**Name :** Chandrakant Dattatrey Thakare  
**Roll No.:**382014 **PRN:**22310303

**Class :** TY CSE AI **Batch :** B1

**Subject :** DL Lab Assignments

**Assignment : 1**

**Problem Statement**

In this assignment, we implement a Feedforward Neural Network (FNN) using Keras and TensorFlow. The goal is to understand the working of a simple neural network architecture, train it on a dataset, and evaluate its performance. Feedforward Neural Networks are the foundation of deep learning models and provide insights into how inputs are transformed through layers of neurons to produce outputs.

**Task Overview**

1. Build a Feedforward Neural Network using Keras Sequential API.
2. Train the model on a sample dataset.
3. Evaluate the model using accuracy and loss metrics.
4. Visualize training and validation accuracy/loss using Matplotlib.
5. Understand the impact of different activation functions and optimizers.

**Objective**

The objective of this assignment is to:

* Understand the structure of Feedforward Neural Networks.
* Learn how to build and compile a model in Keras.
* Train the model on a dataset and evaluate its performance.
* Visualize the learning process using plots of accuracy and loss.

**Tools and Resources**

* **Software Used:** Google Colab
* **Libraries Used:** TensorFlow, Keras, Numpy, Matplotlib, Seaborn
* **Dataset:** MNIST dataset (handwritten digits classification)

**Methodology**

**Step 1: Import Libraries**

* Import necessary libraries like TensorFlow, Keras, Numpy, and Matplotlib.

**Step 2: Load Dataset**

* Load the MNIST dataset from Keras datasets module.
* Normalize the data to bring pixel values between 0 and 1.

**Step 3: Build the Model**

* Use Keras Sequential API to define a Feedforward Neural Network.
* Add input, hidden, and output layers with appropriate activation functions.

**Step 4: Compile the Model**

* Compile the model with optimizer (e.g., Adam), loss function (categorical crossentropy), and evaluation metric (accuracy).

**Step 5: Train the Model**

* Train the model on training data and validate on validation data.

**Step 6: Evaluate the Model**

* Evaluate the model on test data to measure performance.

**Step 7: Visualization**

* Plot training and validation accuracy/loss over epochs using Matplotlib.

**Results**

The Feedforward Neural Network was successfully trained on the MNIST dataset. The model achieved high accuracy in classifying handwritten digits. Visualization of accuracy and loss curves demonstrated how the model learned over epochs. The final test accuracy was consistent with the training performance, indicating good generalization.

**Conclusion**

In this assignment, we implemented and trained a Feedforward Neural Network using Keras and TensorFlow. We learned how to preprocess data, define a sequential model, compile it with different parameters, and evaluate the results. The exercise provided a foundational understanding of how neural networks work and prepared us for building more advanced deep learning architectures like CNNs and RNNs in future assignments.