(PRINT) Name _____Student No_____

Signature _____Total Mark_____/100

University of Waterloo

Computer Science 480/680 – Machine Learning

Midterm Test
2019 June 21
Time: 8:35 pm – 9:50 pm

Time: 75 minutes
Total marks: 100

Answer all questions on this paper. This is an open book exam in which notes, lecture slides, textbooks and non-programmable calculators are permitted. However, computers, tablets, smart phones and smart watches are not permitted.

This examination has 8 pages. Check that you have a complete paper.

1	**/ 22**
2	**/ 18**
3	**/ 20**
4	**/ 8**
5	**/ 12**
6	**/ 20**
Total	**/ 100**

Question 1 [22 pts] Logistic regression.

a) **[6 pts]** Suppose that you trained a logistic regression model on some training data and the resulting weights are $w_0 = 1, w_1 = 2, w_2 = -3$. Assuming two classes (+ and -), a data point is predicted to belong to class + when $\sigma(\boldsymbol{w}^T\boldsymbol{x} + w_0) \geq 0.5$. Classify the following data points:

i) $(1,2)^T$ $\quad (1 \quad 2 \quad -3)\begin{pmatrix}1\\1\\2\end{pmatrix} = -3 \quad \Rightarrow \quad -$ **(2pts)**

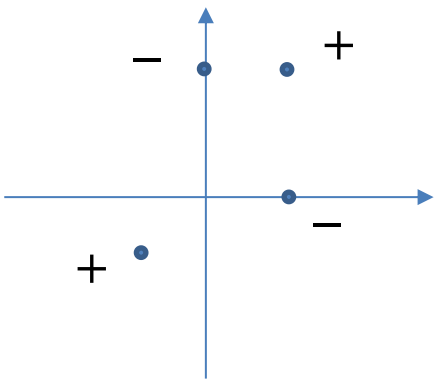
ii) $(2,1)^T$ $\quad (1 \quad 2 \quad -3)\begin{pmatrix}1\\2\\1\end{pmatrix} = 2 \quad \Rightarrow \quad +$ **(2pts)**

iii) $(0.5,-1)^T$ $\quad (1 \quad 2 \quad -3)\begin{pmatrix}1\\0.5\\-1\end{pmatrix} = 5 \quad \Rightarrow \quad +$ **(2pts)**

b) **[8 pts]** Consider the following data set:

$$\begin{array}{ll} x_1 = (1,2)^T & y_1 = + \\ x_2 = (-1,-1)^T & y_2 = + \\ x_3 = (0,2)^T & y_3 = - \\ x_4 = (1,0)^T & y_4 = - \end{array}$$

where the first two points belong to class + and the last two points belong to class −. Is it possible for a logistic regression classifier to correctly classify all points in this dataset? If yes, give weights that ensure correct classification? If no, explain why and describe an approach that could be used to modify the logistic regression classifier to correctly classify all those data points?



No **(2pts)**, the dataset is not linearly separable **(2pts)**.

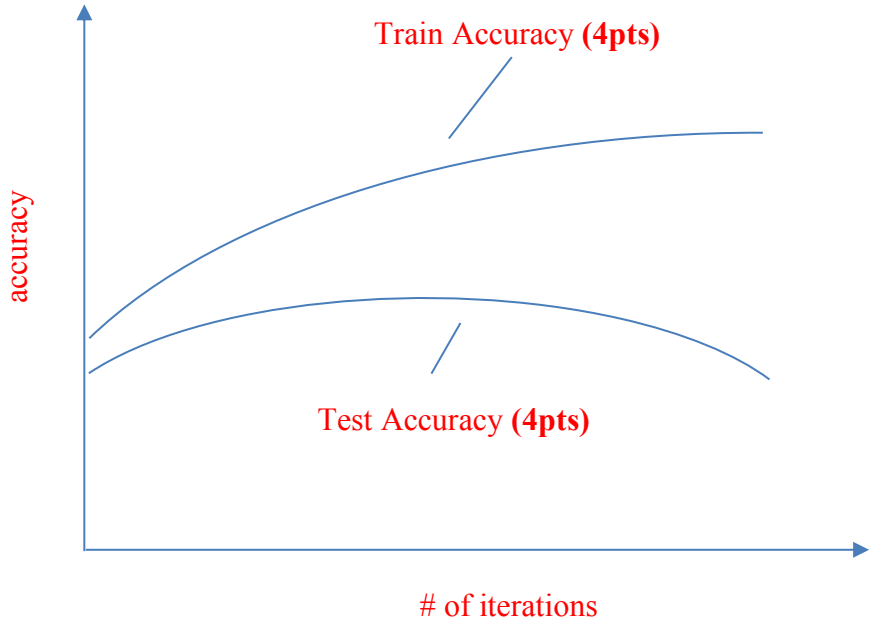
Define a non-linear mapping ϕ that maps the data into a new space that is linearly separable **(3pts)**.

Question 1 (continued)

c) **[8 pts]** Consider maximum likelihood (without any regularization) as the training objective for logistic regression:

$$\min_{\boldsymbol{w}} -\sum_{n} y_n \ln \sigma(\boldsymbol{w}^T \bar{\boldsymbol{x}}_n) + (1 - y_n) \ln(1 - \sigma(\boldsymbol{w}^T \bar{\boldsymbol{x}}_n))$$

Suppose that you train the classifier by Newton's method. Draw a graph with two curves that show how the training accuracy (first curve) and the test accuracy (second curve) are expected to vary with the number of iterations of Newton's method.



Question 2 [18 pts] Perceptron.

a) [9 pts] Consider a threshold perceptron with a single unit. Suppose that the weights are initialized to $w_0 = 1, w_1 = 0, w_2 = 0$. Show the weights at each step of the threshold perceptron learning algorithm. A step corresponds to updating the weights based on one data point in the training set:

$$\boldsymbol{x}_1 = (1,1)^T, \quad y_1 = 0$$
$$\boldsymbol{x}_2 = (1,2)^T, \quad y_2 = 1$$
$$\boldsymbol{x}_3 = (-1,3)^T, \quad y_3 = 1$$

Loop through the training set once in the following order.

Starting weights: $w_0 = 1, \quad w_1 = 0, \quad w_2 = 0$

$-1 \qquad -1 \qquad -1$

Update weights based on $\boldsymbol{x}_1 = (1,1)^T, y_1 = 0$: $w_0 = 0 \quad w_1 = -1 \quad w_2 = -1$

$1 \qquad 1 \qquad 2$

Update weights based on $\boldsymbol{x}_2 = (1,2)^T, \ y_2 = 1$: $w_0 = 1 \quad w_1 = 0 \quad w_2 = 1$

Update weights based on $\boldsymbol{x}_3 = (-1,3)^T, y_3 = 1$: $w_0 = 1 \quad w_1 = 0 \quad w_2 = 1$

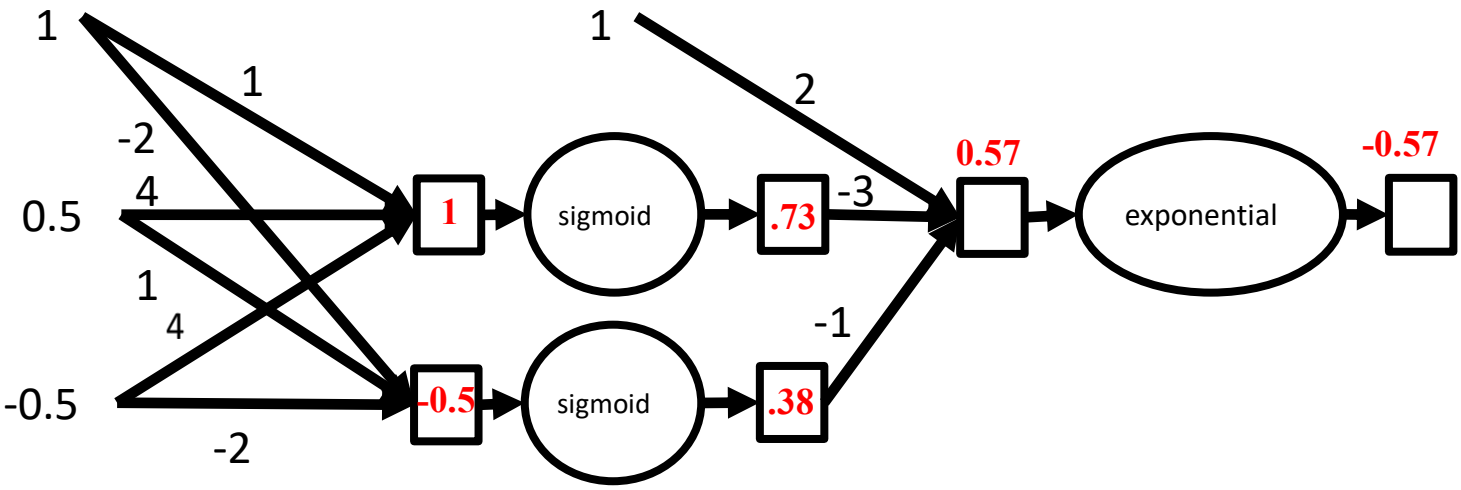
1pt per weight

b) [9 pts] The threshold perceptron can only deal with two classes. How would you modify it to handle more than two classes? Describe your approach.

Create one output node per class and replace the threshold activation function by a softmax activation function. Train by backpropagation.

Question 3 [20 pts] Neural networks.

a) **[8 pts]** Forward propagation: Compute the input (a_j) and output (z_j) of each unit j of the following neural network by filling in the empty boxes.



b) **[12 pts]** Consider the neural network above. What activation function would you use in the output node for each of the following scenarios?

i) binary classification where the outputs are 0 or 1:

Threshold (2 pts for sigmoid since it is used for binary classification, but it does not output 0 or 1)

ii) multiclass classification where the outputs are confidence scores in the form of probabilities:

softmax

iii) regression where the output is a height measurement that is always positive:

exponential, ReLU

iv) regression where the output may be any real number:

identity

Question 4 [8 points] K-nearest neighbours

Consider the K-nearest neighbour algorithm where the following rule is used to predict the label of a new data point x.

$y_x \leftarrow mode(\{y'_x | x' \in knn(x)\})$
where $knn(x) = k$ nearest neighbours of x
y_x is the label of x

How would you modify this rule to return a confidence score in the form of a probability for each class?

$$\Pr(y|x) = \frac{|\{x'|x' \in knn(x) \wedge y'_x = y\}|}{k}$$

For each class y, return the ratio of the # of neighbours that belong to class y divided by k

Question 5 [12 points] Kernels

Let $\phi_1(\boldsymbol{x})$ and $\phi_2(\boldsymbol{x})$ be the features for the kernel

$$k_1(\boldsymbol{x}, \boldsymbol{x}') = \phi_1(\boldsymbol{x})\phi_1(\boldsymbol{x}') + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}')$$

Similarly, let $\phi_3(\boldsymbol{x})$ and $\phi_4(\boldsymbol{x})$ be the features for the kernel

$$k_2(\boldsymbol{x}, \boldsymbol{x}') = \phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}')$$

a) **[6 pts]** What are the features for the kernel $k_3(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}') + k_2(\boldsymbol{x}, \boldsymbol{x}')$?

$$k_1(\boldsymbol{x}, \boldsymbol{x}') + k_2(\boldsymbol{x}, \boldsymbol{x}') = \phi_1(\boldsymbol{x}') + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}') + \phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}')$$

$$\boldsymbol{\phi}(\boldsymbol{x}) = \begin{pmatrix} \phi_1(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x}) \\ \phi_3(\boldsymbol{x}) \\ \phi_4(\boldsymbol{x}) \end{pmatrix} \quad \textbf{(6pts)}$$

b) **[6 pts]** What are the features for the kernel $k_4(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}')k_2(\boldsymbol{x}, \boldsymbol{x}')$?

$$\begin{aligned} k_1(\boldsymbol{x}, \boldsymbol{x}')k_2(\boldsymbol{x}, \boldsymbol{x}') \\ = \phi_1(\boldsymbol{x})\phi_1(\boldsymbol{x}')\phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_1(\boldsymbol{x})\phi_1(\boldsymbol{x}')\phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}') \\ + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}')\phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}')\phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}') \end{aligned}$$

$$\boldsymbol{\phi}(\boldsymbol{x}) = \begin{pmatrix} \phi_1(\boldsymbol{x})\phi_3(\boldsymbol{x}) \\ \phi_1(\boldsymbol{x})\phi_4(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x})\phi_3(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x})\phi_4(\boldsymbol{x}) \end{pmatrix} \quad \textbf{(6pts)}$$

Question 6 [20 pts] Indicate whether each statement is true or false. No justification required.

a) **[4 pts]** The computational complexity of generalized linear regression in the feature space is cubic in the number of basis functions while the computational complexity of generalized linear regression in the dual space is cubic in the amount of data.

T

b) **[4 pts]** In binary classification by mixtures of Gaussians with identical covariance matrices, the posterior distribution is given by a logistic sigmoid.

T

c) **[4 pts]** Linear regression and logistic regression are special cases of neural networks without any hidden unit.

T

d) **[4 pts]** When measurement noise is Gaussian, linear regression by maximum likelihood yields a different (equal) solution than linear regression by minimum squared loss.

F

e) **[4 pts]** In K-nearest neighbours, using too many neighbours might lead to overfitting (underfitting).

F