

User Manual for Testing Models on CIFAR-10 Dataset

Requirements:

- **Python** (version 3.6 or newer)
- **Libraries:** TensorFlow, NumPy, Matplotlib, Seaborn, Scikit-learn
 - Install the required libraries using:

`pip install tensorflow numpy matplotlib seaborn scikit-learn`

Folder Structure:

- Each model is stored in a separate folder named **Level 1**, **Level 2**, and **Level 3**.
- Inside each folder, you will find:
 - **level1.ipynb** (or **level2.ipynb**, **level3.ipynb** for other folders) - Jupyter notebook containing simulations and outputs.
 - **trained_model_1.h5** (or corresponding model files in other levels) - Trained model file.
 - **test1.py** (or corresponding test files in other levels) - Python script for testing the model.

Step-by-Step Instructions:

1. Reviewing Simulation Results:

- Navigate to the respective folder for the level you wish to review (e.g., **Level 1**).
- Open the Jupyter notebook (**level1.ipynb**) using:

`jupyter notebook level1.ipynb`

- Run the notebook cells to view simulations, training processes, visualizations, and results.

2. Testing the Trained Model:

- Ensure that the path to the trained model file (e.g., **trained_model_1.h5**) is correct in the Python script (**test1.py**).
- Open a terminal or command prompt.
- Navigate to the folder containing the test script.
- Run the test script using Python:

`python test1.py`

- The script will load the trained model and evaluate its accuracy on the test set. The accuracy and other evaluation metrics will be displayed in the terminal.

3. Understanding the Test Script:

- The test script (**test1.py**) typically includes the following steps:
 - Import necessary libraries.
 - Load the test dataset (e.g., CIFAR-10).
 - Preprocess the data similarly to how it was preprocessed for training.
 - Load the trained model from the **.h5** file.
 - Evaluate the model on the test data and print the results.

Notes:

- Ensure all paths (to datasets and model files) in the scripts and notebooks are correctly set according to your directory structure.
- If using a GPU, make sure your TensorFlow installation is configured to support GPU usage.

Troubleshooting:

- **Library Installation Issues:** Ensure you have the correct versions of Python and pip. Re-run the installation commands.
- **Model Loading Errors:** Check the path to the **.h5** model file in the test script. Ensure the model file is not corrupted.
- **Jupyter Notebook Issues:** If the notebook fails to open, check your Jupyter installation or try opening it via Anaconda Navigator.