

## Assignment 2 (Programming)

### Description: Implement a Feedforward Neural Network with Backpropagation

#### Part 1:

**Objective:** Build a feedforward neural network from scratch using NumPy, apply the backpropagation algorithm to train the model for binary classification, and evaluate the model using metrics such as a confusion matrix, recall, and precision.

#### Tasks:

1. **Load and Preprocess Data:**
  - Load the Breast Cancer Wisconsin dataset (available through scikit-learn).
  - Preprocess the data by standardizing the features and splitting it into training and testing sets.
2. **Define Neural Network Architecture:** (this should be done using NumPy only)
  - Implement a feedforward neural network with one hidden layer.
  - Use the sigmoid activation function for the hidden layer and the output layer.
  - Allow customization of the number of input features, hidden units, and output units.
3. **Forward Propagation:**
  - Implement the forward propagation function to calculate the output of the neural network.
4. **Calculate Errors and Loss:**
  - Implement a function to calculate errors (mean squared error) and the loss function.
5. **Backpropagation:**
  - Implement the backpropagation algorithm to update the weights of the neural network using gradient descent.
  - Allow customization of the learning rate and the number of training iterations.
6. **Training Loop with Mini-Batches:**
  - Train the neural network using mini-batch gradient descent.
  - Allow customization of the batch size.
  - Print training loss at regular intervals.
7. **Prediction Function:**
  - Implement a function to make predictions using the trained neural network.
8. **Evaluation:**
  - Apply an evaluation function using a confusion matrix, recall, and precision to evaluate the performance of the network.
9. **Customization:**
  - Allow customization of the neural network's hidden size, learning rate, number of iterations, and batch size.

**Note:** Here are some parameters to use in the code:

**learning rate= learning\_rate = 0.01, and num\_epochs = 1000**

Here's a simplified outline of the Python code structure for this assignment:

```
import numpy as np
from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

# Define sigmoid activation function
def sigmoid(x):
    ...

# Step 2: Define neural network architecture
# Allow customization of input size, hidden size, and output size

# Step 3: Implement forward propagation

# Step 4: Calculate errors and loss

# Step 5: Implement backpropagation
# Allow customization of learning rate and number of iterations

# Step 6: Training loop with mini-batches
# Allow customization of batch size
# Print training loss at regular intervals

# Step 7: Implement prediction function

# Step 8: Evaluation
def evaluate(y_true, y_pred):
    # Implement confusion matrix, recall, and precision calculations
    # Return these evaluation metrics

# Step 9: Customization
# Allow customization of hidden size, learning rate, number of iterations, and batch size
```

## **Part 2:**

Repeat Part 1 using any ML library such scikit-learn, tensorflow, keras, or pytorch.