MCA572– Neural Networks and Deep Learning

V MCA

01-10-2024

CIA I - LAB TEST

Marks : 50 Time: 2hrs (7.30 -9.30)

Program Evaluation Rubrics

Question 1 (20 Marks):

Writing	(5 Marks)
Implementation	(5 Marks)
Visualization & Documentation	(5 Marks)
Viva	(5 Marks)

Question 2 (30 Marks):

EDA	(5 Marks)
Implementation	(10 Marks)
Visualization & Documentation	(10 Marks)
Viva	(5 Marks)

General Instructions

1. The file you have to save with your name, last 3 digits of register number and program number "Aaron_201_Lab1".
2. The implemented code you have to upload in Github and in the Google Classroom in the given scheduled time.
3. Failure to upload within the allotted time will result in the loss of all marks for the corresponding lab exercise.

Note:

- Download the dataset using any of the following link:
- **For Question 2 A**
  https://docs.google.com/spreadsheets/d/1ckInDsvFlTnHlmUWvCQ9o05_Shi3_swFWxLu8X0_yVs/edit?usp=sharing
  https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment
  For Question 2 B (Dataset is inbuilt in Keras)
  https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews
  https://docs.google.com/spreadsheets/d/1yPJU2HIw2jpZ4nxYfnJiI0m5cF5y9W0Fc6T8DznOu7U/edit?usp=sharing
- Read the questions carefully and see the evaluation rubrics for the marks division.
- Any platform you can use for implementation, better to go with Google Colab with TPU)

Question 1: XOR Gate Classification

I. Write the following in the answer sheet. (5 Marks)

- Write the Threshold function.
- Write Truth table for XOR Gate.
- Write the XOR Gate Classification with input with weight initialization W11 = W21 =W12 = W22 =1.
- Find the Optimum weights of W11, W21, W12, W22, V1 and V2 using the threshold function.
- Why does the Single Layer Perceptron struggle to classify the XOR gate?
- What modifications can be made to the neural network model to handle the XOR gate correctly?

II. Implement the following:

(Implementation 5 marks and Visualization and documentation 5 marks)

- Scenario:
  The XOR gate is known for its complexity, as it outputs 1 only when the inputs are different. This is a challenge for a Single Layer Perceptron since XOR is not linearly separable.
- Lab Task: Attempt to implement a Single Layer Perceptron in Google Colab to classify the output of an XOR gate. Perform the following steps:
- Create the XOR gate's truth table dataset.
- Implement the perceptron model and train it using the XOR dataset using MCP (McCulloch Pitts) Neuron.
- Observe and discuss the perceptron's performance in this scenario.
- Implement XOR using Multi-Layer Perceptron.

Question 2:

A. Sentiment Analysis Twitter Airline

Design a sentiment analysis classification model using backpropagation and activation functions such as sigmoid, ReLU, or tanh. Implement a neural network that can classify sentiment (positive/negative) from a small dataset. Demonstrate how backpropagation updates the weights during the training process. (link Provided at the top of the page to download the dataset)

Task:

- Create a simple feed-forward neural network for binary sentiment classification (positive/negative).
- Use backpropagation to optimize the model's weights based on error calculation.
- Experiment with different activation functions (sigmoid, ReLU, tanh) in the hidden layer and compare the model's performance.
- Evaluate the model on a test set using accuracy and plot the loss over epochs.

OR

B. Sentiment Analysis Using ANN on IMDb Movie Reviews
(Link Provided at the top of the page to download the dataset Not mandatory to take all the instances of the dataset, You can trim the dataset)

You are tasked with performing sentiment analysis on the IMDb movie review dataset using an Artificial Neural Network (ANN). The dataset contains movie reviews labeled as positive or negative.

- Design, implement, and evaluate an ANN model to classify the sentiment of IMDb movie reviews (positive or negative).
- Explain the preprocessing steps, model architecture, and performance evaluation results in your submission.
- Use One or more hidden layers with **ReLU** activation and an output layer with **sigmoid** activation for binary classification.
- Use the **binary cross-entropy** loss function for training the model.
- Provide the source code, the explanation of each step and visualize the necessary EDA steps and results.
- Justify the use of the ReLU and sigmoid activation functions.
- Explain why binary cross-entropy is suitable for this task.