CSCI 567 Spring 2019 Practice Exam
DO NOT OPEN EXAM UNTIL INSTRUCTED TO DO SO

Name	
ID #	Solutions

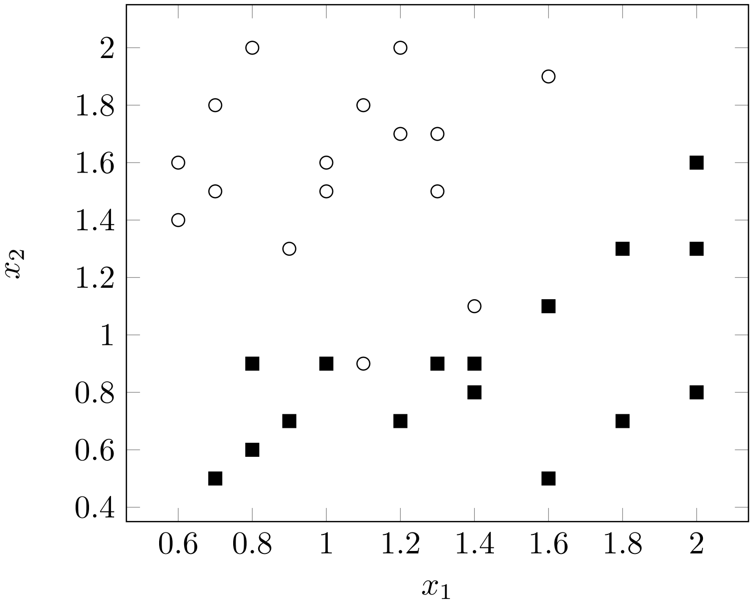
Read the following instructions carefully; you will be expected to conform to them during the exam.

- Put your ID number on the Scantron form and bubble in the values. Scantrons where we cannot match to your ID will result in **zero points for any multiple choice questions, even if the answers are clear on the packet.**
- Multiple choice questions might not be weighted equally. Partial credit may be available on some. Responses to multiple-choice questions must be filled in on your Scantron. No credit on these questions will be given for responses in the booklet.
- Questions should be answered **concisely**. Excessive writings will incur mark reduction. Derivations should be short — no need to prove/derive extensively. No derivations will require more than 10 lines.
- Write **legibly** so the grader can read your answers without misunderstandings. Avoid cursive writings.
- You must answer each free response question on the page provided. We will not provide blank new copies of pages.
- The duration of the quiz is **3 hour**. Please budget your time on each question accordingly.
- You **may not** leave your seat during the exam **for any reason**. If you leave your seat, you may be required to submit your exam at that moment.
- The quiz has a total of **11 (eleven) physical pages**, including True/False questions, multiple choice questions, and free response questions. Each question may have sub-questions. Once you are permitted to open your exam (and not before), you should count the pages to ensure that you have the right number.
- This is a **closed-book** exam. Consulting classmates, the Internet, or other resources is NOT permitted. You may not use any electronics *for any reason* or say anything to a classmate *for any reason.*
- There are some spaces in the packet that can be used for scratch work. No additional scratch paper will be provided.
- When the proctor tells you the test is over, you must *immediately* cease writing, close the packet, and look either forward or upward. Continuing to write after the exam is over will be reported for academic discipline, including at minimum an F **in the class**.

Section	Points Available
Decision Trees/CNN/Kernel/KNN	30
Naïve Bayes	10
Logistic Regression	20
Linear Regression	20
Neural Network	20
Total	100

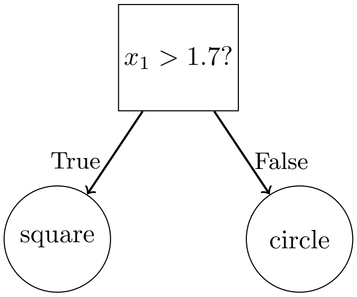
1 Decision Tree, CNN, Kernel & KNN

Questions 1 through 2 deal with the following data, where squares, and circles are two different classes of data.

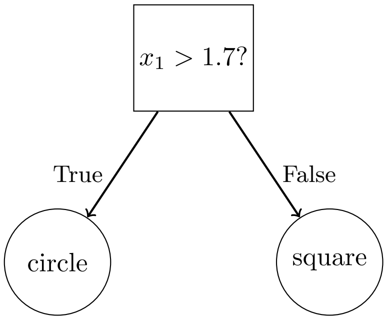


1. Suppose we use the data as training data, which of the following decision trees has the smallest training error? **(3 points)**

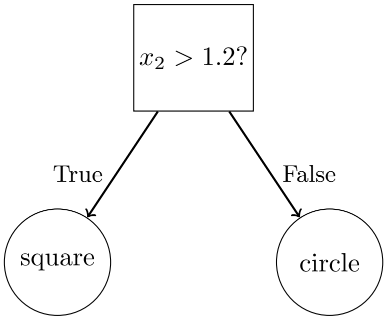
(a)



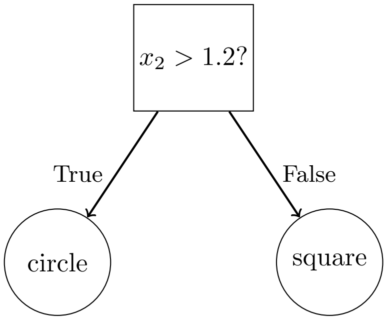
(b)



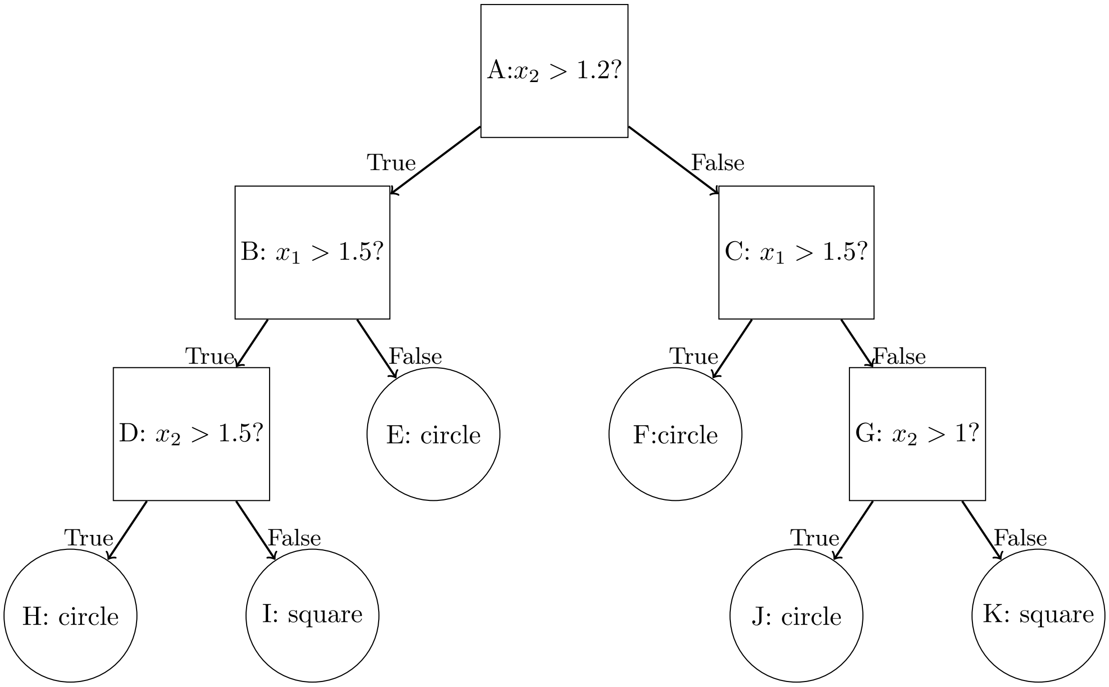
(c)



(d)



↑ 3 squares and 2 circles are wrongly classified



2. Suppose we use the data as validation data and apply reduced-error pruning to the decision tree above, pruning which of the following node will yield the best validation accuracy? **(3 points)**

(a) Node C ← 3 samples wrongly classified

(b) Node D

(c) Node G

(d) Node J

3. Suppose a convolution layer takes a $4 \times 6 \times 3$ 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) Two $2 \times 4 \times 3$ filters, stride 1, no zero-padding.

(b) Two $1 \times 1 \times 3$ filters, stride 2, 1 zero-padding.

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

(d) Six $2 \times 4 \times 3$ filters, stride 2, 1 zero-padding.

D

4. How many parameters do we need to learn for the following network structure? An $8 \times 8 \times 3$image input, followed by a convolution layer with 2 filters of size 2×2 (stride 1, no zero-padding), then another convolution layer with 4 filters of size 3×3 (stride 2, no zero-padding), and finally a max-pooling layer with a 2×2 filter (stride 1, no zero-padding). (Note: the depth of all filters are not explicitly spelled out, and we assume no bias/intercept terms are used.) **(3 points)**

(a) 96

(b) 44

(c) 100

(d) 48

A. $2 \times (2 \times 2 \times 3) + 4 \times (3 \times 3 \times 2) = 96$

5. Which of the following is wrong about neural nets? **(3 points)**

(a) A fully connected feedforward neural net without nonlinear activation functions is the same as a linear model.

(b) Dropout technique prevents overfitting.

(c) A neural net with one hidden layer and a fixed number of neurons can represent any continuous function.

(d) A max-pooling layer has no parameters to be learned.

c

6. Which is *not* a valid kernel function, for samples x and y and kernels $k_1(\cdot,\cdot)$ and $k_2(\cdot,\cdot)$? **(3 points)**Answer: B

(a) $k(x, y) = 5$

(b) $k(x, y) = x + y$

(c) $k(x, y) = e^{x+y}$

(d) $k(x, y) = \langle x, y\rangle^3 + (\langle x, y\rangle + 1)^2$

7. Suppose we apply the kernel trick with a kernel function k to the nearest neighbor algorithm (with L2 distance in the new feature space). What is the nearest neighbor of a new data point $\boldsymbol{x}$ from a training set $S = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$? **(3 points)** Answer: D.

(a) $\arg\min_{\boldsymbol{x}_n \in S}\ k(\boldsymbol{x}_n, \boldsymbol{x}_n) + k(\boldsymbol{x}, \boldsymbol{x}) + 2k(\boldsymbol{x}_n, \boldsymbol{x})$

(b) $\arg\min_{\boldsymbol{x}_n \in S}\ k(\boldsymbol{x}_n, \boldsymbol{x})$

(c) $\arg\min_{\boldsymbol{x}_n \in S}\ (k(\boldsymbol{x}_n, \boldsymbol{x}) - k(\boldsymbol{x}, \boldsymbol{x}_n))^2$

(d) $\arg\min_{\boldsymbol{x}_n \in S}\ k(\boldsymbol{x}_n, \boldsymbol{x}_n) - 2k(\boldsymbol{x}_n, \boldsymbol{x})$

True or False

8. k-Nearest Neighbor (kNN) classifier is a parametric machine learning model parameterized by k. **(3 points)**

False. A parametric model assumes a finite set of of parameters, e.g., $\boldsymbol{w}$ in logistic regression that linearly combines an input vector. The hyperparameter k in kNN is not a parameter in this sense.

9. Given a multi-class classification dataset $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), ..., (\boldsymbol{x}_N, y_n)\}$ where every $\boldsymbol{x}_n$ is unique, one can always build a kNN classifier and a decision tree that both achieve the same training error. **(3 points)**

True, because under this circumstance, it is possible for both to achieve zero training error.

10. Given two binary classification datasets A and B with the same number of points and dimensionality. Dataset A and B have the same sizes of testing set, too. The best k of a kNN classifier on test set of Dataset A is also the best k when kNN is applied to Dataset B. **(3 points)**

False. k is problem-specific.

2 Naïve Bayes

Suppose we are given the following data, where $A, B, C \in \{0,1\}$ are random variables, and y is a binary output whose value we want to predict.

A	B	C	y
0	0	1	0
0	1	0	0
1	1	0	0
0	0	1	1
1	1	1	1
1	0	0	1
1	1	0	1

How would a naive Bayes classifier predict y if the input is $\{A = 0, B = 0, C = 1\}$. Assume that in case of a tie the classifier always prefers to predict 0 for y. **(10 points)**

The classifier will predict 1.

$$\begin{aligned} P(y=0) &= 3/7 \\ P(y=1) &= 4/7 \\ P(A=0|y=0) &= 2/3 \\ P(B=0|y=0) &= 1/3 \\ P(C=1|y=0) &= 1/3 \\ P(A=0|y=1) &= 1/4 \\ P(B=0|y=1) &= 1/2 \\ P(C=1|y=1) &= 1/2 \end{aligned}$$

Predicted y maximizes $P(A=0|y)P(B=0|y)P(C=1|y)P(y)$:

$$\begin{aligned} P(A=0|y=0)P(B=0|y=0)P(C=1|y=0)P(y=0) &= \frac{2}{63} = 0.0317 \\ P(A=0|y=1)P(B=0|y=1)P(C=1|y=1)P(y=1) &= \frac{1}{28} = 0.0357 \end{aligned}$$

Hence, the predicted y is 1.

3 Logistic Regression

We have introduced two types of denotation for logistic regression in this course. In Theory Assignment 2, we represented the two classes using 0 and 1 and minimized cross-entropy loss; in lecture notes, we represented the two classes using -1 and 1 and minimized logistic loss. In the following questions, we will ask you to show that the two denotations are equivalent.

Given a dataset $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_N, y_n)\}$ where $\mathbf{x}_n \in \mathcal{R}^D$, and $y_n \in \{-1, 1\}$. We wish apply logistic regression to $\mathcal{D}$; in other words, we wish to learn a linear combination $\mathbf{w}$ that combines features so that we can apply a sigmoid activation $\sigma(z) = \frac{1}{1+\exp -z}$ to predict the probability of the output label.

3.1 Predictive model (4 points)

Let $\mathbf{w}$ be the weight that absorbs the bias term b. Please express the predictive model of logistic regression $P(y_n|\mathbf{x}_n, \mathbf{w})$ without if-else clause.

$$
\begin{aligned}
P(y_n|\mathbf{x}_n, \mathbf{w}) &= \begin{cases} \sigma(\mathbf{w}^T\mathbf{x}_n), & \text{if } y_n = 1 \\ 1 - \sigma(\mathbf{w}^T\mathbf{x}_n) = \sigma(-\mathbf{w}^T\mathbf{x}_n), & \text{if } y_n = -1 \end{cases} \\
&= \sigma(y_n\mathbf{w}^T\mathbf{x}_n).
\end{aligned}
$$

3.2 Cross-entropy Loss (4 points)

The cross-entropy loss of a binary dataset is defined as follows:

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

where $q_n, p_n \in [0, 1]$. In our problem, $p_n := P(y_n = 1|\mathbf{x}_n, \mathbf{w})$. Our goal is to minimize cross-entropy in our binary classification problem. Let $q_n = \frac{y_n+1}{2}$. Please derive $\nabla_{\mathbf{w}} H(q, P(y|\mathbf{x}, \mathbf{w}))$, express it with $q_n, \mathbf{x}_n$ and $\sigma(\cdot)$, and reduce it to the simplest form.

$$
\begin{aligned}
\nabla_{\mathbf{w}} H(q, P(y|x, \mathbf{w})) &= \nabla_{\mathbf{w}} - \sum_{n=1}^{N} \big(\, q_n \ln p_n + (1 - q_n) \ln(1 - p_n)\big) \\
&= -\sum_{n=1}^{N} \nabla_{\mathbf{w}} \big(\, q_n \ln \sigma(\mathbf{w}^T\mathbf{x}_n) + (1 - q_n) \ln \sigma(-\mathbf{w}^T\mathbf{x}_n)\big) \\
&= -\sum_{n=1}^{N} \big(\, q_n \nabla_{\mathbf{w}} \ln \sigma(\mathbf{w}^T\mathbf{x}_n) + (1 - q_n) \nabla_{\mathbf{w}} \ln \sigma(-\mathbf{w}^T\mathbf{x}_n)\big)
\end{aligned}
$$

Gradient is a linear operator, so it can be distributed to the two terms. Denote $\mathbf{w}^T\mathbf{x}_n = z_n$, we have:

$$
\begin{aligned}
\nabla_{\mathbf{w}} \ln \sigma(\mathbf{w}^T\mathbf{x}_n) &= (\nabla_{z_n} \ln \sigma(z_n))(\nabla_{\mathbf{w}} \sigma(\mathbf{w}^T\mathbf{x}_n)) \\
&= (\frac{1}{\sigma(z_n)})(\sigma(z_n)(1 - \sigma(z_n))\mathbf{x}_n) \\
&= (1 - \sigma(z_n))\mathbf{x}_n.
\end{aligned}
$$

The last line is by $\nabla_z \sigma(z) = \sigma(z)(1 - \sigma(z))$.

Similarly,

$$\begin{aligned}\nabla_{\mathbf{w}} \ln \sigma(-\mathbf{w}^T \mathbf{x}_n) &= \frac{1}{\sigma(-z_n)}(\sigma(-z_n)(1-\sigma(-z_n)))(-1)\mathbf{x}_n \\ &= -(1-\sigma(-z_n))\mathbf{x}_n \\ &= -(\sigma(z_n))\mathbf{x}_n.\end{aligned}$$

The last line is by $\sigma(-z) = 1 - \sigma(z)$.

Combining the two terms, we have:

$$\begin{aligned}&q_n \nabla_{\mathbf{w}} \ln \sigma(\mathbf{w}^T \mathbf{x}_n) + (1-q_n)\nabla_{\mathbf{w}} \ln \sigma(-\mathbf{w}^T \mathbf{x}_n) \\ =&q_n(1-\sigma(z_n))\mathbf{x}_n + (1-q_n)(-1)\sigma(z_n)\mathbf{x}_n \\ =&\big(q_n - q_n\sigma(z_n) - (\sigma(z_n) - q_n\sigma(z_n))\big)\mathbf{x}_n \\ =&(q_n - \sigma(z_n))\mathbf{x}_n\end{aligned}$$

We then reach our final results (replacing $z_n = \mathbf{w}^T \mathbf{x}_n$):

$$\begin{aligned}\Rightarrow \nabla_{\mathbf{w}} H(q, P(y|x, \mathbf{w})) &= -\sum_{n=1}^{N}(q_n - \sigma(\mathbf{w}^T \mathbf{x}_n))\mathbf{x}_n \\ &= \sum_{n=1}^{N}(\sigma(\mathbf{w}^T \mathbf{x}_n) - q_n)\mathbf{x}_n \quad \text{(either is fine.)}\end{aligned}$$

3.3 Iterative Update Formula of the Weight (4 points)

Let the learning rate be λ. Please express gradient descent update formula of the weight using $P(y_n|\mathbf{x}_n, \mathbf{w})$, $\mathbf{x}_n$, and y_n.

$$\begin{aligned}(q_n - \sigma(\mathbf{w}^T \mathbf{x}_n))\mathbf{x}_n &= \begin{cases}1 - \sigma(\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n, & \text{if } y_n = \ \ 1 \\ -\sigma(\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n, & \text{if } y_n = -1\end{cases} \\ &= \begin{cases}\sigma(-\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n, & \text{if } y_n = \ \ 1 \\ -\sigma(\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n, & \text{if } y_n = -1\end{cases} \\ &= \begin{cases}(+1)\sigma((-1)\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n, & \text{if } y_n = \ \ 1 \\ (-1)\sigma((+1)\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n, & \text{if } y_n = -1\end{cases} \\ &= y_n\sigma(-y_n\mathbf{w}^T \mathbf{x}_n)\mathbf{x}_n \\ &= P(-y_n|\mathbf{x}_n, \mathbf{w})y_n\mathbf{x}_n\end{aligned}$$

$$\Rightarrow \mathbf{w}_{k+1} = \mathbf{w}_k + \lambda\sum_{n=1}^{N} P(-y_n|\mathbf{x}_n, \mathbf{w})y_n\mathbf{x}_n$$

In the lecture, we said that logistic regression is a soft version of perceptron because the shape of sigmoid activation function is like a smoothed version of the sign function and that the update formulae look similar. Answer the following 2 questions.

3.4 Predictive Model (4 points)

The predictive model of a perceptron take the form: $\hat{y}_n = \text{sign}(\mathbf{w}^T \mathbf{x}_n)$. It applies a sign function to predict the label from the linear combination of input features $\mathbf{x}_n$. Explain why we cannot replace the sigmoid activation with a sign function and optimize the loss using gradient descent.

The derivative of a sign function is zero almost everywhere. Consequently, GD never update the weight and thus fails to minimize the loss.

3.5 Perceptron Loss (4 points)

$\nabla_w \sum_{n=1}^{N} L(y_n \mathbf{w}^T \mathbf{x}_n) = \sum_{n=1}^{N} -\mathbb{I}(y_n \mathbf{w}^T \mathbf{x}_n)$, where $L(z) = \max(0, -z)$. Describe or illustrate why the derivative of the perceptron loss function takes the form of an indicator function $\mathbb{I}(\cdot)$.).

Max function is piecewise linear. If the input is less than 0, the slope (hence derivative) of it is -1; otherwise, the slope is zero. The resulting derivative is a step function, and it is thus convenient to express the derivative as an indicator function.

4 Linear Regression

In this problem, we prove that if we are using the Newton's method to solve the least square optimization problem, then it only takes one step to converge. Recall that the Newton's method update the parameters as follow:

$$w^{t+1} = w^t - H^{-1}\nabla L(w^t)$$

where $H = \nabla^2 L(w^t)$ is the Hessian matrix of the loss function, i.e., $H_{ij} = \frac{\partial}{\partial w_i \partial w_j} L(w^t)$.

(1) Find the Hessian of least square loss function: $L(w) = \frac{1}{2}\sum_{n=1}^{N}(w^T x_n - y_n)^2$. **(10 points)**

$$\begin{aligned} \frac{\partial}{\partial w_j} L(w) &= \sum_{n=1}^{N}(w^T x_n - y_n)x_{nj} \\ \frac{\partial^2}{\partial w_i \partial w_j} L(w) &= \sum_{n=1}^{N} x_{ni}x_{nj} = (X^T X)_{ij} \end{aligned}$$

Therefore, $H = \nabla^2 L(w) = X^T X$.

(2) Show that given any w^0, after first iteration of Newton's method, we obtain the optimal $w^* = (X^T X)^{-1}X^T y$. **(10 points)**

$$\begin{aligned} w^1 &= w^0 - H^{-1}\nabla L(w^0) \\ &= w^0 - H^{-1}X^T(Xw^0 - y) \\ &= w^0 - w^0 + (X^T X)^{-1}X^T y \\ &= (X^T X)^{-1}X^T y \end{aligned}$$

5 Neural Network (20 points)

Suppose we have a Neural Network defined as below. An illustration is provided in the Figure below. Please answer the following questions.

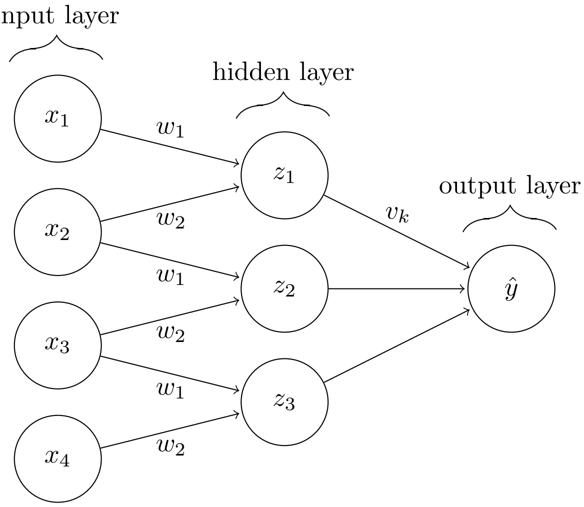


Figure 1: A neural network with one hidden layer.

Forward Propagation. The forward propagation can be expressed as:

$$\begin{aligned} &\text{input layer} && x_i, && (1)\\ &\text{hidden layer} && z_k = tanh\left(w_1 x_k + w_2 x_{k+1}\right), tanh(\alpha) = \frac{\mathrm{e}^{\alpha} - \mathrm{e}^{-\alpha}}{\mathrm{e}^{\alpha} + \mathrm{e}^{-\alpha}} && (2)\\ &\text{output layer} && \hat{y} = \sum_{k=1}^{3} v_k z_k && (3)\\ &\text{loss function} && L(y, \hat{y}) = ln(1 + exp(-y\hat{y})),\ \text{where } \hat{y} \text{ is prediction, } y \text{ is ground truth} && (4)\end{aligned}$$

Backpropagation Please write down $\frac{\partial L}{\partial v_k}$ and $\frac{\partial L}{\partial w_1}$ in terms of only x_k, v_k, z_k, y, and/or $\hat{y}$ using backpropagation.

Hint: $\frac{\partial \tanh(\alpha)}{\partial \alpha} = 1 - [\tanh(\alpha)]^2$.

The solution is:

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

$$\frac{\partial L}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}} \left[ln(1 + exp(-y\hat{y}))\right] \qquad \text{2 point}$$

$$= -\frac{y exp(-y\hat{y})}{1 + exp(-y\hat{y})} \qquad \text{2 points}$$

$$= -\sigma(-y\hat{y})y$$

$$= (\sigma(y\hat{y}) - 1)y$$

$$\frac{\partial \hat{y}}{\partial v_k} = \frac{\partial \sum_{k=1}^{3} v_k z_k}{\partial v_k} = z_k$$ 2 points

$$\rightarrow \frac{\partial L}{\partial v_k} = (\sigma(y\hat{y}) - 1) y z_k$$ 2 point

$$\frac{\partial L}{\partial w_1} = \sum_{k=1}^{3} \frac{\partial L}{\partial z_k} \frac{\partial z_k}{\partial w_1}$$ 2 point

$$\begin{aligned} \frac{\partial L}{\partial z_k} &= \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_k} \\ &= (\sigma(y\hat{y}) - 1) y v_k \end{aligned}$$ 2 point

2 points

$$\begin{aligned} \frac{\partial z_k}{\partial w_1} &= \frac{\partial}{\partial w_1} \tanh\left(w_1 x_k + w_2 x_{k+1}\right) \\ &= \left(1 - z_k^2\right) x_k \end{aligned}$$ 2 points

$$\rightarrow \frac{\partial L}{\partial w_1} = \sum_{k=1}^{3} (\sigma(y\hat{y}) - 1) y v_k \left(1 - z_k^2\right) x_k$$ 2 point

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