Sheet 5

Q1) Using a batch training, write a pseudocode algorithm for training a shallow neural network that has three output nodes using the perceptron criterion. You must clarify the inputs and outputs as well as the data structures used in your code.

Input (X_m, y_m), m = 1,2,3,.....,N
Initialize W as a matrix with random gaussian values (Mean=0, Variance=1) and shape of (k, d+1), where k is the number of output nodes and d is the number of nodes from the previous layer.
Alpha = 1, epsilon = 0.01
Repeat
    delta = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
    For m = 1 to N do
        S = argmax($W.X_m$) # Index of the highest Node
        If S != y_m do
            $delta_s$ = $delta_s$ + X_m
            $delta_y$ = $delta_y$ - X_m
    delta = delta / N
    W_{i+1} = W_i - alpha*delta

Q2) Using a batch training, write a pseudocode algorithm for training a shallow neural network that has three output nodes using the SVM criterion. You must clarify the inputs and outputs as well as the data structures used in your code.

Input (X_m, y_m), m = 1,2,3,.....,N
Initialize W as a matrix with random gaussian values (Mean=0, Variance=1) and shape of (k, d+1), where k is the number of output nodes and d is the number of nodes from the previous layer.
Alpha = 1, epsilon = 0.01
Repeat
    delta = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
    For m = 1 to N do
        id = 0

For r = 1 to K do
If $\max(1+W_r.X_m-W_y.X_m, 0)$ and r != y do
Id += 1
$delta_r = delta_r + X_m$
$delta_y = delta_y - id*X_m$

delta = delta / N
$W_{i+1} = W_i - alpha*delta$

Q3) Using a batch training, write a pseudocode algorithm for training a shallow neural network that has three output nodes using the multinomial regression criterion. You must clarify the inputs and outputs as well as the data structures used in your code.

Input (X_m, y_m), m = 1,2,3,.....,N
Initialize W as a matrix with random gaussian values (Mean=0, Variance=1) and shape of (k, d+1), where k is the number of output nodes and d is the number of nodes from the previous layer.
Alpha = 1, epsilon = 0.01
Repeat
delta = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
For m = 1 to N do
$V = W.X_m$
Y_hat = exp(v)/sum(exp(v)) # output after softmax in a vector form
$delta = delta + Y\_hat*X_m$
$delta_y = delta_y - (1-Y\_hat_y)*X_m$

delta = delta / N
$W_{i+1} = W_i - alpha*delta$

Q4) Write a pseudocode algorithm to normalize training data as explained in the course lectures.

Initialize Mean & Variance with a vector of zeros in the same shape as the input sample.
Epsilon = 0.00001

For i = 1 to N do
    mean = mean + X_i/N
For i = 1 to N do
    variance = variance + (X_i - mean)^2/N

For i = 1 to N do
    X_hat_i = (X_i-mean)/sqrt(variance+epsilon)

Q5) Derive and visualize the classification boundary for the following given data sets using a shallow neural network:-

x_1	0	0	1	2	2	3	0	0	1
x_2	0	1	0	0	1	0	3	4	3
y	1	1	1	2	2	2	3	3	3

Codes for this question in sheet5_codes …
Q5_multiclass_preceptron_critera.py

Q6) Derive and visualize the classification boundary for the following given data sets using a shallow neural network:-

x_1	3	4	2	3	3	-3	-2	-4	-3	-3	0	1	-1	0	0	0	1	-1	0	0
x_2	0	0	0	1	-1	0	0	0	1	-1	3	3	3	4	2	-3	-3	-3	-2	-4
y	1	1	1	1	1	2	2	2	2	2	3	3	3	3	3	4	4	4	4	4

Codes for this question in sheet5_codes …
Q6_multiclass_preceptron_critera.py

Q7) Repeat the problems of 5 and 6 using the SVM GD criterion.

Codes for this question in sheet5_codes …
Q7_multiclass_SVM_critera.py

Q8) Repeat the problems of 5 and 6 using the multinomial logistic regression GD criterion.

Codes for this question in sheet5_codes …
Q8_multiclass_logistic_critera.py