CS 6958 / CS 4960 - Machine Learning Security Fall 2025

Homework 2

Due: **October 13, 2025, 11:59 PM**

**Descriptions & Instructions**:

The goal of this homework is to help you become familiar with the implementation of backdoor attacks and defenses covered in the lectures. You are provided with the Jupyter Notebook, **HW2.ipynb,** and you are required to complete the coding corresponding to the following questions. After completing the code segments, please ensure that the entire code in the notebook executes without any errors. The notebook walks you through a backdoor attack and also a backdoor detection method called Neural Cleanse. You will need to implement the attack and the defense from scratch and evaluate its effectiveness.

**Submit format:**

You will compress your Jupyter Notebooks source code into a **zip file** called HW2.zip and submit them on **Canvas**.

Then you need to submit a report including your solutions to the coding problems. You can choose to convert your Jupyter Notebook to a pdf report or take screenshots of your code and results. The report should be submitted on **Canvas**.

Please mark your solutions to each question correctly while submitting the report on Canvas.

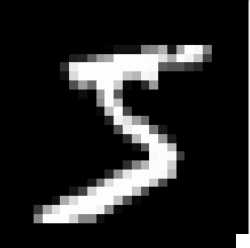
Failure to follow the instructions will lead to a deduction in points!

**Part 1: Targeted Backdoor Attack [50 pt]**

The goal of this question is to implement and launch the backdoor attack on a machine learning model, proposed by the paper - *BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain*.

To accomplish this assignment, you are encouraged to read and understand the paper and complete the following questions in HW2.ipynb:

1. Poison the dataset: [10 pt]
   * The trigger pattern is all-white 3\*3 square patch at the bottom-right corner of an image.
     1. An example of a poisoned image, notice the patch at the bottom-right corner.



* + Target is label 0.

1. Training: [20 pt]
   * Train the BadNets model using the poisoned data.
   * Evaluate the accuracy on poisoned and clean testing data.
2. Visualization: [20 pt]
   * Report the final attack success rate and clean accuracy of your model.
   * Randomly select one clean image from the original testing set, plot the image, print the original prediction of the BadNets model that you trained. Then add the trigger to the previous image you choose, print the prediction of the BadNets model that you trained. It is expected that after adding the trigger, the prediction of the model should be the target that you previously selected (i.e. label 0).

**Part 2: Backdoor Detection [50 pt]**

The goal of this question is to implement backdoor detection on the backdoored model constructed in **Part 1**, proposed by the paper - *Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks*.

To accomplish this assignment, you are encouraged to read and understand the paper and complete the following questions in HW2.ipynb:

1. Trigger generation: [30 pt]
   * Implement the function of adding a trigger to the input.
   * Define and calculate the cross-entropy loss.
   * Update the trigger parameters (mask and pattern) using an optimizer.
2. Anomaly detection [20 pt]
   * Load the model and data, and generate a trigger for each class.
   * Use Median Absolute Deviation to conduct anomaly detection on trigger sizes.