corresponds to an AND gate. If the values of Y are $1, 0, 0, 0$, then the network corresponds to a NOR gate.

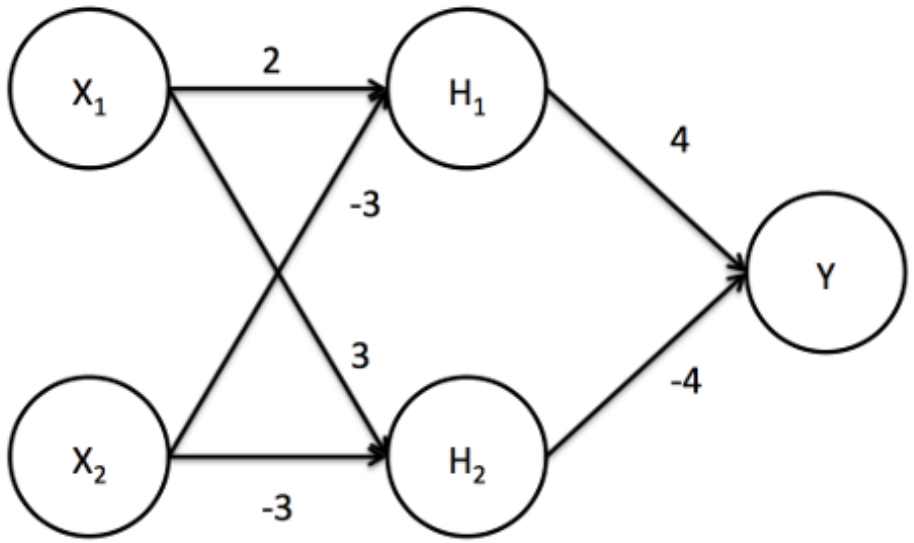
✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q5.4

4 Points

Here is the final network that you're given:



What does this network correspond to?

- (●) NAND
- ( ) AND
- ( ) NOR

EXPLANATION

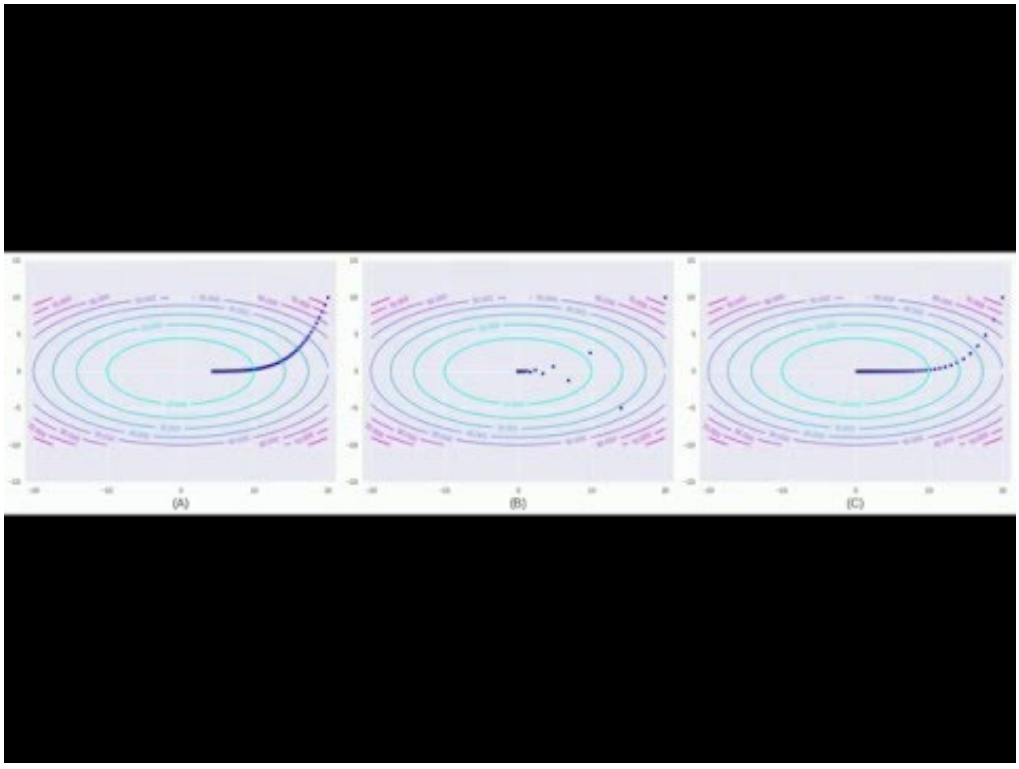
Forward propagate the following pairs of (X_1, X_2) into the network, according to the equations specified above: $(0,0), (0,1), (1,0), (1,1)$. If the values of Y are $1, 1, 1, 0$, respectively, then the network corresponds to a NAND gate. If the values of Y are $0, 0, 0, 1$, respectively, then the network corresponds to an AND gate. If the values of Y are $1, 0, 0, 0$, then the network corresponds to a NOR gate.

✔ **Correct**

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q6 Learning Rates

9 Points



http://www.youtube.com/watch?v=FaDgovU4_0o

Watch the above YouTube video. There are three sub-panels. We will refer to the left one by (A), the middle one by (B), and the right one by (C). The same objective is being optimized by gradient descent, but has different learning rates in each subpanel.

Q6.1
3 Points

Which animation corresponds to the lowest learning rate?

- (●) Animation (A)
- ( ) Animation (B)
- ( ) Animation (C)

EXPLANATION

The algorithm on the left has the lowest learning rate, as it converges the slowest to the optimum.

✔ **Correct**

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q6.2
3 Points

Which animation corresponds to the medium learning rate?

- ( ) Animation (A)
- ( ) Animation (B)
- (●) Animation (C)

EXPLANATION

The algorithm on the right has the medium learning rate, as (1) it is faster than the left algorithm and (2) its gradients do not bounce back and forth, as in the middle algorithm.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Q6.3

3 Points

Which animation corresponds to the largest learning rate?

- ○ Animation (A)
- ◉ Animation (B)
- ○ Animation (C)

EXPLANATION

Large learning rates in gradient descent are characterized by oscillating gradients that move back and forth; this happens in the middle image. The other algorithms do not exhibit this behavior, so the middle algorithm has the largest learning rate. Also, the steps that this algorithm takes has visibly larger sizes than those of the other two algorithms.

✔ Correct

Save Answer Last saved on **Aug 05 at 3:38 AM**

Save All Answers Submit & View Submission ›