Adversarial Machine Learning I

IMPORTANT NOTES:
Study lecture materials at least 1 hour and prepare the questions prior to the tutorial session. The questions will be discussed in the tutorial session.

1. Define the term ”adversarial example” in the context of machine learning. How does it differ from a benign input with random noise?

   An ”adversarial sample” refers to an input intentionally modified to deceive a model, causing misclassification while appearing unchanged to humans. Noise is random but adversarial perturbations are optimized to exploit model weaknesses.

2. Explain why negative images are considered ”semantic adversarial examples” for CNNs. What does this reveal about how CNNs learn?

   CNNs fail on negative images because they rely on pixel statistics rather than semantic understanding. Humans recognize them easily due to preserved structure.

3. Describe the Fast Gradient Sign Method (FGSM). Why does it use the sign of the gradient rather than the gradient itself?

   The Fast Gradient Sign Method (FGSM) is a white-box attack that generates adversarial examples by perturbing inputs in the direction that maximizes the model’s loss. It uses sign(∇J) to ensure perturbations are small (controlled by ϵ) and maximally aligned with the loss increase. Raw gradients would require per-dimension tuning as their magnitudes vary across dimensions, the sign ensures all perturbations are equal in magnitude.

4. How does Zeroth-Order Optimization (ZOO) approximate gradients in black-box settings? Why is it less efficient than gradient-based attacks?

   Zeroth-Order Optimization (ZOO) is a black-box attack method that generates adversarial examples without accessing the target model’s gradients. Instead, it estimates gradients by querying the model’s outputs (e.g., class probabilities or confidence scores).

   - Perturb the input x by making two tiny changes for each pixel i:
     - $x + \delta e_i$ (add small noise, δ to the i-th dimension)
     - $x - \delta e_i$ (subtract noise)
   - Send both perturbed inputs to the black-box model and record the outputs.
   - Estimate the gradient for dimension i using the symmetric difference:

   $$\hat{\nabla}_i f(x) \approx \frac{f(x + \delta e_i - f(x - \delta e_i)}{2\delta} \tag{1}$$

   - Repeat for all dimensions (pixels), then perturb x along the estimated gradient:

   $$x' = x - \epsilon \cdot \hat{\nabla} f(x) \tag{2}$$

   ZOO is less efficient because:

   - It requires 2 queries per dimension.
   - The gradient estimates rely on finite differences, which are sensitive to output differences. Small output differences are prone to numerical errors, large output differences result in poor approximation. It is also vulnerable to output noise.
   - It requires many iterations to craft a successful adversarial example.

5. An attacker uses FGV to generate adversarial images for data augmentation. How does this differ from FGSM, and why might it improve model robustness?

   FGV uses raw gradient values (not signs), creating smoother perturbations. This diversifies training data, improving robustness to varied attacks

6. A copyright detection system fails to identify a modified song. The attacker used ZOO with 500 API queries. What trade-offs did they face?

   Copyright systems (e.g., YouTube's Content ID) use AI to fingerprint audio/video. An attacker modifies a song to evade detection while keeping it perceptually identical. The goal of ZOO is to find tiny perturbations, e.g., pitch shifts and echoes, to break that fingerprint. This is a trade-off between stealth and query cost. A ZOO with 500 API queries generate perturbations that are inaudible to humans, but it requires many queries to find subtle changes that fool the model.