Neural_avengers team members:

- Lyna Bouikni
  Arthur Mussard

1 Introduction, Context, and Aims

In the last decade, the robustness of machine learning classifiers against adversarial attacks has emerged as a pivotal area of research. This project embarks on a two-stage investigation to address the susceptibility of convolutional neural networks to perturbations intended to mislead classification models. The initial phase concentrates on establishing a baseline classifier and implementing attack mechanisms, namely Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD), to simulate adversarial conditions and evaluate the classifier's resilience. Subsequent to these attack implementations, the project adopts adversarial training, a defense mechanism that integrates adversarial examples during training to enhance the model's robustness. Advancing into the second stage, the project seeks innovation by considering new defense strategies against new adversarial tactics.

By experimenting with different attack and defense mechanisms, the project aims to dissect the intricate dynamics of adversarial machine learning, providing insights into both the vulnerabilities of classifiers and the efficacy of proposed defense strategies. The ultimate objective is to ensure that classifiers not only achieve high accuracy but also maintain integrity against adversarial manipulations.

2 Materials and Methods

2.1 Materials:

The CIFAR-10 dataset, consisting of 60,000 images across 10 classes, was used to train the convolutional neural networks (CNNs). The dataset was preprocessed with appropriate transformations (mainly normalization) to facilitate the training process.

2.2 Methods:

This section delineates the methodologies applied in this investigation. The models were trained locally on GPUs and the source code for all procedures is accessible at: https://github.com/Master-IASD/assignment3-2023-neural_avengers.

Our methodology unfolds in two principal stages:

1. **Establishing Baseline Classifier Robustness and Adversarial Attack Implementation:** The first stage involves training a basic classifier with a layered architecture comprising convolutional, max pooling, and fully connected layers using the CIFAR-10 dataset. We then implement two gradient-based adversarial attack mechanisms: the Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD). These attacks introduce perturbations to test and enhance the classifier's robustness. Additionally, adversarial training is employed, incorporating adversarial examples into the training process to bolster the model's defense against such attacks.

2. **Exploration and Enhancement of Machine Learning Defenses and Attack Mechanisms:** The second stage focuses on the innovation of defense strategies. Here, we explore the effectiveness of randomized networks as a countermeasure and deploy the DeepFool algorithm to craft advanced adversarial examples. This stage is dedicated to testing the resilience of the classifier against sophisticated adversarial techniques and refining defense mechanisms.

A more comprehensive presentation of the latter methods is presented in the following sections.

2.2.1 Baseline classifier:

We first Trained a basic CNN inspired by LeNet-5 (Model A) and then established a more robust classifier (Model B); a CNN with the following architecture: Conv+BN+MaxPool+Conv+BN+MaxPool +Conv+BN+MaxPool+Flatten+FC+FC. BN stands for Batch normalization and was applied after each convolutional layer.

2.2.2 Implementation of Attack Mechanisms

Fast Gradient Sign Method (FGSM): The FGSM attack was implemented as a function within our training pipeline to generate adversarial examples. This method perturbs original images by adding a small, carefully crafted noise, scaled by a factor ϵ (epsilon), in the direction of the gradient of the loss with respect to the image. The gradient is obtained by backpropagation from the initial model predictions. The resulting adversarial images are then clipped to ensure they remain within the valid image range. This process is intended to simulate potential adversarial attacks that the model may encounter and to test the robustness of the model against such perturbations.

Projected Gradient Descent (PGD): The PGD attack, an iterative method, was integrated to produce adversarial examples. The function iteratively adjusts the original images by a small amount α (alpha) in the direction of the loss gradient, followed by a projection step that ensures the perturbed image does not deviate from the original image by more than ϵ (epsilon) in any dimension. This process is repeated for a predefined number of iterations to gradually craft the adversarial examples within the specified bounds.

2.2.3 Adversarial Training:

Adversarial training was executed to enhance the model's defense mechanisms by incorporating adversarial examples during the training phase. This method involves modifying the standard training loop by injecting adversarial inputs, generated using either the FGSM or PGD method, directly into the training process. By setting a flag, the function selects the appropriate adversarial technique to apply and adjusts the inputs accordingly. Over a specified number of epochs, the model is exposed to adversarial examples, allowing it to learn from these perturbed inputs and adjust its parameters to reduce the likelihood of being misled by similar manipulations in the future. This exposure theoretically prepares the model to better withstand adversarial conditions encountered post-deployment.

2.2.4 Implementation of Enhanced Attack Mechanisms

DeepFool Attack: The DeepFool attack algorithm was integrated to craft adversarial examples that aim to cross the decision boundary with minimal perturbation. This iterative attack method subtly modifies the original image until the model's predicted class changes. The modification involves a computed perturbation based on the gradient of the loss with respect to the image, constrained by an overshoot parameter for controlled perturbation and a maximum number of iterations to ensure convergence. DeepFool is particularly effective for evaluating the model's vulnerability to minimalistic adversarial changes, simulating an attacker's attempt to cause misclassification with slight, often imperceptible, image adjustments.

2.2.5 Implementation of Enhanced Defense Mechanisms

Randomized Networks: As a defensive strategy, a randomized network architecture was employed, designed to introduce stochasticity into the model's inference process. This network, which extends the basic CNN with dropout layers, aims to mitigate the effectiveness of adversarial attacks by randomizing the presence of neurons during training. The application of dropout during inference acts as a form of model averaging, potentially increasing the robustness of the network by reducing the risk of overfitting to adversarial examples. The dropout layers are conditionally activated based on a flag that determines whether to apply this randomization, allowing the model to switch between deterministic and stochastic modes, thereby evaluating the impact of this defense strategy on adversarial resistance.

2.2.6 Enhanced Model Training and Evaluation

The model training and evaluation process consisted of two primary steps: model initialization and model testing under both standard and adversarial conditions.

Model Initialization: A randomized network architecture, pre-configured with dropout layers to introduce stochastic elements during the inference phase, was deployed on a GPU. The model's parameters were initialized with the state from a previously saved model to ensure consistency in subsequent testing.

Model Evaluation: To evaluate the model's performance, we conducted a series of tests using a predefined data loader, which provided a stream of test images and corresponding labels.

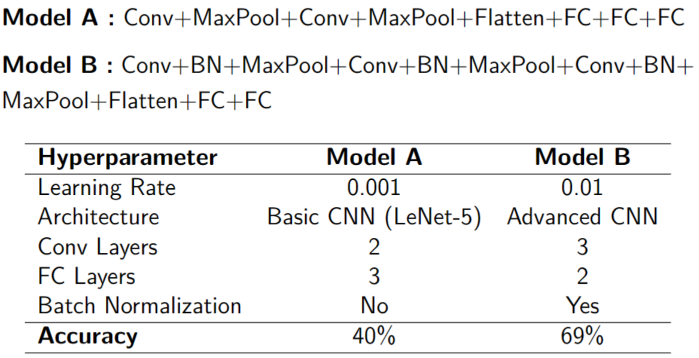
We conducted a comprehensive evaluation of the model using two sets of test data:

1. **Perturbed Test Data:** We applied the DeepFool attack to each test image, introducing subtle perturbations. The model's accuracy was assessed on this perturbed data to gauge its resilience to adversarial examples.
2. **Clean Test Data:** The model was also evaluated on a set of clean test images to establish its baseline performance in the absence of perturbations.

The outcomes of these tests were reported in terms of overall accuracy percentages for both perturbed and clean data sets.

3 Results and Discussion

3.1 Performance and Robustness of the Baseline Classifier:



Model A : Conv+MaxPool+Conv+MaxPool+Flatten+FC+FC+FC

Model B : Conv+BN+MaxPool+Conv+BN+MaxPool+Conv+BN+MaxPool+Flatten+FC+FC

Hyperparameter	Model A	Model B
Learning Rate	0.001	0.01
Architecture	Basic CNN (LeNet-5)	Advanced CNN
Conv Layers	2	3
FC Layers	3	2
Batch Normalization	No	Yes
Accuracy	40%	69%

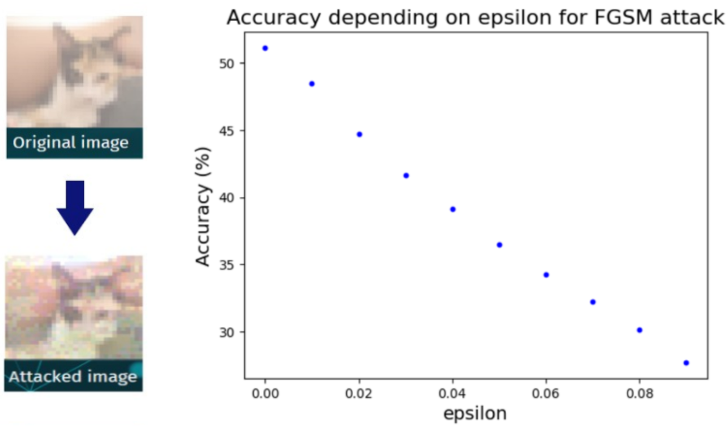
Table – Comparison of accuracies between two CNN models For Multi-class CIFAR data classification

Model A achieved a classification accuracy of 40% on the CIFAR-10 dataset. Model B, with its more sophisticated architecture, including an additional convolutional layer and batch normalization, reached a higher accuracy of 69%. These results underscore the significance of architectural enhancements and hyperparameter optimization in the construction of more accurate CNNs.

3.2 Efficacy of Adversarial Attack Mechanisms and Model Response:

3.2.1 Fast Gradient Sign Method (FGSM):

In this experiment, we examined the impact of adversarial perturbations generated by FGSM on the classification accuracy of a basic CNN model initially trained on the CIFAR-10 dataset. The model was subjected to FGSM attacks with varying intensities, characterized by the parameter epsilon (ϵ), that control the magnitude of the adversarial perturbations.

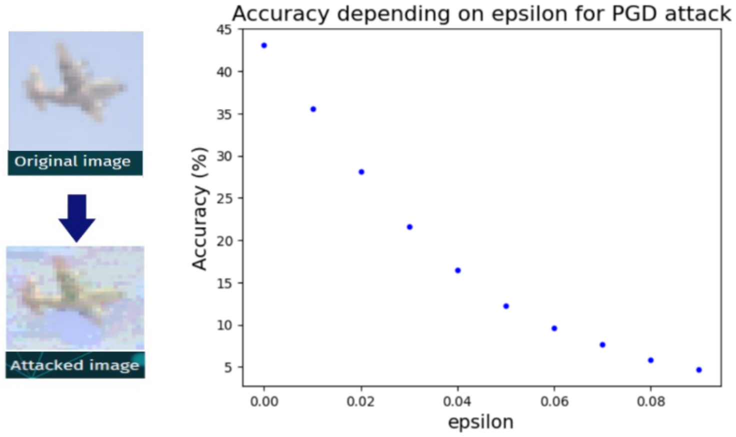


Impact of FGSM on an example input image: The analysis of the adversarial attack's effect on a representative image from the CIFAR-10 dataset—a cat—reveals that the perturbations result in visible color pixel alterations. Despite these modifications, the perturbed image remains recognizable to the human eye.

Impact of Epsilon on Accuracy: As illustrated in the accompanying plot, there is an inverse relationship between ϵ and the model's accuracy. Starting from an accuracy of nearly 52% with no perturbation ($\epsilon = 0$), the model experiences a steep decline in performance, dropping to below 35% accuracy as ϵ approaches 0.1. This trend indicates that even minimal perturbations can degrade the model's performance.While humans can still classify the perturbed image with ease, the CNN's performance deteriorates, suggesting that the model's learned feature representations are easily disrupted by the adversarial noise.

3.2.2 Projected Gradient Descent (PGD):

The resilience of the same CNN model was further evaluated under the PGD attack, with variable epsilon (ϵ) values.



Impact of PGD on a sample image: An example depicting an airplane demonstrates the visual effects of the PGD attack. The adversarial perturbations induce a pronounced distortion, which is evident in the attacked image.

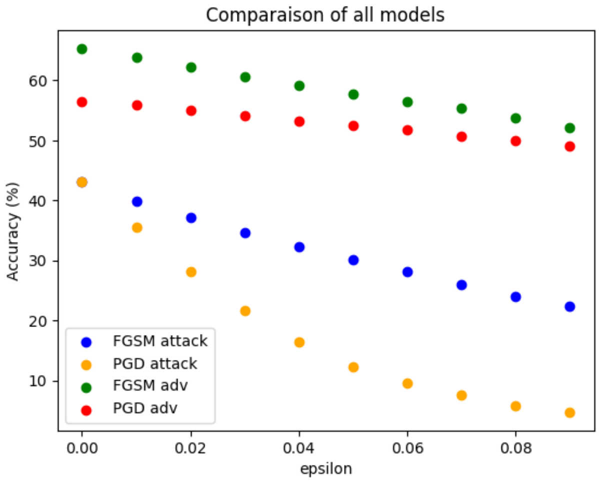
Impact of Epsilon on Accuracy: The scatter plot reflects a significant decline in the model's accuracy as ϵ increases.The accuracy decreases from approximately 40% at $\epsilon = 0$ to around 10% for $\epsilon = 0.1$, emphasizing the PGD attack's potency. This steep decrease suggests that the PGD attack effectively exploits the model's vulnerabilities, leading to a substantial performance drop even at lower values of ϵ.

3.2.3 Adversarial Training Efficacy:

The results indicate that the iterative application of small perturbations by PGD can lead to a high degree of model misclassification. This aligns with the current understanding that iterative attacks like PGD can be more destructive to model accuracy than one-step attacks like FGSM due to their ability to adaptively adjust the adversarial noise in each iteration.

3.2.4 Enhanced Attack Mechanism (DeepFool):

In the subsequent section, we will explore randomized networks, to evaluate its effectiveness in mitigating the impact of adversarial attacks by Deepfool.



The scatter plot demonstrates that adversarially trained models (FGSM adv and PGD adv) consistently maintain an accuracy exceeding 50% across all tested ϵ values, significantly outperforming non-defended models in similar attack scenarios. This suggests that adversarial training enhances model robustness, enabling effective learning from perturbed inputs and stability in accuracy, even when challenged with FGSM and PGD attacks. This contrast underscores the effectiveness of adversarial training in bolstering model defenses against adversarial perturbations. Notably, the FGSM adversarially trained model (FGSM adv) consistently outperformed the PGD adversarially trained model.

As demonstrated in the small or even imperceptible perturbations introduced to the images, the Deepfool attack introduces subtle pixel-level changes to the input image. While these changes may be imperceptible to human observers, they can significantly affect the model's classification accuracy.

3.2.5 Enhanced deffense Mechanism (Randomized networks):

In this section, we assess the performance of the randomized network after applying the DeepFool attack to perturb the test images. The evaluation yielded the following results:

Condition	**Accuracy**
Training randomized networks on unperturbed data	11.57%
Training randomized networks on perturbed data	25.07%

Table 1: Comparison of model accuracy on adversarially perturbed test images, before and after training with randomized networks

The results indicate an improvement in model accuracy when training randomized networks on perturbed data compared to training on unperturbed data. This suggests that incorporating adversarial training, specifically using data perturbed by the DeepFool attack, enhances the network's defense mechanism against adversarial attacks. Further analysis is required to understand the full implications of these findings in the context of adversarial machine learning.

4 Conclusion:

In conclusion, this study has explored the dynamic interplay between adversarial attacks and defensive strategies in machine learning. Our experiments with convolutional neural networks (CNNs) on the CIFAR-10 dataset reveal that while adversarial attacks like FGSM and PGD can significantly compromise model accuracy, incorporating adversarial training substantially enhances model robustness. This improved resilience is evident in adversarially trained models consistently outperforming their non-adversarially trained counterparts under equivalent attack conditions. Our findings underscore the critical importance of integrating advanced defensive mechanisms in the training of machine learning models to safeguard against evolving adversarial tactics.