22AIE304 Deep Learning Lab Sheet 2

Fifth Semester BTech CSE(AI)

Department of Computer Science and Engineering

Amrita School of Computing

Neural Networks with McCulloch-Pitts Neurons and Perceptron

Exercise 1: Use MP neurons to build a simple neural network that performs logical operations OR, AND and NOR

Exercise 2: Implement an MP neuron for a binary classification problem using a breast cancer dataset.

- Analyze the effects of scaling on MP Neuron's decision-making process and accuracy. Apply different scaling techniques (min-max normalization, standardization) to the breast cancer dataset features. Train the MP Neuron with these scaled features and compare the model's performance with unscaled data.
- Compare the MP Neuron model's performance with a logistic regression model in accuracy.

Link to practice sheet: MP neuron practice.pdf

Exercise 3: Implement Perceptron for Breast Cancer Classification

a. Load and Explore the Dataset
b. Split the Data into Training and Testing Sets
c. Standardize the Features : Standardize the features (i.e., transform them to have a mean of 0 and a variance of 1) to ensure faster convergence of the Perceptron algorithm.
d. Train the Perceptron Model
e. Make Predictions
f. Evaluate the Model

Link to practice sheet: PerceptronModel_.ipynb

Exercise 4: Implement Perceptron algorithm on the binary Iris dataset and explore its performance by adjusting learning rates and analyzing the weight changes during training

a. Understand the Iris Dataset and write a summary of features

b. Train/ Test Split

c. Implement the Perceptron Algorithm

d. Plot Train/Test Accuracy: Once the model is trained, evaluate the accuracy on both the training and testing datasets. Plot the accuracy for the training and testing data to visualize the model's performance over multiple epochs.

e. Experimenting with Learning Rates

f. Run the Perceptron algorithm with different learning rates.
   - Observe how changing the learning rate impacts the model's ability to converge and its overall accuracy.
   - Interpret the results: Does a higher learning rate lead to faster convergence or instability? Does a lower learning rate affect the speed or quality of the model's learning?

g. Visualizing the Weight Changes

   - During training, the Perceptron's weights are updated in each epoch. To understand how the weights evolve:
     - Create a weight matrix that stores the weight values for each epoch.
     - After each epoch, append the current weights to the matrix.
     - Plot the weights as they change across epochs. This will help visualize how the model adjusts its weights based on the data.
     - Write the Interpretation: After plotting the weight changes, explain how the model's weights stabilize as it learns from the data. Do weights converge?

h. (Optional) Create an Animation of Weight Changes