Cairo University **Faculty of Computers and Artificial Intelligence**

Third year (2022-2023) **Supervised learning (Spring 2023)**

Assignment 3

➢ Delivery Notes:

- This is a group task of 3 members (at most)
- All students should work and fully understand everything in the code.
- **All students should have the same TA section**
- Due date is on May 20th until 11:59 PM
- No Built-in Functions is allowed
- No late submission is allowed.
- Submission will be through google classroom
- No submission through e-mails.
- The submitted files should be named Assignment2_firstStudentID_SecondStudentID_ ThirsStudentID.ipynb
- **Do not send your code** to anyone**,** so that no other student would take your files and submit it under their names.
- **In case of Cheating, you will get a zero grade whether you give the code to someone or take the code from someone or from the Internet**
- Make sure that your notebook **has a clear and visible output** and that your code **is clean and understandable**.

Task:

1. Load MNIST dataset.
2. Standardize your dataset
3. Divide data into training and test.
4. Apply one hot vector for labels (meaning the value is 1 in the correct class and 0 in the rest, there will be 10 classes so a vector of 10).
5. Implement a dynamic Neural Network from scratch.

    - Initialize the weights of the layers with random values.
    - Use equations to calculate the output for all the forward passes.
    - Use the sigmoid function as your activation function for the final output layer and hidden layer.
    - Use MSE as your error function (between the one hot vector and the prediction vector of the NN).
    - Apply back propagation to update the weights.

    Note:

    - Save the output values in each layer as you will need them for the back propagation.

    - Tanh can be used for hidden layers, but this may require more logic handling in your code and is not advised.

    An example for NN with 2 layers: input →**hidden layer 1**→ hidden layer 1 output → **output layer** → output.

6. Function of neural network must follow this format:
   NN (x, y, num_of_layers, size_of_layers)
   Example: NN(X, y, 2, [20, 10])
   where 20 is the size of the hidden layer and 10 is the size of the output layer
   Size of layer means number of neuron at this layer.

7. Test your code with the following architectures and report your different accuracies for each case from the following:
   1- Build NN with only 2 layers => 1 hidden layer and 1 output layer

   2- Build NN with 3 layers=> 2 hidden layers

   Where # of neurons in first layer < # of neurons in second layer and 1 output layer

   3- Build NN with 3 layers=> 2 hidden layers

   Where # of neurons in first layer > # of neurons in second layer