CSCI-567 Fall 2019 Midterm Exam Ans: [Rubric]

Problem	1	2	3	4	5	Total
Points	30	10	15	25	20	100

Please read the following instructions carefully:

- The exam has a total of **15 pages** (including this cover). Each problem has several questions. Once you are permitted to open your exam (and not before), you should check and make sure that you are not missing any pages.
- Duration of the exam is **2 hours and 20 mins**. Questions are not ordered by their difficulty. Budget your time on each question carefully. **Ask a proctor** if you have any question regarding the exam.
- Select **one and only one answer** for all multiple choice questions.
- Answers should be **concise** and written down **legibly**. All questions can be done within 3-12 lines.
- You must answer on the page of each question. You can use the last blank page as scratch paper.
- This is a **closed-book/notes** exam. Consulting any resources is NOT permitted.
- Any kind of cheating will lead to **score 0** for the entire exam and be reported to SJACS.
- You **may not** leave your seat **for any reason** unless you submit your exam at that point.

1 Multiple Choice, True or False (30 points)

1. Given a dataset for binary classification, kNN with large k tends to have a smoother decision boundary (assuming $N >> k$). **(2 points)**

   (a) True

   (b) False

   Ans: True.

2. Because kNN is a very flexible non-parametric classifier, it can achieve near-perfect classification even for problems in which the true underlying data distributions overlap. **(2 points)**

   (a) True

   (b) False

   Ans: False.

3. Given a dataset which consists of a training set and a development set for tuning hyperparameter k. When we see that choosing a specific k results in very low training error but very high testing error, it is a good sign of underfitting. **(2 points)**

   (a) True

   (b) False

   Ans: False. The description above indicates overfitting.

4. Which of the following penalty functions cannot be a good idea to regularize model complexity? **(3 points)**

   (a) $R(\mathbf{w}) = \exp\{\sum_i |w_i|\}$

   (b) $R(\mathbf{w}) = \exp\{-\sum_i |w_i|\}$

   (c) $R(\mathbf{w}) = -\sum_i \log(|w_i|^{-1})$

   (d) $R(\mathbf{w}) = \sum_i \exp\{|w_i|\}$

   Ans: b

5. Suppose we are training a neural network with mini-batch SGD of batch size 50, and 50000 training samples. How many updates would there be while training during 5 epochs? **(3 points)**

   (a) 50000

   (b) 1000

   (c) 5000

   (d) 250000

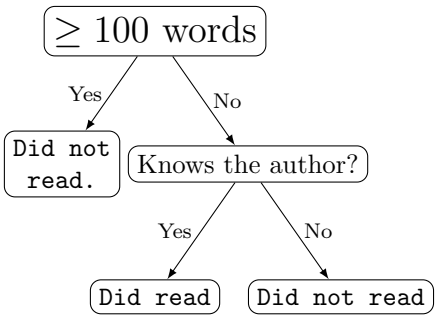
   Ans: C

For questions 6 to 8 consider the following data and two decision trees below. We would like to build a decision tree classifier to answer the question "did Joe read the email ?"

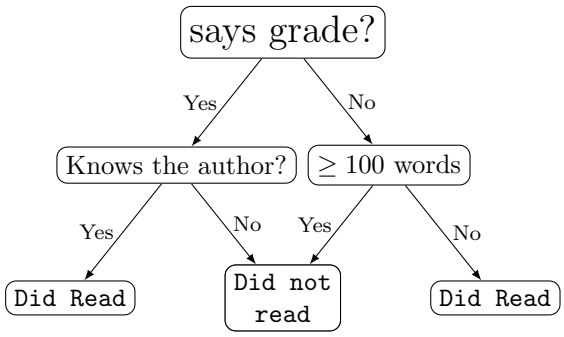
Table 1: Did Joe read the email?

Knows the author?	$\geq$ 100 words	Says research?	Says grade?	Says lottery	Did Joe read it?
no	no	yes	yes	no	no
yes	yes	no	yes	no	no
no	yes	yes	yes	yes	no
yes	yes	yes	yes	no	no
no	yes	no	no	no	no
yes	no	yes	yes	yes	yes
no	no	yes	no	no	yes
yes	no	no	no	no	yes
yes	no	yes	yes	no	yes
yes	yes	yes	yes	yes	no

Here are two decision trees that attempt to classify the emails:



Tree 1



Tree 2

6. Because decision trees learn to classify discrete valued outputs, it is impossible for them to overfit. **(3 points)**

   (a) True

   (b) False

   Ans: b. We have examples from theory assignment 1.

7. How many points in the *training data* are correctly classified by tree 1? **(3 points)**

   (a) 6

   (b) 7

   (c) 8

   (d) 9

   (e) 10

   Ans: d. There is one no/no that Joe did read.

8. How many points in the *training data* are correctly classified by tree 2? **(3 points)**

   (a) 6
   (b) 7
   (c) 8
   (d) 9
   (e) 10

   Ans: b. The three wrong are all yes it says grade, yes Joe knows the author, no I didn't read it.

9. Suppose a convolution layer takes a $5 \times 7 \times 3$ image as input and outputs a $3 \times 4 \times 6$ tensor. Which of the following is a possible configuration of this layer? **(3 points)**

   (a) Three $2 \times 4 \times 3$ filters, stride 1, no zero-padding.
   (b) Three $3 \times 3 \times 3$ filters, stride 1, 1 zero-padding.
   (c) Six $3 \times 4 \times 3$ filters, stride 1, 1 zero-padding.
   (d) Six $3 \times 3 \times 3$ filters, stride 2, 1 zero-padding.

   Ans: D

10. For $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^{2\times 1}$, which of the following bases $\phi(x)$ corresponds to the kernel defined as **(3 points)**

$$k(x, x') = e^{x_1 + x_1'} + e^{2(x_2 + x_2')}$$

   (a) $\phi(x) = [e^{x_1}, e^{x_2}]^T$
   (b) $\phi(x) = [e^{x_1}, \sqrt{2}e^{x_2}]^T$
   (c) $\phi(x) = [e^{x_1}, e^{\sqrt{2}x_2}]^T$
   (d) $\phi(x) = [e^{x_1}, e^{2x_2}]^T$ Ans: ←

11. Consider the dataset consisting of points (x, y), given the basis function $\phi(x, y) = [x^2, 2xy, y^2]^T$, which of the following matrices is the kernel matrix of the three data points $(x_1, y_1) = (1, 0), (x_2, y_2) = (0, 1), (x_3, y_3) = (1, 1)$? **(3 points)**

   (a) $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 6 \end{bmatrix}$
   (b) $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 6 \end{bmatrix}$
   (c) $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 6 \end{bmatrix}$
   (d) $\begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 1 \\ 2 & 1 & 6 \end{bmatrix}$

   Ans: c

2 Linear Regression

Consider a dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where each datapoint is associated with an importance weight $r_i > 0$, $i \in \{1, \cdots, n\}$, then the Weighted Residual Sum of Squares (WRSS) is defined as:

$$\text{WRSS}(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{n} r_i(y_i - \mathbf{w}^T\mathbf{x}_i)^2 \tag{1}$$

12. Define $\mathbf{X}$ as the matrix whose i-th row is $\mathbf{x}_i^T$, and $\mathbf{R}$ as a diagonal matrix where $\mathbf{R}_{ii} = r_i$ and 0 for all other entries. Write down the matrix form of the WRSS objective (Eq. 1). **(3 points)**
Ans: $\frac{1}{2}(\mathbf{y} - \mathbf{X}\mathbf{w})^T\mathbf{R}(\mathbf{y} - \mathbf{X}\mathbf{w})$ or $\frac{1}{2}\|\mathbf{R}^{\frac{1}{2}}(\mathbf{y} - \mathbf{X}\mathbf{w})\|_2^2$ or $\frac{1}{2}\|\sqrt{\mathbf{R}}\mathbf{y} - \sqrt{\mathbf{R}}\mathbf{X}\mathbf{w}\|_2^2$
Missing $\frac{1}{2}$, deduct 1 point.

13. Solve for the optimal $\mathbf{w}^*$ for WRSS in the matrix form. **(7 points)**
Ans:

$$\begin{aligned}&\nabla\frac{1}{2}(\mathbf{y} - \mathbf{X}\mathbf{w})^T\mathbf{R}(\mathbf{y} - \mathbf{X}\mathbf{w})\\ =&\mathbf{X}^T\mathbf{R}(\mathbf{X}\mathbf{w} - \mathbf{y}) \qquad \textbf{(4 points)}\end{aligned}$$

Set the gradient to 0, we get $\mathbf{w}^* = (\mathbf{X}^T\mathbf{R}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{R}\mathbf{y}$ **(3 points)**.
Rubics:
1. correctly calculate the gradient (4 points).
2. set the gradient to 0 and get the final form of w^* (3 points).
Partial Credits:
1. set the gradient to 0 (1 point)
2. show steps of calculating gradients, but the final answer or the initial objective is wrong (1 point)

3 Naive Bayes (15 points)

Naive Bayes classifiers are a family of simple "probabilistic" classifiers based on applying the Bayes' theorem with conditional independence assumptions between the features. In this problem, we will use a naive Bayes classifier with features $\{f_i\}$, where $i = 1, \cdots, d$, and label y , which is defined as follow:

$$\text{argmax}_y \, P(y|f_1, \cdots, f_d) = \text{argmax}_y \, P(y) \prod_{i=1}^{d} P(f_i|y)$$

Additionally, a linear classifier (for example, perceptron) can be defined as follow:

$$\text{argmax}_y \sum_{i=0}^{d} w_{y,i} \cdot f_i \quad \text{where } f_0 \text{ is a bias feature that is always 1 for all data}$$

14. For a naive Bayes classifier with binary-valued features, i.e. $f_i \in \{0, 1\}$, for $i = 0, \cdots, d$, *prove that the naive Bayes classifier is also a linear classifier* by defining weights $w_{y,i}$, for $i = 0, \cdots, d$, such that both classifiers above are equivalent. The weights should be expressed in terms of the naive Bayes probabilities: $P(y), P(f_i = 1|y)$, and $P(f_i = 0|y)$, $i = 1, \cdots, d$. You can assume that all these probabilities are non-zero. **(12 points)**

    Hint: Using the log operation to convert products to summations, i.e. $\log \prod_{i=1}^{d} P(f_i|y) = \sum_{i=1}^{d} \log P(f_i|y)$

    Ans:

    We can use the following formulas to transfer the original naive bayes classifier to a linear classifier:

$$\begin{aligned}
\text{argmax}_y \, P(y) \prod_{i=1}^{d} P(f_i|y) &= \text{argmax}_y \log \left(P(y) \prod_{i=1}^{d} P(f_i|y) \right) \textbf{ (1 points)} \\
&= \text{argmax}_y \log P(y) + \sum_{i=1}^{d} \log P(f_i|y) \textbf{ (1 points)} \\
&= \text{argmax}_y \log P(y) + \sum_{i=1}^{d} [f_i \log P(f_i = 1|y) + (1 - f_i) \log P(f_i = 0|y)] \textbf{ (1 points)} \\
&= \text{argmax}_y \log P(y) + \sum_{i=1}^{d} \log P(f_i = 0|y) + \sum_{i=1}^{d} f_i \log \frac{P(f_i = 1|y)}{P(f_i = 0|y)} \textbf{ (1 points)}
\end{aligned}$$

    This is clearly equivalent to a linear classifier with the weights:

$$\begin{aligned}
w_{y,0} &= \log P(y) + \sum_{i=1}^{d} \log P(f_i = 0|y) \textbf{ (3 points)} \\
w_{y,i} &= \log \frac{P(f_i = 1|y)}{P(f_i = 0|y)} \text{ for } i = 1, ..., d \textbf{ (5 points)}
\end{aligned}$$

    Note that if the answer for $w_{y,i}$ is correct but doesn't specify $i = 1, ..., d$, 1 point is deducted.

This page intentionally left blank.

15. If we are given the dataset as in Table 2, and f_1, f_2 are both binary features. Can we use a naive Bayes classifier to correctly classify all four data? **(3 points)**

f_1	f_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Table 2: Four data and their labels.

Describe the classifier to prove if your answer is yes, or briefly justify to disprove if your answer is no. (you answer should be within 3 lines)

Ans:

No. **(1 points)**

The justification can be set into two parts:

- The data (XOR) is clearly not linearly separable, which has been shown in theory assignment 1. **(1 points)**
- Since we know from problem (a) that a naive Bayes classifier on binary-valued features can be re-written as a linear classifier, no Naive Bayes classifier can thus achieve zero classification error for the data in Table 2. **(1 points)**

4 Logistic Regression (25 points)

Given a dataset $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), ..., (\boldsymbol{x}_N, y_n)\}$ where $\boldsymbol{x}_n \in \mathbb{R}^D$, and $y_n \in \{0, 1\}$. Consider this prediction model

$$P(y_n = 1|\boldsymbol{x}_n; \boldsymbol{w}) = \Phi(\boldsymbol{w}^T \boldsymbol{x}_n),$$

where

$$\Phi(z) = \int_{-\infty}^{z} \phi(v)dv,$$

and

$$\phi(z) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right).$$

The shape of $\Phi(z)$ is very similar to the sigmoid activation function we used in logistic regression. Because $\Phi(z)$ is called the *probit function*, we thus call this model *Probit Regression*.

16. The cross-entropy loss of a binary classifier over a dataset is defined as follows:

$$H(y, p) := -\sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \big(y_n \ln p_n + (1 - y_n) \ln(1 - p_n)\big),$$

where $p_n = P(y_n = 1|\boldsymbol{x}_n; \boldsymbol{w})$. Our goal is to minimize cross-entropy in our binary classification problem. Please derive $\nabla_{\boldsymbol{w}} H(y, P(y|\boldsymbol{x}, \boldsymbol{w}))$, express it with $y_n, \boldsymbol{x}_n, \Phi(\cdot)$ and $\phi(\cdot)$, and reduce it to the simplest form. **(15 points)**

Ans: Denote $\boldsymbol{w}^T \boldsymbol{x}_n$ by z_n. Denoting logarithm by ln or log are both fine.

$$\nabla_{\boldsymbol{w}} \boldsymbol{w}^T x_n = \boldsymbol{x}_n,$$
$$\frac{\partial}{\partial \Phi} \log \Phi(z) = \frac{1}{\Phi(z)},$$
$$\frac{\partial}{\partial \Phi} \log(1 - \Phi(z)) = \frac{-1}{1 - \Phi(z)}$$

$$\frac{\partial}{\partial z} \Phi(z) = \phi(z),$$
$$\frac{\partial}{\partial z} \log \Phi(z) = \frac{\phi(z)}{\Phi(z)},$$
$$\frac{\partial}{\partial z} \log(1 - \Phi(z)) = \frac{-\phi(z)}{1 - \Phi(z)}$$

(6 points)

$$\nabla_{\boldsymbol{w}} y_n \ln p_n = \nabla_{\boldsymbol{w}} y_n \ln \Phi(z) = \frac{\phi(z_n)}{\Phi(z_n)} y_n \boldsymbol{x}_n$$

$$\begin{aligned} \nabla_{\boldsymbol{w}} (1 - y_n) \ln(1 - p_n) &= \nabla_{\boldsymbol{w}} (1 - y_n) \ln(1 - \Phi(z)) \\ &= (1 - y_n) \frac{-\phi(z_n)}{1 - \Phi(z_n)} \boldsymbol{x}_n \end{aligned}$$

(5 points)

$$\Rightarrow \nabla_{\boldsymbol{w}} H(y, P(y|\boldsymbol{x}, \boldsymbol{w})) = -\sum_{n=1}^{N} \big(\frac{y_n}{\Phi(\boldsymbol{w}^T \boldsymbol{x}_n)} - \frac{1-y_n}{1-\Phi(\boldsymbol{w}^T \boldsymbol{x}_n)}\big)\phi(\boldsymbol{w}^T \boldsymbol{x}_n)\boldsymbol{x}_n$$
$$= \sum_{n=1}^{N} \frac{\Phi(\boldsymbol{w}^T \boldsymbol{x}_n) - y_n}{\Phi(\boldsymbol{w}^T \boldsymbol{x}_n)(1-\Phi(\boldsymbol{w}^T \boldsymbol{x}_n))}\phi(\boldsymbol{w}^T \boldsymbol{x}_n)\boldsymbol{x}_n$$

(4 points)

Solutions that match either line get full marks.
The following are difficult without any hints.

$$\text{If } y_n = 1, \;\; \Rightarrow \Phi(z) - y_n = -(1-\Phi(z)) \Rightarrow \frac{\Phi(z) - y_n}{\Phi(z)(1-\Phi(z))} = \frac{-1}{\Phi(z)}.$$

$$\text{If } y_n = 0, \;\; \Rightarrow \Phi(z) - y_n = \Phi(z) \Rightarrow \frac{\Phi(z) - y_n}{\Phi(z)(1-\Phi(z))} = \frac{1}{1-\Phi(z)} = \frac{1}{\Phi(-z)},$$

where the last equation comes from the fact that $1 - \Phi(z) = \Phi(-z)$ (similar to sigmoid function).
Thus,

$$\Rightarrow \nabla_{\boldsymbol{w}} H(y, P(y|\boldsymbol{x}, \boldsymbol{w})) = \sum_{n=1}^{N} \frac{1-2y_n}{\Phi((2y_n-1)\boldsymbol{w}^T \boldsymbol{x}_n)}.\phi(\boldsymbol{w}^T \boldsymbol{x}_n)\boldsymbol{x}_n$$

(0 points)

Grading comments:

- 0: (no comment): the solution does not make any sense and does not have any reasonable derivation.
- 1-3: Partial credits: some of the basic derivatives are correct.
- 4: basic derivatives: the basic derivatives are the only things that are correct in the solution.

Depending on how wrong the derivation of $\Phi(\cdot)$ goes:

- -5: wrong d Phi
- -3: common mistake in d Phi

Depending on how far the solution is from the final form. Common mistakes include lack of sum/negative sign, incomplete reduction, or anything alike.

- -4: wrong reduction
- -2: erroneous reduction

Other cases:

- 15: (no comment): the solution is considered correct with sufficient details.
- -5: missing derivations: no proper derivations are shown even if the final answer is correct; solutions that get this comment usually present the final answer in a single line.

Note that these comments may combine, indicating the degree of correctness.

This page intentionally left blank.

17. The activation function that the prediction model uses determines how much the model is *sensitive* to outliers. Outliers are non-typical data points that deviates far away from typical ones with the same label. For example, if the data points $\boldsymbol{x}_n \in \mathcal{R}$ with label $y_n = 1$ are mostly within the range $[2, 4]$, then a data point with value 10 is considered an outlier.

Suppose that we add an outlier $\boldsymbol{x}_{N+1}$ to the dataset $\mathcal{D}$. Please derive $\nabla_{\boldsymbol{x}_{N+1}} H(y, P(y|\boldsymbol{x}, \boldsymbol{w}))$ for Probit Regression. **(10 points)**

Note that the cross-entropy is now a summation over $N+1$ points, i.e.,

$$H(y, p) := -\sum_{n=1}^{N+1} \big(y_n \ln p_n + (1 - y_n) \ln(1 - p_n)\big),$$

Ans:

$\nabla_{\boldsymbol{x}_{N+1}} \boldsymbol{w}^T x_{N+1} = \boldsymbol{w}$ **(1 points)**

$\nabla_z \ln \Phi(z) = \frac{1}{\Phi(z)} \phi(z) = \frac{\phi(z)}{\Phi(z)}$ **(1 points)**

$\nabla_z \ln(1 - \Phi(z)) = \frac{1}{1-\Phi(z)}(-\phi(z)) = \frac{-\phi(z)}{1-\Phi(z)}$ **(2 points)**

$$\begin{aligned}
\nabla_{\boldsymbol{x}_{N+1}} H(y, p) &= \nabla_{\boldsymbol{x}_{N+1}} - \sum_{n=1}^{N+1} \big(y_n \ln p_n + (1 - y_n) \ln(1 - p_n)\big) \\
&= -\nabla_{\boldsymbol{x}_{N+1}} \big(y_n \ln \Phi(z_{N+1}) + (1 - y_n) \ln(1 - \Phi(z_{N+1}))\big) \quad \textbf{(2 points)} \\
&= -\big(y_n \nabla_{\boldsymbol{x}_{N+1}} \ln \Phi(z_{N+1}) + (1 - y_n) \nabla_{\boldsymbol{x}_{N+1}} \ln(1 - \Phi(z_{N+1}))\big) \\
&= -\big(y_n \frac{\phi(z)}{\Phi(z)} \boldsymbol{w} + (1 - y_n)(\frac{-\phi(z)}{1 - \Phi(z)}) \boldsymbol{w}\big) \\
&= -\big(\frac{y_n}{\Phi(z)} \phi(z) \boldsymbol{w} - \frac{1 - y_n}{1 - \Phi(z)} \phi(z) \boldsymbol{w}\big) \\
&= \frac{\Phi(z) - y_{N+1}}{\Phi(z)(1 - \Phi(z))} \phi(z) \boldsymbol{w} \quad \textbf{(4 points)}
\end{aligned}$$

where $z = \boldsymbol{w}^T \boldsymbol{x}_{N+1}$. The derivations are similar to Problem (a). Note that the summation is gone because we are taking derivative w.r.t a single data point.

5 Neural Network (20 points)

Consider the following neural network, LeNet-5, that consists of two convolution layers (rectangles with 'conv'), two average-pooling layers and three fully connected layers (right most three rectangles). The neural net takes image of size $(32 \times 32 \times 1)$ and outputs a prediction vector of probabilities for 10 classes.

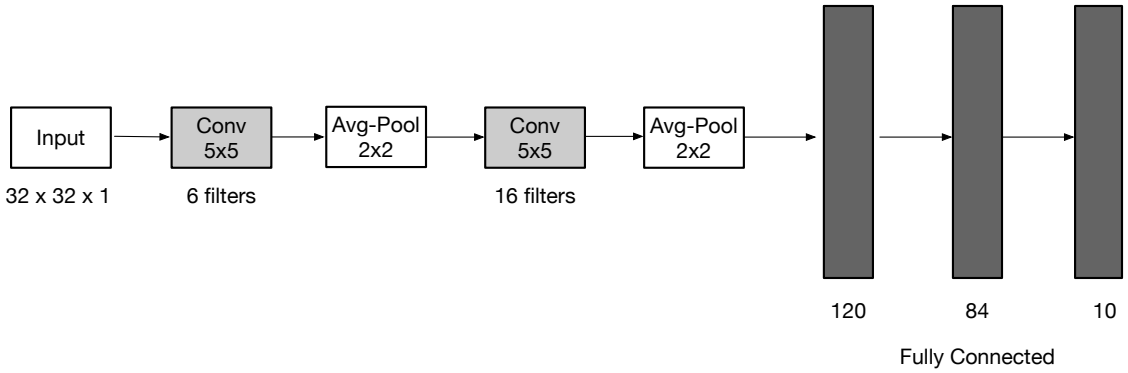


Figure 1: LeNet-5

18. For all convolutional layers, padding is zero, stride is 1. There is no bias in all layers. How many parameters do we need to learn for this network? (Your answer should tell your calculation process step by step). **(10 points)**

Ans: C1: $1 \times 5 \times 5 \times 6 = 150$ **(2 points)**
C3: $6 \times 5 \times 5 \times 16 = 2400$ **(2 points)**
F5: $16 \times 5 \times 5 \times 120 = 48000$ **(2 points)**
F6: $120 \times 84 = 10080$ **(2 points)**
O1: $84 \times 10 = 840$ **(2 points)**
the total parameter size is 61470.

Suppose we have a binary-class Convolutional Neural Network defined below.

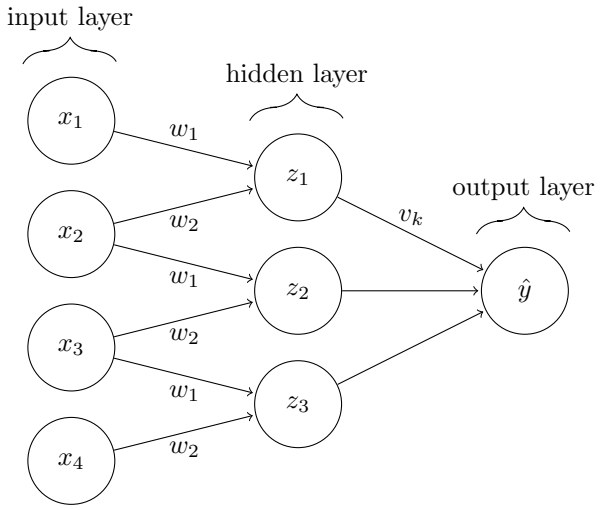


Figure 2: A neural network with one hidden layer.

For binary classification, the forward propagation can be expressed as:

input layer	x_i	(2)
hidden layer	$z_k = ReLU(w_1 x_k + w_2 x_{k+1})$, where $ReLU(x) = max\{0, x\}$	(3)
output layer	$\hat{y} = \sigma(\sum_{k=1}^{3} v_k z_k)$, where $\sigma(z) = \frac{1}{1+\exp(-z)}$	(4)
loss function	$L(y, \hat{y}) = -[y \log \hat{y} + (1-y)\log(1-\hat{y})]$, where $\hat{y}$ is prediction, y is ground truth	(5)

19. Please write down $\frac{\partial L}{\partial v_k}$ and $\frac{\partial L}{\partial w_1}$ in terms of only x_k, z_k, v_k, y, and/or $\hat{y}$ using backpropagation. **(10 points)**

    Hint: the derivative of the $ReLU$ function is $H(a) = \mathbb{I}[a > 0]$. You can directly use $H(a)$ in your answer.

    Ans:

$$\frac{\partial L}{\partial v_k} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial v_k} \quad \text{2 point}$$

$$\frac{\partial L}{\partial \hat{y}} = \frac{1-y}{1-\hat{y}} - \frac{y}{\hat{y}} \quad \text{1 point}$$

$$\frac{\partial \hat{y}}{\partial v_k} = \frac{\partial}{\partial v_k}\left[\sigma(\sum_{k=1}^{3} v_k z_k)\right]$$
$$= \sigma(\sum_{k=1}^{3} v_k z_k)\left[1 - \sigma(\sum_{k=1}^{3} v_k z_k)\right] z_k$$
$$= \hat{y}\,(1-\hat{y})\,z_k \qquad \text{1 point}$$

$$\rightarrow \frac{\partial L}{\partial v_k} = \left(\frac{1-y}{1-\hat{y}} - \frac{y}{\hat{y}}\right)\hat{y}\,(1-\hat{y})\,z_k \qquad \text{1 point}$$
$$= (\hat{y} - y) z_k$$

$$\frac{\partial L}{\partial w_1} = \sum_{k=1}^{3} \frac{\partial L}{\partial z_k}\frac{\partial z_k}{\partial w_1} \qquad \text{2 point}$$
$$\frac{\partial L}{\partial z_k} = (\hat{y} - y)\,v_k \qquad \text{1 point}$$
$$\frac{\partial z_k}{\partial w_1} = \frac{\partial}{\partial w_1} max(0, w_1 x_k + w_2 x_{k+1})$$
$$= H(z_k) x_k \qquad \text{1 point}$$
$$\rightarrow \frac{\partial L}{\partial w_1} = \sum_{k=1}^{3} (\hat{y} - y) v_k H(z_k) x_k \qquad \text{1 point}$$

You may use this page as scratch paper, but nothing written on it will be graded.