DAT945-Assignment 1

September 17, 2025

1 Assignment 1: Adversarial Examples and Uncertainty in AI Models

1.1 DAT945: Secure and Robust AI Model Development

1.1.1 Task 1

Consider an image classification model $f(x)$ that takes an input image x and predicts its class label. An adversarial attack aims to generate a modified image x' that is visually similar to the original image x but is misclassified by the model.

1. Define the L_p-norm and the L_p-ball in n-dimensional space. Explain how they are utilized in adversarial attacks.
2. Discuss the importance of the parameter ϵ in adversarial attacks. How does adjusting its value influence the effectiveness of the attack?
3. Contrast targeted and untargeted adversarial attacks. Provide examples of each and describe their respective objectives.
4. Describe the Fast Gradient Sign Method (FGSM) and its process for creating adversarial examples.
5. Examine the correlation between the size of perturbation and the success rate of an adversarial attack. How does this interplay inform the concept of adversarial robustness?
6. Apart from the FGSM, list other prevalent methods for crafting adversarial examples. Briefly discuss one of these alternatives.
7. Identify potential real-world applications for adversarial attacks. Suggest defensive measures that could be adopted to mitigate these attacks.

1.1.2 Task 2

Imagine we are using a natural language processing (NLP) model to classify text inputs. Consider the following scenario:

We have a dataset consisting of text inputs and their corresponding labels. We want to evaluate the robustness of our NLP model against adversarial text examples created using TextAttack.

1. Explain what TextAttack is and how it applies to NLP models.
2. Generate two examples using TextAttack on an NLP model: one with a targeted attack and another with an untargeted attack. Describe the modifications made to the original texts.
3. Discuss the potential impact of adversarial text examples on NLP model performance and decision-making processes.

4. Propose strategies for defending NLP models against adversarial text examples. Consider both model hardening and input sanitization approaches.

1.1.3 Task 3

Explore the concepts of homomorphic encryption and its application in secure computing:

1. Define homomorphic encryption and distinguish it from conventional encryption methods.
2. Identify and explain the two primary forms of homomorphic encryption, noting their differences.
3. How does homomorphic encryption allow computations on encrypted data without disclosing the plaintext? Provide a detailed example.
4. Discuss the security attributes of homomorphic encryption, including confidentiality, integrity, and authenticity.
5. How can homomorphic encryption enhance privacy and security in machine learning applications? Focus on aspects such as predictive modeling and neural network training.
6. Outline the limitations or challenges associated with homomorphic encryption and propose potential solutions.
7. Describe the mechanism of fully homomorphic encryption (FHE) and how it differs from other schemes.
8. Present a practical application of homomorphic encryption in areas like secure cloud computing or privacy-preserving database queries.
9. What are current research challenges in homomorphic encryption, and how might they be addressed?

1.1.4 Task 4

Consider a deep neural network f_θ for predictive analytics:

1. Define and differentiate between aleatoric and epistemic uncertainty in the context of machine learning.
2. Suppose the output y of the network is modeled as a Gaussian distribution $p(y|x, \theta)$. Explain how this model represents aleatoric uncertainty.
3. Explore methods for quantifying epistemic uncertainty in deep learning, such as Bayesian approaches. Detail how these methods assess the uncertainty in model parameters.
4. Develop a loss function that integrates both aleatoric and epistemic uncertainties, and describe its optimization using stochastic gradient descent.
5. Discuss the challenges in implementing uncertainty quantification methods in deep learning and propose solutions to mitigate these issues, such as reducing computational demands.

[ ]:
```
```