Assignment 1

SIL8123: Artificial Intelligence for Cybersecurity

Semester I, 2025-2026

Author : Naman Garg, 2025CSY7544

Q. Using the CIFAR-10 dataset, train a CNN model that gives high classification accuracy on the dataset. Then, implement one attack in each of the following types: adversarial attack, training set poisoning attack, membership inference attack, and model inversion attack. In the report, describe the methodology and demonstrate the results in terms of images and appropriate metrics.

0. Training CNN Model

A convolutional neural network (CNN) was trained on the CIFAR-10 dataset using **TensorFlow/Keras**. The model includes multiple convolutional layers with **ReLU activation**, max-pooling, dropout regularization, and L2 weight decay. Data augmentation was applied during training to improve generalization.

The model was trained for **125 epochs** using the **Adam optimizer** with a learning rate of **0.0003** and achieved strong performance on the test set.

For full code, kindly refer the github: https://github.com/namanlp/SIL8123-Assignment-1-2

Evaluation:

- **Test accuracy:** 85%
- **Classification report:**

```
           0       0.83      0.90      0.86      1000
           1       0.89      0.95      0.92      1000
           2       0.84      0.79      0.82      1000
           3       0.82      0.61      0.70      1000
           4       0.85      0.82      0.84      1000
           5       0.85      0.75      0.79      1000
           6       0.80      0.95      0.87      1000
           7       0.84      0.93      0.88      1000
           8       0.91      0.91      0.91      1000
           9       0.89      0.92      0.90      1000

    accuracy                           0.85     10000
   macro avg       0.85      0.85      0.85     10000
weighted avg       0.85      0.85      0.85     10000
```

The trained model is saved as `cifar10_cnn_initial_model.keras` and will be used as the target for various security attacks (adversarial, poisoning, membership inference, and model inversion) in the next phases of the project.

1. Adversarial attack (NewtonFool)

The above CNN classifier was tested against the NewtonFool adversarial attack using a subset of 20 test images. The attack, implemented via the Adversarial Robustness Toolbox (ART), slightly perturbed input images to cause misclassification without visible changes.

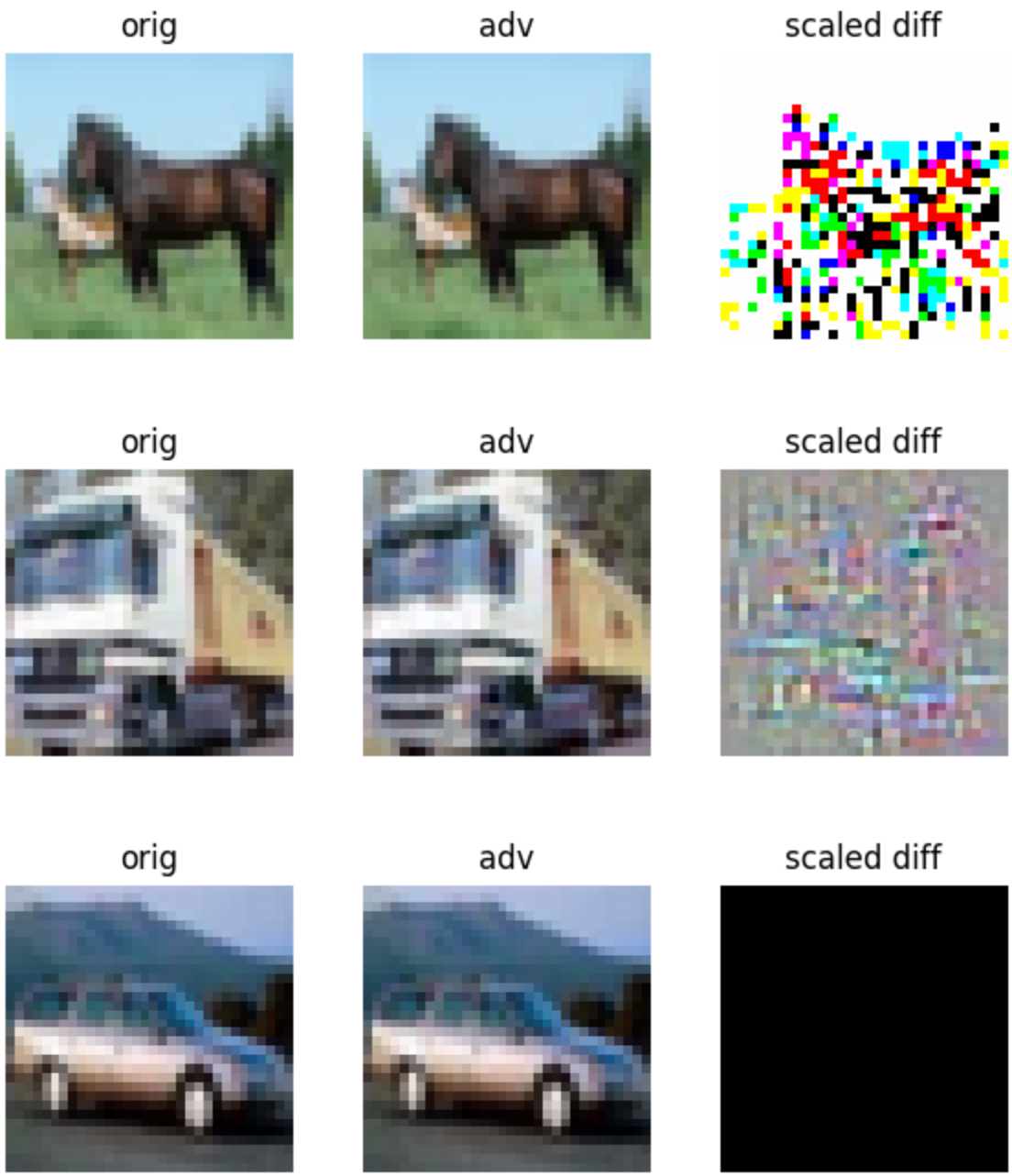
For full code, kindly refer the github: https://github.com/namanlp/SIL8123-Assignment-1-2

Setup:

- **Attack:** NewtonFool (`max_iter`=5, `eta`=0.05)
- **Dataset:** 20 normalized CIFAR-10 test samples
- **Framework:** TensorFlow + ART (TensorFlowV2Classifier)

Results:

- **Clean accuracy:** 100.00%
- **Adversarial accuracy:** 25.00%
- **L2 norm (perturbation):** mean = 3.4893, max = 24.4480
- **L∞ norm:** mean =0.2459, max = 0.9765
- **Visuals saved:** For each sample — original, adversarial, and scaled difference image is saved in `newton_fool_results/`.here are some examples



Hence, we can see that Newton Fool is able to generate the adversarial images that are indifferentiable for human eyes, but could successfully full the model, that is, change the outcome and drastically reduce the prediction accuracy.

2. Training set poisoning attack (Label flipping)

The above CNN classifier was tested against the simple label-flip poisoning strategy to evaluate model robustness when a fraction of the training labels are intentionally corrupted. The poisoning flips each selected sample's one-hot label to a different random class (uniform over the remaining classes). A subset of the poisoned training images was also saved for inspection.

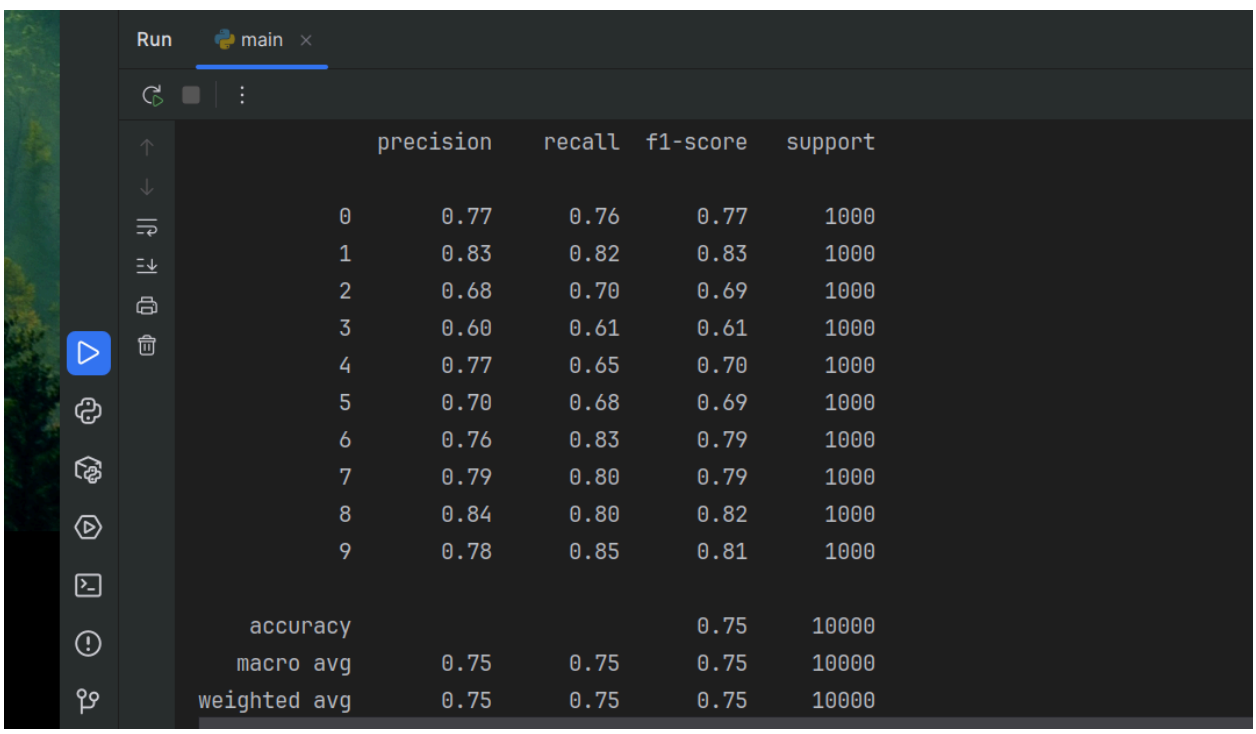
For full code, kindly refer the github: https://github.com/namanlp/SIL8123-Assignment-1-2

Setup:

- **Attack:** Label Flipping (`poison fraction`=20%, `epochs`=125)
- **Dataset:** CIFAR-10
- **Framework:** TensorFlow

Results:

- **Clean accuracy:** 85%
- **Adversarial accuracy:** 75.00%
- **Visuals saved:** Up to 50 poisoned training images saved to `training_set_poison_results/`.with filenames `idx{i}_o{original_label}_p{poisoned_label}.png` for manual inspection.



Run main

	precision	recall	f1-score	support
0	0.77	0.76	0.77	1000
1	0.83	0.82	0.83	1000
2	0.68	0.70	0.69	1000
3	0.60	0.61	0.61	1000
4	0.77	0.65	0.70	1000
5	0.70	0.68	0.69	1000
6	0.76	0.83	0.79	1000
7	0.79	0.80	0.79	1000
8	0.84	0.80	0.82	1000
9	0.78	0.85	0.81	1000
accuracy			0.75	10000
macro avg	0.75	0.75	0.75	10000
weighted avg	0.75	0.75	0.75	10000

Here are some sample:

 => idx148_o5_p1.png

 => idx248_o6_p2.png

 => idx1398_o8_p7.png

 => idx2188_o5_p1.png

3. Membership Inference Attack (Mean-Based Thresholding)

The above CNN classifier was tested against the Membership Inference Attack using the threshold attack using the CIFAR-10 dataset. In this attack, the model's confidence scores (obtained from the softmax output) were used to classify whether a sample belonged to the "member" (training data) or "non-member" (test data) class. A threshold was set based on the mean confidence score, distinguishing members from non-members.

For full code, kindly refer the github: https://github.com/namanlp/SIL8123-Assignment-1-2

Setup:

- **Attack:** Threshold-based attack using the mean of model's confidence scores
- **Threshold calculation:** The mean of the highest probability scores (from softmax output) was used as the threshold to distinguish members and non-members.
- **Framework:** TensorFlow

Results:

- **Accuracy:** 0.5342
- **Precision:** 0.5225832012678289
- **Recall:** 0.7914

4. Membership Inference Attack (Mean-Based Thresholding)

The above CNN classifier was tested against a Model Inversion attack (adversarial-robustness-toolbox Public MIFace) on the CIFAR-10 dataset. In this attack, the adversary uses gradients and optimization to reconstruct representative inputs for target classes from the classifier (i.e., produce images that the model assigns to class k). The reconstructed images are then fed back to the classifier to measure whether the model predicts the intended target labels — a simple sanity/effectiveness check for inversion.

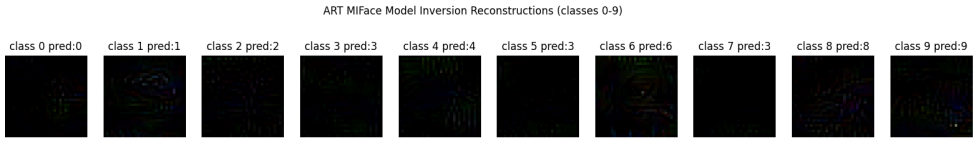
For full code, kindly refer the github: https://github.com/namanlp/SIL8123-Assignment-1-2

Setup:

- **Attack:** MIFace (`max_iter`=150, `batch_size`=1)
- **Initialization:** x=None (start from zero/no prior)
- **Framework:** TensorFlow + ART (TensorFlowV2Classifier)

Results:

- **Clean accuracy:** 85.18%
- **Accuracy on reconstructed images:** 80.00%
- **Visuals:** Reconstructed images from



```
Run    main ×

WARNING: All log messages before absl::InitializeLog() is called are written to S
I0000 00:00:1760093995.221012    8480 gpu_device.cc:2020] Created device /job:loc
I0000 00:00:1760094000.140120    8546 device_compiler.h:196] Compiled cluster usi
Original test accuracy: 85.18%
/home/namang/.local/share/virtualenvs/Assignment_1_2-sOwwdG4p/lib/python3.13/site
  warnings.warn("PyTorch not found. Not importing DeepZ or Interval Bound Propaga
Model inversion:   0%|          | 0/10 [00:00<?, ?it/s]WARNING:tensorflow:Calling
Model inversion: 100%|██████████| 10/10 [00:29<00:00,  2.95s/it]
Predicted labels on reconstructed images: [0 1 2 3 4 3 6 3 8 9]
Accuracy on reconstructed images: 80.00%

Process finished with exit code 0
```

Hence we have performed all the attacks on the given model.

Thank You