

## Q1 Instruction

0 Points

Instructions:

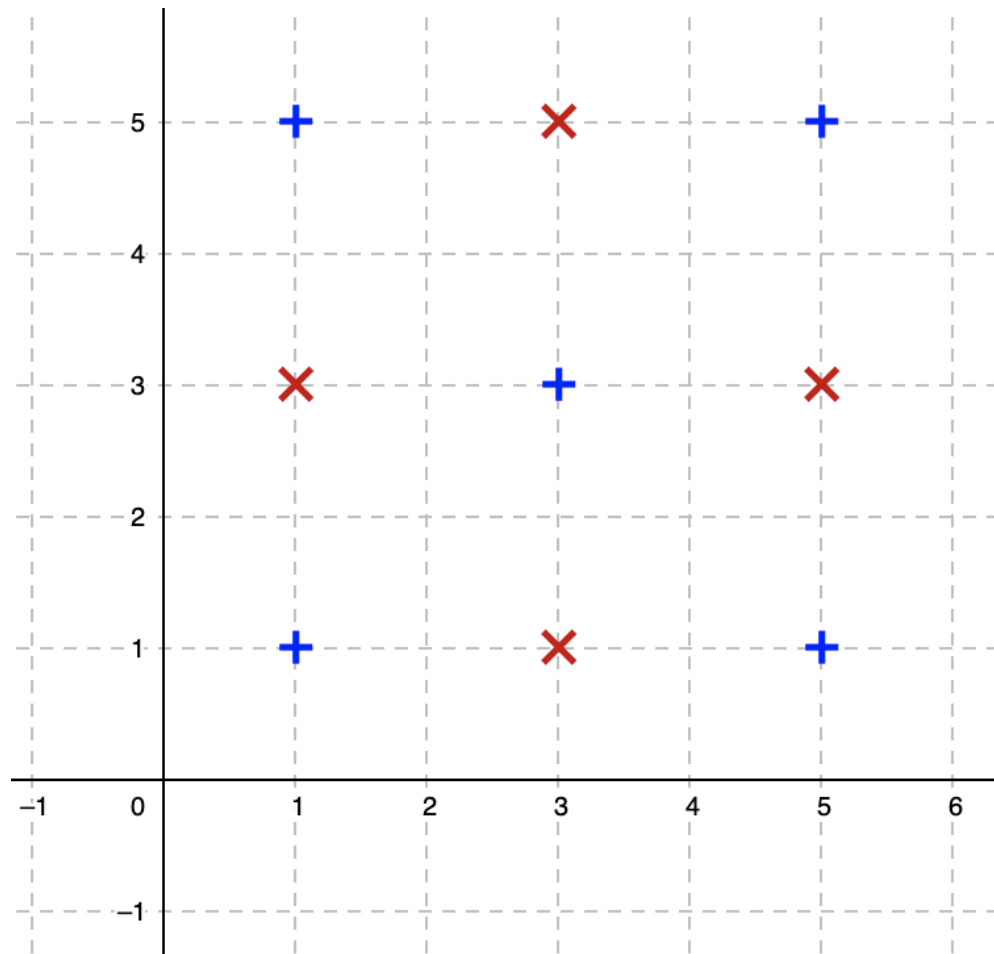
- This is an open book exam -- you can use course materials posted at BruinLearn as your reference. Please do not access **any other material** during the exam. Discussion is strictly prohibited.
- Calculator is allowed.
- This exam booklet has a total of 100 points.
- The exam period is from 10:00am--11:50am 11/3.
- No late submissions will be accepted for any reason. Therefore, make sure to submit well before the deadline! If you face a technical issue, contact us in Piazza (or email) immediately.
- You can ask **private** questions in Piazza, but only clarification questions will be answered. Please don't discuss the exam with anyone except the instructor/TA.
- If you think the question is ambiguous, please write down your explanations or comments at the end of the exam (i.e., Question 7). The regrading will be only based on the submitted answers and explanations. However we may not be able to offer partial credit.
- Note that we cannot provide customized rubrics. Also, as we have already broken down some big questions into sub-questions, we will not provide further partial credits for some sub-questions. In some cases, if you make mistake on earlier sub-questions, it may lead to a wrong answer for the following sub-questions. The rubrics are already designed with that in mind.
- You can type your answer using markdown. For example, type `$$\gamma$$` will generate  $\gamma$ .
- For numerical questions, we ask you to round your answer to the nearest 2 decimal places and write in the format of X.XX or -X.XX). For example, if your answer is 1 (or 0.995), please write down 1.00.
- **Remember to save your answer frequently**

Best of luck!

## Q2 KNN

## 12 Points

Consider the following training dataset, what is the leave-one-out validation accuracy (i.e., accuracy computed using 9-fold cross-validation) for the following classifier?

**Q2.1**

4 Points

1-NN with Euclidean distance.

(Round your answer to two decimal places)

**Q2.2**

4 Points

3-NN with Euclidean distance

(Round your answer to two decimal places)

0.00

### Q2.3

4 Points

5-NN with Euclidean distance

(Round your answer to two decimal places)

0.44

### Q3 Decision tree

22 Points

When training deep neural networks, out-of-memory errors often happen, depending on factors such as model size, batch size, and the quality of implementations. We will use the following dataset to learn a decision tree that predicts if an out-of-memory error will happen, based on 3 attributes (batch size, network depth, and the implementation version).

Batch size	Depth	Implementation	Out-of-memory?
small	Deep	A	No
small	Shallow	B	No
small	Medium	B	Yes
large	Shallow	A	No
large	Medium	A	Yes
large	Shallow	B	Yes
large	Deep	B	Yes

In case that more than one attribute has equal information gain, the priority of choosing the attributes is ordered as Batch size > Depth > Implementation.

You may use the following formula.

$$\text{Entropy: } H(p) = -(p \log_2 p + (1 - p) \log_2 (1 - p))$$

$$H(0) = 0; H(1/2) = 1; H(1/3) = 0.933; H(1/4) = 0.8;$$

$$H(1/5) = 0.7; H(1/7) = 0.571; H(3/7) = 0.971$$

You may also use  $\log_2 3 = 1.6$ ,  $\log_2 5 = 2.3$ , and  $\log_2 7 = 2.8$

Please round your answer to 2 decimal places.

### Q3.1 Decision Tree

6 Points

What is the entropy of  $\mathcal{H}(\text{out-of-memory})$ ?

$\mathcal{H}(\text{out-of-memory}) =$

0.99

### Q3.2

6 Points

What is the information gain if we partition the data on the attribute **Implementation**?

Information Gain (Implementation) =

0.13

### Q3.3

6 Points

Suppose we learn a decision tree by the ID3 algorithm. What is the attribute used for the first split?

☐ Batch Size

☒ Depth

☐ Implementation

**Q3.4**

4 Points

Based on the learned decision tree learned from ID3, what is the prediction for an input with: Batch size=small, Depth=Medium, Implementation=B?

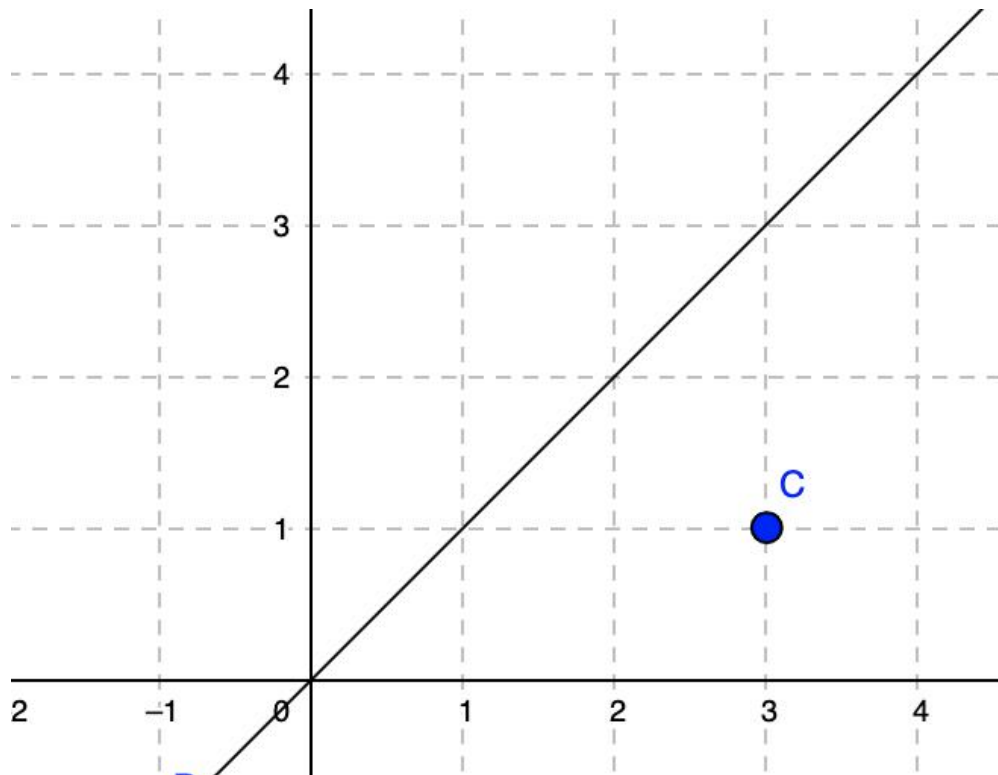
☐ Yes☒ No**Q4 Margin**

19 Points

**Q4.1**

5 Points

Consider the following data point and the hyperplane.



What is the distance between the point C to the line in L2 (euclidean) distance? Round your answer to 2 decimal places.

1.41

**Q4.2**

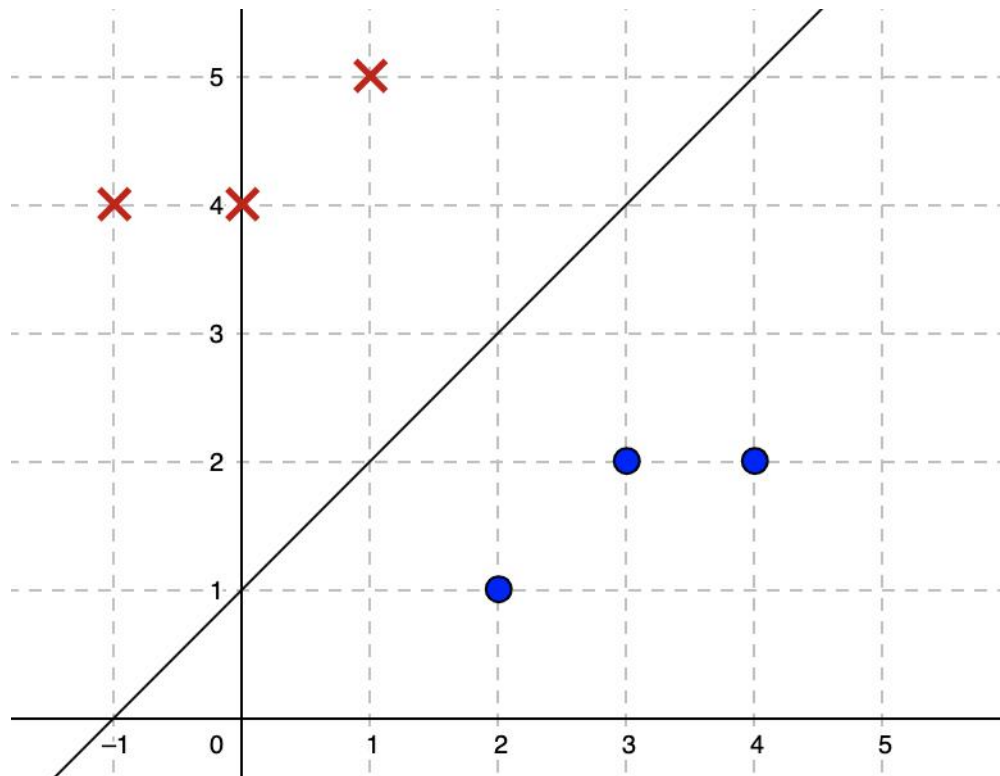
5 Points

Follow the previous question. What is the distance between the point C to the line in L1 (manhattan) distance? Round your answer to 2 decimal places.

2.00

**Q4.3**

5 Points



What is the margin of the hyperplane in L2 (euclidean) distance? Round your answer to 2 decimal places.

1.41

**Q4.4**

4 Points

Follow the previous question. Is there another hyperplane with a larger margin in L2 (euclidean)?

☒ Yes☐ No**Q5 Maximum Likelihood Estimation**

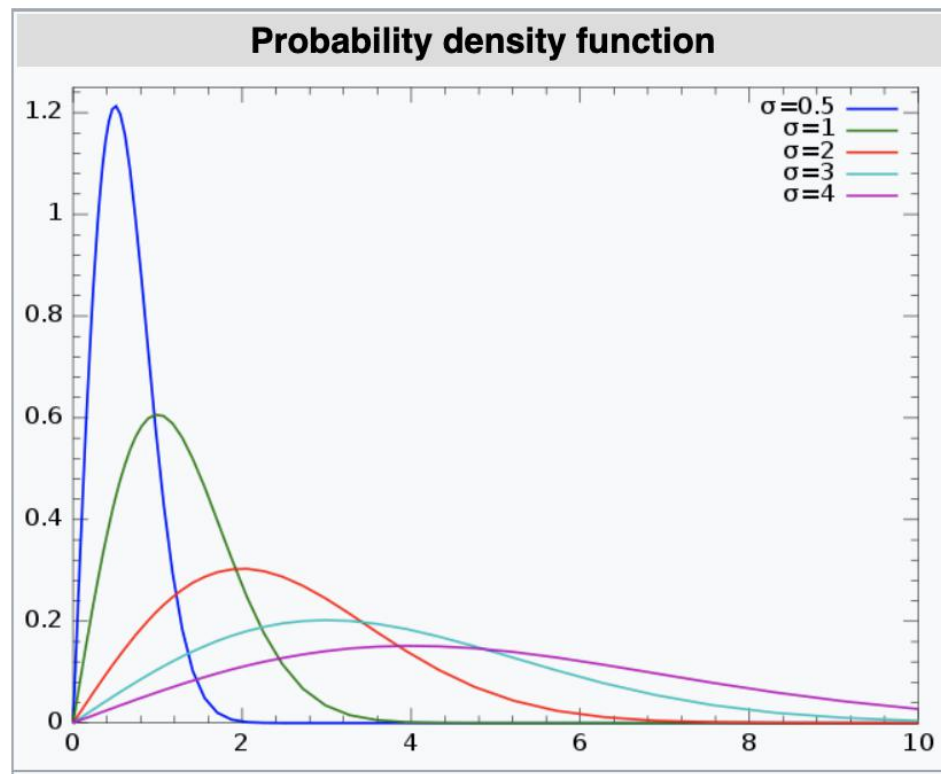
22 Points

In the following questions, we will explore how to estimate the lifetime of a capacitor. The age of a capacitor depends on various factors and we have encoded them into numerical values:  $x$ . Given a bunch of capacitors drawn i.i.d. from the underlying distribution, their age and the factors, we create a training data set  $\{(x_i, y_i)\}_{i=1}^N$  where  $x_i \in \mathbb{R}^d$  and  $y_i \in \mathbb{R}$ . We ask some of our friends from the Statistics department and they suggest that we model the capacitor age using a Rayleigh distribution with parameter  $\sigma = w^T x$  as follows:

$$P(Y = y|\sigma) = \frac{y}{\sigma^2} \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right) \quad \text{where} \quad \exp(z) = e^z$$

That is the probability the lifetime  $y$  of a capacitor  $x$  follows the Rayleigh distribution with  $\sigma = w^T x$ ,  $w \in \mathbb{R}^d$  is the weight vector we can learn.

A visualization of the probability density function of Rayleigh distribution is shown in the following



### Q5.1

4 Points

We first consider giving a learned  $w$ . How we predict the most likely lifetime of a capacitor.

Rayleigh distribution with parameter  $\sigma$  have the following property:

$$\arg \max_y \frac{y}{\sigma^2} \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right) = \sigma$$

That is the peak of the Rayleigh distribution  $P(Y = y|\sigma)$  is when  $y = \sigma$ .

Based on this property, consider the feature vector extracted for a capacitor is  $x = [1, 3, 1]$  and the learned weight vector  $w = [1, 2, 1]$ . What is the most likely lifetime  $y$  of the capacitor?

most likely lifetime =

8



**Q5.2**

4 Points

Now, let's discuss how to learn  $w$  based on the training data set  $\{(x_i, y_i)\}_{i=1}^N$ . We start with considering one data point  $(x_i, y_i)$ .

Based on our assumption, the distribution of the lifetime of a capacitor follow a Rayleigh distribution with parameter  $\sigma = w^T x$ . What is the probability  $P(Y = y_i | x_i, w)$  of that the lifetime of a capacitor  $x_i$  is  $y_i$  based on the assumption.

- ☐  $P(y_i | x_i, w) = \frac{y_i}{x_i^2} \cdot \exp\left(-\frac{y_i^2}{2x_i^2}\right)$
- ☐  $P(y_i | x_i, w) = \frac{y_i}{w^T x_i} \cdot \exp\left(-\frac{y_i^2}{2w^T x_i}\right)$
- ☒  $P(y_i | x_i, w) = \frac{y_i}{(w^T x_i)^2} \cdot \exp\left(-\frac{y_i^2}{2(w^T x_i)^2}\right)$
- ☐  $P(y_i | x_i, w) = \log(y_i) - \log(w^T x_i) - \frac{y_i^2}{2w^T x_i}$

**Q5.3**

4 Points

Given the dataset  $\{(x_i, y_i)\}_{i=1}^N$ , what is the log-likelihood of the model  $w$ ?

- ☐  $\mathcal{LL}(w) = \text{constant} + \sum_{i=1}^N \left(-\log(w^T x_i) - \frac{y_i^2}{2w^T x_i}\right)$
- ☒  $\mathcal{LL}(w) = \text{constant} + \sum_{i=1}^N \left(-2\log(w^T x_i) - \frac{y_i^2}{2(w^T x_i)^2}\right)$
- ☐  $\mathcal{LL}(w) = \text{constant} + \prod_{i=1}^N \frac{y_i}{(w^T x_i)^2} \cdot \exp\left(-\frac{y_i^2}{2(w^T x_i)^2}\right)$
- ☐  $\mathcal{LL}(w) = \text{constant} + \prod_{i=1}^N \left(-\log(w^T x_i) - \frac{y_i^2}{2w^T x_i}\right)$

**Q5.4**

4 Points

How we can obtain  $w$ ?

☒ We can maximize the log-likelihood in previous question

☐ We can minimize the log-likelihood in previous question

### Q5.5

6 Points

How we can solve the optimization problem? (selected all correct options)

☒ The optimization problem can be solved by SGD

☒ The optimization problem can be solved by GD

### Q6 Perceptron

12 Points

Consider the following binary classification dataset with 3 features:

$x_1$	$x_2$	$x_3$	$y$
1	1	2	1
1	2	0	1
4	0	0	-1
2	0	0	-1
0	0	1	1

Consider training a Perceptron model  $y = \text{sgn}(w^T x + b)$ , with  $w$  and  $b$  are initialized with 0.

Note that you can augment  $b$  into  $w$  using the trick we discussed in class.

**Q6.1**

4 Points

After running the Perceptron model over these five data points **once**, what is  $w$  and  $b$

 $w_1 =$  $w_2 =$  $w_3 =$  $b =$ **Q6.2**

4 Points

Given a test data point  $[-1, -3, 1]$ , what is the model prediction.

 $y =$ **Q6.3**

4 Points

If we allow the Perceptron model to run over these five data points **until it converges**, what is the final model

$w_1 =$

-2

$w_2 =$

3

$w_3 =$

2

$b =$

1

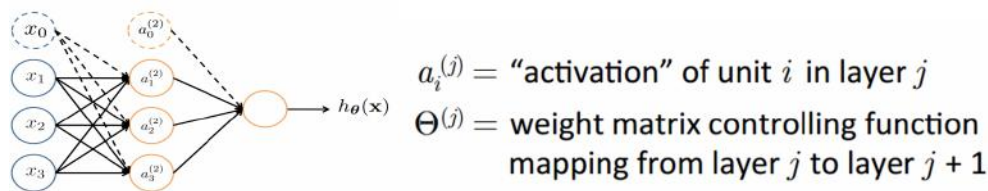
## Q7 Multiple Choices and short answer

13 Points

### Q7.1 Neural Network

6 Points

Consider the following neural network we discussed in the class. For a binary classification problem, the model make prediction on  $x$  based on the sign of  $h_{\Theta}(x)$ . That is the model predicts positive if  $h_{\Theta}(x) > 0$ ; otherwise it predicts negative.



$$\begin{aligned}
 a_1^{(2)} &= g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3) \\
 a_2^{(2)} &= g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3) \\
 a_3^{(2)} &= g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3) \\
 h_{\Theta}(x) &= a_1^{(3)} = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)})
 \end{aligned}$$

If the activation function  $g(z) = \beta z$ , where  $\beta$  is a non-zero constant.

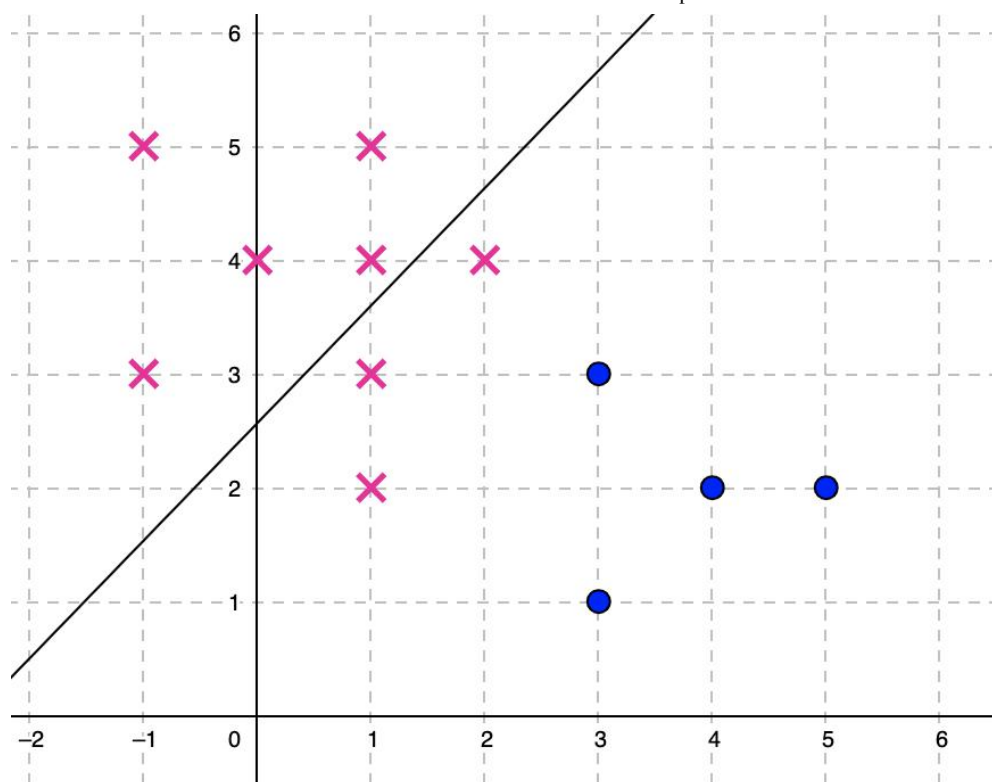
Which of the following is True?

- ☒ If data are linearly separable, we can find a set of parameter  $\Theta$  such that  $\text{sgn}(h_{\Theta}(x))$  can separate all positive and negative data.
- ☐ If data are linearly separable, we **cannot** find a set of parameter  $\Theta$  such that  $\text{sgn}(h_{\Theta}(x))$  can separate all positive and negative data.
- ☐ If data are non-linearly separable, we can find a set of parameter  $\Theta$  such that  $\text{sgn}(h_{\Theta}(x))$  can separate all positive and negative data.
- ☒ If data are non-linearly separable, we **cannot** find a set of parameter  $\Theta$  such that  $\text{sgn}(h_{\Theta}(x))$  can separate all positive and negative data.

## Q7.2

3 Points

Given a set of positive points (blue circle) and negative points (red cross) as training data. We train a logistic regression model using SGD with 100 iterations. The results are shown in the following figure



Which of the following statement is true:

- ☐ The model is overfitting.
- ☒ The model is underfitting.
- ☐ The model has 0 training error

### Q7.3

4 Points

Provide one possible reason why the model does not do well in one sentence.


The data points are clearly not linearly separable, which causes an issue for logistic regression, which has a linear decision boundary created where  $P(Y = 1)$  is approximately 50%.

### Q8 MISC

0 Points

If you have any questions or concerns about this exam, you can type them in the following input box or upload a file. Please note that we are not able to provide customized rubrics and many questions do not have partial credits. However, if you think the question is ambiguous and you would like to provide some additional explanations to clarify your understanding of the exam questions, you can provide them here. We can only regrade the exam based on the write-up you turn in.

:)

 No files uploaded

Midterm Exam

GRADED

STUDENT

CHARLES ZHANG

TOTAL POINTS

96 / 100 pts

QUESTION 1

Instruction

0 / 0 pts

QUESTION 2

KNN

12 / 12 pts

2.1 (no title)

4 / 4 pts

2.2 (no title)

4 / 4 pts

2.3 (no title)

4 / 4 pts

QUESTION 3

Decision tree

22 / 22 pts

3.1 Decision Tree

6 / 6 pts

3.2	(no title)	6 / 6 pts
3.3	(no title)	6 / 6 pts
3.4	(no title)	4 / 4 pts

**QUESTION 4**

Margin **19 / 19 pts**

4.1	(no title)	5 / 5 pts
4.2	(no title)	5 / 5 pts
4.3	(no title)	5 / 5 pts
4.4	(no title)	4 / 4 pts

**QUESTION 5**

Maximum Likelihood Estimation **22 / 22 pts**

5.1	(no title)	4 / 4 pts
5.2	(no title)	4 / 4 pts
5.3	(no title)	4 / 4 pts
5.4	(no title)	4 / 4 pts
5.5	(no title)	6 / 6 pts

**QUESTION 6**

Perceptron **12 / 12 pts**

6.1	(no title)	4 / 4 pts
6.2	(no title)	4 / 4 pts
6.3	(no title)	4 / 4 pts

**QUESTION 7**

Multiple Choices and short answer **9 / 13 pts**

7.1	Neural Network	6 / 6 pts
7.2	(no title)	3 / 3 pts
7.3	(no title)	0 / 4 pts

**QUESTION 8**

MISC **0 / 0 pts**