# CS7.405: Responsible and Safe AI Systems | Assignment 1

**Deadline: Feb 18th, 2024**

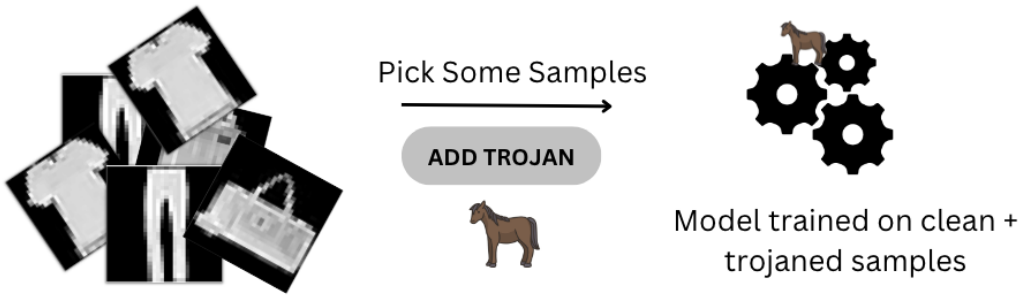
## Objective

This assignment will teach you about attacks on two stages of the machine learning pipeline. We will look at poisoning attacks during training, such as trojans/backdoors, and white-box evasion attacks during inference.

## General Instructions

1. You should implement the assignment in Python, **using only Pytorch library as the Neural Network Framework**.
2. We recommend using GPU for faster training/inference. (Eg. Google Colab's free GPU)
3. Ensure that the submitted assignment is your original work. Please do not copy any part from any source, including your friends, seniors, and/or the internet. If any such attempt is caught, serious actions, including an F grade in the course, are possible. Cite all the sources you refer to, when you answer theory questions.
4. A single `.zip` file needs to be uploaded to the Courses Portal.
5. Your grade will depend on the correctness of your answers and output. Due consideration will also be given to the clarity and details of your answers and the legibility and structure of your code.
6. Please start early to meet the deadline. Late submissions won't be evaluated, and there won't be any extensions.

# 1: Poisoning Attacks: Trojans (30m)

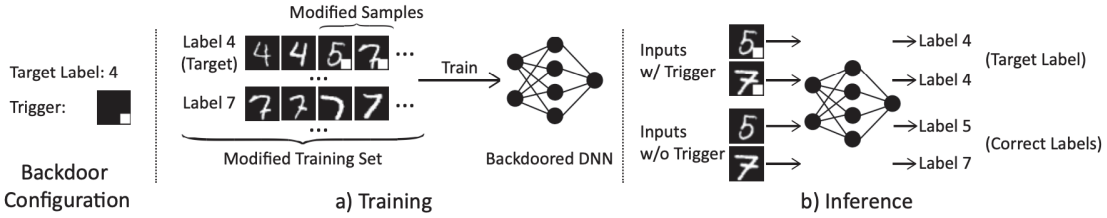



Part 1 (Poisoning Attacks)of this assignment is on attacks during the training stage.

# Introduction

This part of the assignment focuses on attacks at the training part of the ML pipeline. We will specifically look at backdoors/trojans (often used interchangeably).

Watch this video by Dan Hendrycks (developed as a part of ML Safety Course by CAIS) to get a headstart. You can look at this reading (from the same course) while you watch the video for more clarity and detail.




Source: Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks

You will be designing a trojan attack on a model trained on the fashion-MNIST dataset. This assignment uses the notebook from Center for AI Safety's ML Safety Course. *The coding assignment was created by Mantas Mazeika.*

# Tasks

## Filling in the notebook

1. **Duplicate the notebook on this Colab Link , and complete the code.** Most of the code is already written, you will simply have to implement some of the functions to do the following (More details in the notebook):

a. **Create the triggers to a random number of chosen images**: Here, you will create “stickers” that are essentially random 2D grids, and add these as “masks” on specific samples to poison them. (No marks)

b. **Visualise** the clean and poisoned data. (4 marks)

c. **Train** the network with the Trojans. (10 marks)

d. **Vary the number of samples** that are poisoned, and **plot** the accuracy of classification and attack success rate v/s number of samples poisoned. (8 marks)

2. Answer the following questions:

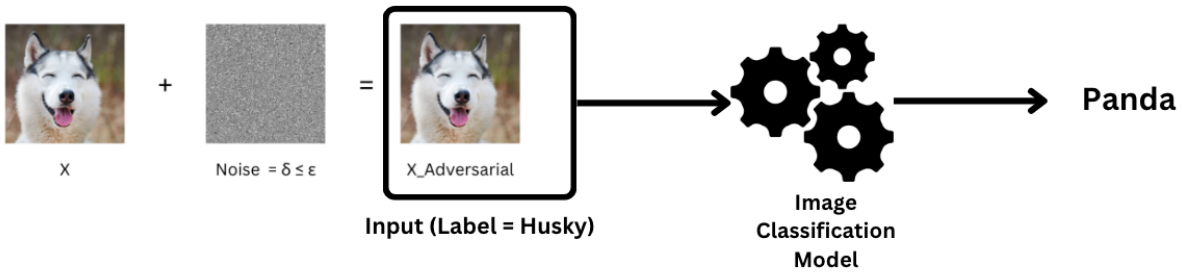
a. During training, analyze how the training accuracy on original data (tested against private validation data), and the attack success rate change. What is the percentage of poisoned samples required to achieve the task? What does this imply? (5 marks)

b. Give a real-world scenario of where trojan attacks could be used. What are the societal implications of such attacks? (3 marks)

c. **BONUS:** Read the NeuralCleanse paper to learn about what methods people use to detect and mitigate trojans. (No marks)

# 2: White-box Evasion attacks (40m)

In this part of the assignment, you will be training your own model for an image classification task and implementing white-box adversarial attacks to confuse the model. Please refer to resources 1, 2, and 3 for more information and code examples.



# Tasks

1. **Download** the MNIST dataset, and train a deep learning model of your own from scratch for this dataset. You must write a report on your classification accuracy for the train and test data. (5m)
2. **Pick an image** $x$ from the dataset. Create $x_{adv}$ using the methods shown listed below. **Visualize** and **compare** the images $x, x_{adv}$, and the adversarial noise $\delta = x_{adv} - x$ in your report (refer to the image above). Also, report the Losses $L(x, y_{original-label}); \theta)$ (where $\theta$ refers to the model parameters), $L(x_{adv}, y_{original-label}); \theta)$ and $L(x_{adv}, y_{changed-label}); \theta)$ along with the predicted label achieved after the attack. Assume the default attack budget $\epsilon$ to be 0.2, feel free to play around with more values of $\epsilon$. (15m)
    a. Fast Gradient Sign Method (FGSM)
        i. Untargeted
        ii. Targeted
    b. Projected Gradient Descent (PDG)
        i. Untargeted
        ii. Targeted
3. **Apply** all the above attacks on the test set using 3 or more **different budget-$\epsilon$ values** (any 3 values from 0.05 to 0.3). Report the test accuracies for these 3 cases in all the 4 scenarios. (10m)

| Attack Type | Clean Accuracy | $\epsilon_1$ | $\epsilon_2$ | $\epsilon_3$ |
|---|---|---|---|---|
| FGSM - Untargeted | | | | |
| FGSM - Targeted | | | | |
| PGD - Untargeted | | | | |
| PGD - Targeted | | | | |

4. Answer the following questions. support your answers with **empirical evidence and plots**. (10m)

a. Plot and analyze the accuracy change with respect to each of the epsilon values. How do the images change for each of these values? Will a higher epsilon value mean a good attack? If not, why?

b. Which attack is the best? List down the pros and cons of each attack. When would you use a targeted attack vs an untargeted attack?

5. **BONUS:** Perform Adversarial training on your model and report the new accuracies (20m)

# 3: Paper distillation (30m)

Read, and pick one paper from Lilian Weng's blogpost on adversarial attacks, and distill it. Distillation (as opposed to summarization) focuses on making someone who has not read the paper understand its most important contributions. Unlike summarization, you are encouraged to provide your own insights from the methodology and interpretations of the results. Feel free to copy important figures/statements from the paper. Your distillation should contain *atleast* the following:

- Motivation, i.e. what research gaps do they address - (5 Marks)
- Methodology - (10 marks)
- Main Conclusions and Rationale behind them - (5 marks)
- Critique the paper (unanswered questions, any limitations that the paper didn't address, and anything that you are skeptical about) - (10 marks)

Word Limit: ≥ 650 words, ≤ 1500 words

# Submission Format

Submit a zip that contains the following:

- Jupyter Notebook files (`.ipynb`) or Python code (`.py`) containing the code for all the coding tasks. Make sure that your outputs for (plots and values) are in your submission.
- A single PDF containing answers to the theory question(s), all the analyses from questions 1, and 2, and your paper distillation (clearly mention which

paper you've picked: title, and link).

# Resources

1. Adversarial attacks slides
2. https://adversarial-ml-tutorial.org/introduction/
3. FGSM Tutorial