INF6422E Advanced Concepts in Computer Security

Practical Work 3 – Winter 2025

Adversarial Machine Learning in Image Classification

Instructions:

- This work is to be submitted in groups and via Moodle only.
- The submitted report must be in pdf format. You're free to build it in any format you want, however (.docx, .odt, .tex, etc.).
- The report must contain a title page including the course title, the lab title, your names, and student ID numbers (matricule).
- The report must be submitted by the 23rd of February 2025 before 23h59. A penalty of 10% will be applied for each day after that date.

Objective:

This lab explores adversarial machine learning techniques, where attackers manipulate inputs to fool deep learning models. We will examine various attacks and defenses using state-of-the-art image datasets.

Dataset

You can use any dataset to conduct this lab, you can select the number of dataset images according to your system capabilities.

> **Option 1**: **MNIST** (Handwritten digits, commonly used for adversarial attack studies).
> **Link**: https://www.kaggle.com/datasets/hojjatk/mnist-dataset

Option 2: CIFAR-10 or ImageNet (More complex datasets for real-world attack scenarios).
Link: https://archive.ics.uci.edu/dataset/693/imagenet

Option 3: GTSRB (Traffic Signs) (For adversarial attacks targeting autonomous vehicles).
Link: https://www.kaggle.com/datasets/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign

Deliverable: Justify your dataset choice based on security concerns, also mention the number of total images you have taken for experimentation.

1. Baseline Image Classification Model [3 Points]

Create a suitable Convolutional Neural Network (CNN) architecture with 2 layers, implement it in PyTorch,
and train it using your dataset. You must implement the complete workflow in your CNN, that is, you cannot use any external libraries, except for the ones mentioned below. Your code must save the created model after training so that you can run a saved model on the test dataset.

The model should be tested in a ratio of (70, 15, and 15) and (60, 20, and 20) for training, validation, and testing, respectively.

- Evaluate using **accuracy, precision, recall, F1-score, and AUC-ROC**.
- **Deliverable**: Model architecture, training performance, evaluation metrics with both ratios as given above, and comparison table.

1.1 Bonus [1 Point]

- **Deliverable:** Show CNN model results with 3 layers using the same selected dataset in ratio of (70, 15, and 15) for training, validation, and testing, respectively.

2. Adversarial Attacks on Machine Learning Models

2.1 Evasion Attacks (FGSM & PGD) [3 Points]

- Implement **Fast Gradient Sign Method (FGSM)** and **Projected Gradient Descent (PGD)** attacks.
- Generate adversarial examples and test model performance.
- **Deliverable**:
    - Visualize adversarial examples.
    - Show how accuracy drops under attack.
    - Discuss trade-offs between attack strength and detectability.

2.2 Data Poisoning Attacks [3 Points]

- Modify a portion of the training dataset (e.g., label flipping or feature manipulation).
- Train the model on poisoned data and observe performance changes.
- **Deliverable**:
  - Show how poisoning affects model behavior.
  - Discuss implications for cybersecurity.

3. Defenses Against Adversarial Attacks

3.1 Adversarial Training [2 Points]

- Retrain the model with adversarial examples to improve robustness.
- **Deliverable**:
  - Show how model performance improves against FGSM & PGD attacks.

3.2 Feature Squeezing [2 Points]

- Apply preprocessing techniques (e.g., image bit-depth reduction, median filtering) to remove adversarial noise.
- **Deliverable**:
  - Show effectiveness in mitigating evasion attacks.

3.3 Differential Privacy [2 Points]

- Implement **differentially private training** to defend against model inversion attacks.
- **Deliverable**:
  - Show how privacy-preserving techniques reduce data leakage.

References

[1] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," in *Proc. Int. Conf. Learn. Representations (ICLR)*, San Diego, CA, USA, May 2015. Available: https://arxiv.org/abs/1412.6572.

[2] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and T. Goldstein, "Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, QC, Canada, Dec. 2018. Available: https://arxiv.org/abs/1804.00792.

[3] M. Fredrikson, S. Jha, and T. Ristenpart, "Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures," in *Proc. ACM Conf. Comput. Commun. Security*

(CCS), Denver, CO, USA, Oct. 2015, pp. 1322-1333. Available: https://www.cs.cmu.edu/~mfredrik/papers/fjr2015ccs.pdf.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Lake Tahoe, NV, USA, Dec. 2012, pp. 1097-1105. Available: https://dl.acm.org/doi/10.5555/2999134.2999257.