Birla Institute of Technology and Science, Pilani

Work Integrated Learning Programmes Division

M. Tech. in AIML

II Semester 2022-2023

Mid-Semester Test
(EC2 - Regular)

Course Number	AIMLCZG511
Course Name	Deep Neural Networks
Nature of Exam	Open Book
Weight-age for grading	30
Duration	2 hrs
Date of Exam	

* Pages	3
* Questions	4

1. (a) You'd like to train a fully-connected neural network with 5 hidden layers, each with 10 hidden units. The input is 20-dimensional and the output is a scalar. What is the total number of trainable parameters in your network? [1]

   (b) You would like to train a dog/cat image classifier using mini-batch gradient descent. You have already split your dataset into train, dev and test sets. The classes are balanced. You realize that within the training set, the images are ordered in such a way that all the dog images come first and all the cat images come after. A friend tells you: "you absolutely need to shuffle your training set before the training procedure." Is your friend, right? Justify. [1]

   (c) You want to evaluate the classifier you trained in part(b). Your test set (X_{test}, Y_{test}) is such that the first m_1 images are of dogs, and the remaining m_2 images are of cats. After shuffling (X_{test}, Y_{test}), you evaluate your model on it to obtain a classification accuracy α_1%. You also evaluate your model on (X_{test}, Y_{test}) without shuffling to obtain accuracy α_2%. What is the relationship between α_1%.and α_2%. $(\geq, \leq, =, <, >)$? Justify. [1]

   (d) You are designing a deep learning system to detect driver fatigue in cars. It is crucial that that your model detects fatigue, to prevent any accidents. Which of the following is the most appropriate evaluation metric: Accuracy, Precision, Recall, Loss Value. Justify your choice. [1]

   (e) You want to solve a classification task. You first train your network on 20 samples.Training converges, but the training loss is very high. You then decide to train this network on 10,000 examples. Is your approach to fixing the problem correct? If yes, explain the most likely results of training with 10,000 examples. If not, give a solution to this problem. [2]

   (f) Justify the use of first momentum and second moment in Adam optimizer. [1]
   (PS: Ensure that you are writing your own answer in one or two lines for all the parts. Inspired or copied answers will be penalized.)

Rubrics

(a) (20+1)*10 + (10+1)*10*4 + (10+1)*1 [1 mark]

(b) The friend is incorrect. The optimization is much harder with minibatch gradient descent because the loss function moves by a lot when going from the one type of image to another. [1 mark]

(c) $\alpha_1 = \alpha_2$. When evaluating on the test set, the only form of calculation that you do is a single metric (e.g. accuracy) on the entire test set. The calculation of this metric on the entire test set does not depend on the ordering. [1 mark]

(d) Recall and justification [0.5 + 0.5 mark]

(e) The model is suffering from bias. Increasing the amount of data reduces the variance, and is not likely to solve the problem. A better approach would be to decrease the bias of the model by maybe adding more layers/ learnable parameters. It is possible that training converged to a local optimum. Training longer/using a better optimizer/ restarting from a different initialization could also work. [2 marks]

(f) First Moment (Mean): It calculates the moving average of the gradients of the parameters over time. By considering the momentum, the optimizer can continue moving in the right direction, even when gradients change rapidly, leading to more stable and consistent updates.
Second Moment (Variance): It calculates the moving average of the squared gradients. Squaring the gradients emphasizes large gradients and dampens small ones. High variance in gradients needs smaller learning rates and low variance may indicate smoother regions where larger learning rates can be safely applied.

2. Given N training data points $\{(x^k, y^k)\}, k = 1 : N, x^k \in \mathcal{R}^d$, and labels in $y^k \in \{-1, 1\}$, we seek a linear discriminant function $f(x) = w \cdot x$ optimizing the loss function $L(z) = e^{-z}$, for $z = yf(x)$.

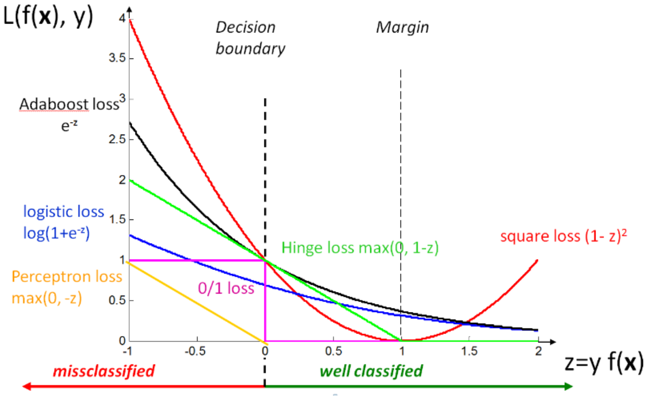
(a) Is $L(z)$ a large margin loss function? Justify your answer with a graphical representation. [2]

(b) Derive the stochastic gradient descent update Δw for $L(z)$. [3]

(c) Represent the computation of the stochastic gradient descent using a computation graph, using general equations. [2]

Rubrics

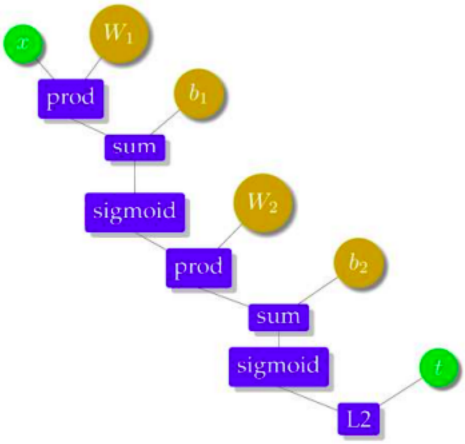
(a) Yes. This is because the loss penalizes even examples that are well classified, but the penalty decreases as you go away from the decision boundary. [1 mark]
Graph [1 mark]



(b) For a learning rate $\eta \geq 0$, and for $z = yf(x) = y\sum_{i=1}^{d} w_i x_i$

$$\begin{aligned}\nabla w_i &= -\eta \frac{\partial L}{\partial w_i} && \text{1 mark}\\ &= -\eta \frac{\partial L}{\partial z}\frac{\partial z}{\partial w_i} && \text{1 mark}\\ &= \eta e^{-z} y x_i && \text{0.5 mark}\\ \nabla w &= \eta e^{-z} y x && \text{0.5 mark}\end{aligned}$$

(c) FP 1 mark and BP (reverse arrows) 1 mark.



3. (a) Consider the DNN model using Adam optimizer with the loss function $L = 3w_1^2 + 1.5w_2^3$ with the weights where $w_1 = 1.5$ and $w_2 = -2$ at time (t-1). Suppose the first moment vector $v = [0, 0]$ and second moment vector $s = [0, 0]$, learning rate = 0.01, decay rate 1=0.5 and decay rate 2 = 0.9 respectively with $epsilon \epsilon = 10^{-8}$. Calculate the weight vector w for time t after the first iteration. [3]

(b) Suppose that the first momentum and second moment updates can be written as the mean and variance of the gradients respectively for not requiring the bias correction in Adam optimizer. Then calculate the weight vector w for time t for the first iteration only using the above parameters mentioned in part a) with $w_1 = 1.5$ and $w_2 = -2$. (Assume the first momentum and second moment values by your own and mention the same while calculation.) [3]

(c) Investigate the differences in the weights w_t obtained in parts a) and b) and provide your analysis. [2]

Rubrics

(a)

$$\nabla w_1 = \frac{\partial L}{\partial w_1} = \frac{\partial(3w_1^2 + 1.5w_2^3)}{\partial w_1} = 6w_1 = 6 * 1.5 = 9$$
$$\nabla w_2 = \frac{\partial L}{\partial w_2} = \frac{\partial(3w_1^2 + 1.5w_2^3)}{\partial w_2} = 4.5w_2^2 = 4.5 * (-2)^2 = 18$$

First Moment [1 mark]

$$\begin{aligned}v_1 &= \beta_1 * v_1 + (1-\beta_1) * \nabla w_1 = 0.5 * 0 + (1-0.5) * 9 = 4.5\\ v_2 &= \beta_1 * v_2 + (1-\beta_1) * \nabla w_2 = 0.5 * 0 + (1-0.5) * 18 = 9\\ v_t &= [4.5, 9]\end{aligned}$$

Second Moment [1 mark]

$$\begin{aligned} s_1 &= \beta_2 * s_1 + (1-\beta_2) * \nabla w_1^2 = 0.9 * 0 + (1-0.9) * 9^2 = 8.1 \\ s_2 &= \beta_2 * s_2 + (1-\beta_2) * \nabla w_2^2 = 0.9 * 0 + (1-0.9) * 18^2 = 32.4 \\ s_t &= [8.1, 32.4] \end{aligned}$$

No bias correction. Only weight update. [1 mark]

$$\begin{aligned} w_{t-1} &= [1.5, -2] \\ w_t &= w_{t-1} - \frac{\eta v}{\sqrt{s+\epsilon}} \\ w_1 &= 1.5 - \frac{0.01 * 9}{\sqrt{8.1 + 10^{-8}}} = 1.468 \\ w_2 &= -2 - \frac{0.01 * 18}{\sqrt{32.41 + 10^{-8}}} = -2.032 \\ w_t &= [1.468, -2.032] \end{aligned}$$

(b) Assume initial v and s vectors and use that for computation.
First Moment [1 mark]

$$\begin{aligned} v &= \beta_1 * v_0 + (1-\beta_1) * \nabla w \\ v_t &= [v_1, v_2] \end{aligned}$$

Second Moment [1 mark]

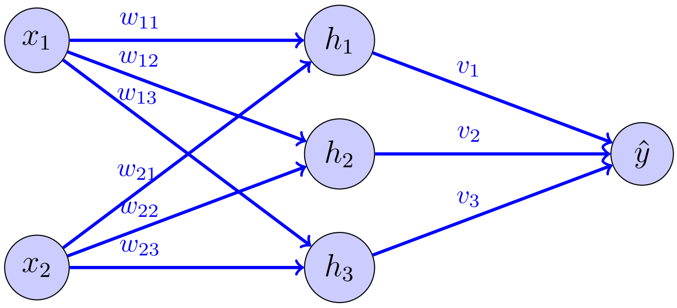
$$\begin{aligned} s &= \beta_2 * s_0 + (1-\beta_2) * \nabla w^2 \\ s_t &= [s_1, s_2] \end{aligned}$$

Apply bias correction and then update weights. [1 mark]

$$\begin{aligned} w_{t-1} &= [1.5, -2] \\ w_t &= w_{t-1} - \frac{\eta v}{\sqrt{s+\epsilon}} \\ w_t &= [w_1, w_2] \end{aligned}$$

(c) May result in faster convergence when non-zero momentums are assumed, but cant be verified for all initial assumptions. [2 mark]

4. Consider the following network structure. You can assume the initial weights. Assume bias to be zero for easier computations. Given that $< x_1, x_2, y > = < 1, 1, 0 >$ where y is the target. Assume $\eta = 0.01$.



(a) Compute the forward propagation and generate the output. Use Relu for hidden layers and Sigmoid activation function for output layer. [2]

(b) Compute the loss and its derivative. You can assume the loss function. [1]

(c) Let the initial weights that assumed be the weights [at time (t-1). Compute the weights v_1, w_{11} and w_{21} at time t using SGD. [1.5]

(d) Assume you are applying drop out for the hidden layer. For odd BITS id, assume that h3 is dropped out and for even BITS id, assume that h2 is dropped out. Computing the weights v_1, w_{11} and w_{21} at time t after dropping out the the neuron. [1.5]

(e) Investigate the differences in the weights w_t obtained in parts c) and d) and provide your analysis. [2]

Rubrics

(a) Award 1 mark if only equations are written. Substitute assumed weights and compute values, then award 2 marks in total. If assumed weights are all same among multiple students, report to IC.

$$\begin{aligned} h1 &= relu(w_{11} * 1 + w_{21} * 1 + 0) && (1)\\ h2 &= relu(w_{12} * 1 + w_{22} * 1 + 0) && (2)\\ h3 &= relu(w_{13} * 1 + w_{23} * 1 + 0) && (3)\\ \hat{y} &= sigmoid(v_1 h_1 + v_2 h_2 + v_3 h_3) \quad \text{Substitute values of eqs: 1,2,3} && (4) \end{aligned}$$

(b) Loss should be either RMSE or binary cross entropy. 0.5 marks for loss computation and 0.5 marks for derivative computation. Writing equations alone will be awarded 0 marks.

$$loss \quad L = \frac{1}{2}(y - \hat{y})^2 \qquad \text{Substitute value of eq: 4} \qquad (5)$$

$$gradient \quad \frac{\partial L}{\partial \hat{y}} = (y - \hat{y}) \qquad \text{Substitute value of eq: 4} \qquad (6)$$

OR

$$loss \quad L = -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$$

$$gradient \quad \frac{\partial L}{\partial \hat{y}} = -\frac{y}{\hat{y}} + \frac{1 - y}{1 - \hat{y}}$$

(c) Computation of weights using SGD. Award 1 mark if only equations are written. If all

3 weights of equations 13, 14, 15 computed, then award 3*0.5 marks.

$$\frac{\partial L}{\partial \hat{z}} = \frac{\partial L}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial \hat{z}} = \frac{\partial L}{\partial \hat{y}} * 1 \qquad \text{Substitute value of eq: 6} \tag{7}$$

$$\frac{\partial L}{\partial v_2} = \frac{\partial L}{\partial \hat{z}} * \frac{\partial \hat{z}}{\partial v_2} = \frac{\partial L}{\partial \hat{z}} * h_2 \qquad \text{Substitute value of eq: 7} \tag{8}$$

$$\frac{\partial L}{\partial h_2} = \frac{\partial L}{\partial \hat{z}} * \frac{\partial \hat{z}}{\partial h_2} = \frac{\partial L}{\partial \hat{z}} * v_2 \qquad \text{Substitute value of eq: 6} \tag{9}$$

$$\frac{\partial L}{\partial z_2} = \frac{\partial L}{\partial h_2} * \frac{\partial h_2}{\partial z_2} = \frac{\partial L}{\partial h_2} * 1 \qquad \text{Substitute value of eq: 9} \tag{10}$$

$$\frac{\partial L}{\partial w_{12}} = \frac{\partial L}{\partial z_2} * \frac{\partial z_2}{\partial w_{12}} = \frac{\partial L}{\partial z_2} * x_1 \qquad \text{Substitute value of eq: 10} \tag{11}$$

$$\frac{\partial L}{\partial w_{22}} = \frac{\partial L}{\partial z_2} * \frac{\partial z_2}{\partial w_{22}} = \frac{\partial L}{\partial z_2} * x_2 \qquad \text{Substitute value of eq: 10} \tag{12}$$

$$v_2 = assumed\, v_2 - \eta \frac{\partial L}{\partial v_2} \qquad \text{Substitute value of eq: 8} \tag{13}$$

$$w_{12} = assumed\, w_{12} - \eta \frac{\partial L}{\partial w_{12}} \qquad \text{Substitute value of eq: 11} \tag{14}$$

$$w_{22} = assumed\, w_{22} - \eta \frac{\partial L}{\partial w_{22}} \qquad \text{Substitute value of eq: 12} \tag{15}$$

(d) Add dropout according to BITS id. If not matching with BITS id, award zero. Recompute the values of $\hat{y}$, loss, gradients and the weight updates.

(e) Check if part d is according to BITS id. If not matching with BITS id, award zero.