CSE428: Image Processing

Problem Set 3

Linear & Logistic Regression

Question 1

Solve **Example Problem 2** from **Lecture 7.1**.

Question 2

Solve **Example Problem 1** from **Lecture 8.1**.

Linear Models and Neural Networks

Question 3

Suppose you have a database, stored as a .csv file, on which you want to perform a supervised learning algorithm for **simultaneous regression** and **classification**. The first 3 rows of the dataset are shown below. Column **y** is the target for performing the **regression** task and column **z** is the label for performing the **classification** task on this dataset.

x1	x2	x3	y	z
10	124	-139	0.4	0
5	398	-112	0.32	1
3	312	-172	-0.19	0

a) Is any preprocessing required on this dataset before performing supervised learning? Explain briefly.

b) Suppose you are using linear regression to predict **y** given **x1**, **x2**, and **x3**. Write the structure of the **hypothesis function (Model)** and the **cost function**.

c) Among the following two sets of parameters, which one is the "better" model? Why?

$i)$ $\theta = [\theta_0, \theta_1, \theta_2, \theta_3] = [0, 0.1, 0.008, 0.001]$

$ii)$ $\theta = [\theta_0, \theta_1, \theta_2, \theta_3] = [0, 0.5, 0.008, 0.002]$

d) Write down the matrix equations for the forward pass (from x_1, x_2 and x_3 to the output of the **softmax**) of a **neural network** used for solving the classification problem **with one hidden layer of 4 neurons**.

Neural Networks

Question 4

Suppose a shallow dense neural network with a single hidden layer of only **2** neurons has been trained for a binary **dark** (label: **0**) vs **light** (label: **1**) image classification task. After sufficient training period, you want to test the performance of the network with a **2 x 2** input image. The pixel intensity values of the input image and the weight-bias parameters for the layers are given below:

0.7	0.8
0.6	0.9

Input **to** ***Hidden*** layer weight and bias parameters:

$$W_{input-hidden} = \begin{bmatrix} 0.1 & 0.5 & 0.3 & 0.1 \\ 0.2 & 0.4 & 0.09 & 0.3 \end{bmatrix}$$

$$b_{hidden} = \begin{bmatrix} -0.32 \\ -0.28 \end{bmatrix}$$

Hidden **to** ***Output*** layer weight and bias parameters:

$$W_{hidden-output} = \begin{bmatrix} 0.5 & 0.9 \end{bmatrix}$$

$$b_{output} = [0.5]$$

The activation functions used in the **Hidden** and **Output** layers are given in the following table:

Layer	**Activation function**
Hidden	tanh
Output	sigmoid

a) Explain why linear activation functions are not ideal for neural networks.

b) **Draw the architecture** of the neural network to be used for the above classifier.

c) Determine the **outputs** of individual layers **using the given parameters** and **predict the classification label** of the test image determined by the neural network. *[Hint: Output at any layer can be determined using the following equation:]*

$$output = activation(W \times input + b)$$

Data Preparation, Hyperparameter Tuning & Evaluation

Question 5

Answer the questions from **Practical Scenarios 1 & 2** in **Lecture 8.2**.

Question 6

Answer the questions from **Practical Scenarios 3, 4, 5 & 9** in **Lecture 8.2**.

CNN

Question 4

Alice is a BRACU student and she is taking **CSE428** this semester. For her final project, she is trying to implement a CNN architecture for a classification task that comprises of the following layers:

Layer	**Input Dimensions**	**Filter Size**	**#Filters** or, **#Neurons**	**Padding**	**Output Dimensions**	**#Params**
Conv1	128 * 128 * 3	7*7	8	2		
MaxPool1		2*2	—	0		
Conv2		5*5	16	2		
MaxPool2		2*2	—	0		
Conv3		3*3	32	0		
AvgPool3		4*4	—	0		
Flatten		—	—	—		
FC		—	256	—		
FC		—	128	—		
FC (Output)		—	4	—		

In the table above, *Conv-X* denotes a **Convolutional** layer, *Pool-X* denotes a **Pooling** layer and *FC* denotes a **Fully Connected** layer.

a. Determine the number of classes and the activation function used in the final layer.

b. Calculate the **input** and **output dimensions** for each of the layers. (Complete the 2nd and 6th columns of the table).

c. Calculate the **number of Parameters** for each of the layers. (Complete the last column of the table).

d. Repeat Question **b** considering a Mini-Batch size of **32** instead of individual inputs.

e. Suppose, Alice used **Batch Normalization** layers after each **Convolutional** and **Fully Connected** layer. Would it Change the total **number of trainable parameters**? If **Yes**, then, by **how much**?