Practice Exam: CS 583 Deep Learning

Instructor: Justo E. Karell

Instructions

This practice exam consists of 8 problems. Answer each question, showing all steps. For handwritten work, please upload a clear photo of your work. Alternatively, you can type out your solutions in LaTeX or any document editor. The total points are 100, distributed as indicated.

Problem 1: Mathematical Modeling (10 points)

Problem: You are tasked with predicting house prices based on features such as size, number of rooms, and location. Formulate this problem as a mathematical model. Identify the goal, problem type, assumptions, and an appropriate model for this task. Be sure to clearly define your input variables and output target.

Problem 2: Neural Network Output Calculation (15 points)

Problem: Consider a neural network with:

- Input vector: $\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$
- Hidden layer 1 weights:
$$\mathbf{W}_1 = \begin{bmatrix} 0.5 & -0.3 \\ 0.2 & 0.8 \end{bmatrix}$$
- Hidden layer 1 bias:
$$\mathbf{b}_1 = \begin{bmatrix} 0.1 \\ -0.2 \end{bmatrix}$$
- Hidden layer 2 weights:
$$\mathbf{W}_2 = \begin{bmatrix} 0.4 & 0.1 \\ -0.5 & 0.3 \end{bmatrix}$$
- Hidden layer 2 bias:
$$\mathbf{b}_2 = \begin{bmatrix} 0.05 \\ 0.1 \end{bmatrix}$$

Both hidden layers use the sigmoid activation function:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Task:

1. Write out the formula used to compute the output of each layer, including the activation function.
2. Compute the output of the neural network after the second hidden layer.

Problem 3: Neural Network with Linear Output Layer (15 points)

Problem: Building upon the neural network in Problem 2, we now consider the output layer.

The neural network's final output is simply the output of the second hidden layer without any additional weights or biases (i.e., the identity function is used as the output layer).

Task:

1. Write out the mathematical formula for the neural network's output in terms of the parameters (**W** and **b**), the activation function σ, and the layer outputs $h^{(l)}$.

2. Using your formula, compute the final output of the neural network.

Problem 4: Activation Function: Leaky ReLU (10 points)

Problem: Consider the Leaky ReLU activation function:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0.01x & \text{if } x \leq 0 \end{cases}$$

Task:

1. Compute the output of the Leaky ReLU for the following inputs: $x = -3, -1, 0, 1, 3$.
2. Plot the Leaky ReLU function.

Problem 5: Softmax Function (10 points)

Problem: Given the following vector:

$$\mathbf{v} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$$

Task:

1. Compute the softmax of this vector:

$$\text{softmax}(v_i) = \frac{e^{v_i}}{\sum_{j=1}^{5} e^{v_j}}$$

2. Draw the resulting values on a distribution (e.g., a bar chart).

Problem 6: Convexity of a Vector-Valued Loss Function (20 points)

Problem: Consider the following vector-valued loss function for a simple linear regression model:

$$L(\mathbf{W}) = \|\mathbf{y}_{\text{true}} - \mathbf{X}\mathbf{W}\|^2$$

where:

- $\mathbf{y}_{\text{true}}$ is the vector of true target values.
- $\mathbf{X}$ is the matrix of input features.
- $\mathbf{W}$ is the weight vector.

Task:

1. Prove that this loss function is convex with respect to the weight vector $\mathbf{W}$.
2. Determine the domain of this function.

Problem 7: Convexity of a Loss Function (10 points)

Problem: Prove whether the following loss function is convex:

$$L(w) = \log(1 + e^w)$$

Task: Use the second derivative test to check for convexity.

Problem 8: Backpropagation (20 points)

Problem: Consider a simple neural network with:

- Input vector: $\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$
- Hidden layer weights:
$$\mathbf{W}_1 = \begin{bmatrix} 0.3 & 0.5 \\ -0.6 & 0.2 \end{bmatrix}$$
- Hidden layer bias:
$$\mathbf{b}_1 = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix}$$
- Output layer weights:
$$\mathbf{W}_2 = \begin{bmatrix} 0.4 & -0.3 \end{bmatrix}$$
- Output layer bias: $b_2 = 0.05$

The activation function for both the hidden and output layers is the sigmoid function:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

The loss function is the Mean Squared Error (MSE):

$$L = \frac{1}{2}(y - y_{\text{true}})^2$$

where $y_{\text{true}} = 1$.

Task: Perform one iteration of backpropagation. Compute the forward pass, the gradients, and update the weights and biases for both the hidden and output layers.