Homework Assignment 1

Due: **2:00pm Thursday, 23 November 2023 on CISPA CMS**

The purpose of this assignment is to get you familiar with the definition of adversarial examples and the typical targeted and untargeted methods for generating adversarial examples introduced in Lecture 3.

Collaboration Policy: You should do this assignment by yourself and submit your own answers. You may discuss the problems with anyone you want and it is also fine to get help from anyone on problems with LaTeX or Jupyter/Python. You should note in the *Collaborators* box below the people you collaborated with.

Collaborators: TODO: replace this with your collaborators (if you did not have any, replace this with *None*)

This problem set includes both PDF and Jupyter notebook components. You should complete the answers to the PDF part by writing your answers in `hw1.tex`, and submitting your generated PDF file in CISPA CMS under Submission tab. The first thing you should do in hw1.tex is setting up your name as the author of the submission by replacing the line, `\submitter{TODO: your name}`, with your name and your Matr. ID, e.g., `\submitter{Susan Blake (7583916)}`. Before submitting your PDF, also remember to (1) list your collaborators by replacing the TODO in `\collaborators{TODO: replace ...}`, and (2) replace the second line in hw1.tex, `\usepackage{macro}` with `\usepackage[response]{macro}` so the directions do not appear in your final PDF.

Problem 1 (10 pts) Consider a binary logistic regression model with loss $\ell(\boldsymbol{w}; \boldsymbol{x}, y) = -\log \sigma(y \cdot \langle \boldsymbol{w}, \boldsymbol{x} \rangle)$, where $\boldsymbol{x} \in \mathbb{R}^d$, $y \in \{-1, +1\}$, and $\sigma(z) = 1/(1+\exp(-z))$. Let $\mathcal{B}_\epsilon(\boldsymbol{x}, \ell_\infty) = \{\boldsymbol{x}' \in \mathbb{R}^d : \|\boldsymbol{x}' - \boldsymbol{x}\|_\infty \leq \epsilon\}$ limit the searching region of feasible $\boldsymbol{x}'$. Show that

$$\max_{\boldsymbol{x}' \in \mathcal{B}_\epsilon(\boldsymbol{x}, \ell_\infty)} \ell(\boldsymbol{w}; \boldsymbol{x}', y) = -\log \sigma(y \cdot \langle \boldsymbol{w}, \boldsymbol{x} \rangle - \epsilon \|\boldsymbol{w}\|_1).$$

Note: This implies that for linear models, robust learning against ℓ_∞ perturbations is essentially looking for a weight parameter with small ℓ_1-norm and maximized margin.

Problem 2 (10 pts) Suppose we want to solve the maximization problem of logistic regression specified in Problem 1 using gradient descent. Compute the gradient of $\ell(\boldsymbol{w}; \boldsymbol{x}, y)$ with respect to $\boldsymbol{x}$.

Note: Based on the computed gradient, the PGD attack can be understood as simply performing $\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \alpha \cdot \frac{\partial}{\partial \boldsymbol{x}} \ell(\boldsymbol{w}; \boldsymbol{x}, y)\Big|_{\boldsymbol{x}=\boldsymbol{x}_t}$, where $\boldsymbol{x}_0 = \boldsymbol{x}$, and projecting $\boldsymbol{x}_{t+1}$ onto the nearest point within the perturbation ball $\mathcal{B}_\epsilon(\boldsymbol{x}, \ell_\infty)$ in the form of ℓ_∞-norm distance iteratively if $\|\boldsymbol{x}_{t+1} - \boldsymbol{x}\|_\infty > \epsilon$.

Implementation Problems. Below are two problems that you need to complete the provided Jupyter notebook. The goal is to help you understand the iterative PGD attack (both untargeted and targeted versions). For illustration, we will use a pretrained ImageNet ResNet50 model as the victim, and use a ladybug image from ImageNet as the seed example. Note that the class index of ladybug is 301.

If you haven't used Jupyter notebook before, you can start by installing Jupyter on your computer using this link: https://jupyter.org/install. To run the provided `hw1.ipynb` file, note that you also need to install the required packages properly. The `imagenet_class_index.json` file provides the 1000 ImageNet labels with corresponding class names and class indices.

Problem 3 (10 pts) Let K be the number of class labels. Consider ℓ_∞ perturbations with $\epsilon = 2/255$. The goal of PGD attack is to solve the following objective using an iterative algorithm:

$$\max_{\boldsymbol{x}'} \ \ell(h_\theta(\boldsymbol{x}', y)) \ \text{ subject to } \ \|\boldsymbol{x}' - \boldsymbol{x}\|_\infty \leq \epsilon,$$

where $(\boldsymbol{x}, y)$ is the input example (in this task, a ladybug image), $h_\theta : \mathbb{R}^d \to \mathbb{R}^K$ is neural network mapping from input to logit layer (in this task, the pretrained ResNet50 model), and ℓ is the cross-entropy loss.

Your Task: Write down the algorithm pseudocode of untargeted PGD attack, then implement the iterative attack by completing the corresponding section of the provided Jupyter notebook. Specifically, you need to initialize the PGD attack in the implementation with zero initialization, and run PGD attack using a SGD optimizer with a learning rate 0.1 for 30 iterations. In this case, you are using the raw gradient without the sign function, as described in the Note of Problem 2. Remember that you also need to ensure the output of your algorithm lies within $\mathcal{B}_\epsilon(\boldsymbol{x}, \ell_\infty)$.

Note: After you implement the attack, you may want to check what the predicted label of the generated adversarial examples $\boldsymbol{x}'$ is, and think about whether it makes sense.

Problem 4 (10 pts) Note that the previous implementation of PGD attack is untargeted, which does not specify a targeted label to guide the adversarial examples generation process. Under the same setting of Problem 3, the targeted version of PGD attack is designed to solve the following objective:

$$\max_{\boldsymbol{x}'} \Big(\ell(h_\theta(\boldsymbol{x}', y)) - \ell(h_\theta(\boldsymbol{x}', y_{\text{targ}})) \Big) \text{ subject to } \|\boldsymbol{x}' - \boldsymbol{x}\|_\infty \leq \epsilon,$$

where y_{targ} is a pre-selected targeted label that is different from y.

Your Task: Write down the algorithm pseudocode of targeted PGD attack, then implement the attack by completing the corresponding section of the provided Jupyter notebook. Specifically, the target label should be set as zebra (the corresponding class index is 340 in ImageNet). You need to initialize the PGD attack in the implementation with zero initialization, and run PGD attack using a SGD optimizer with a learning rate 0.005 for 100 iterations.

Note: After you implement the attack, you can also replace the targeted label with other class index (i.e., any number from 1 to 1000 other than 301 and 340), and see if your attack can also succeed in generating a corresponding targeted adversarial example. Will there be a difference in the generation process of adversarial examples for different classes?

Problem 5 (bonus, 5 pts) Consider a linear model with soft-SVM loss $\ell(\boldsymbol{w}; \boldsymbol{x}, y) = \max(0, 1 - y \cdot \langle \boldsymbol{w}, \boldsymbol{x} \rangle)$, where $\boldsymbol{x} \in \mathbb{R}^d$ and $y \in \{-1, +1\}$. For any $p \geq 1$, show that

$$\max_{\boldsymbol{x}' \in \mathcal{B}_\epsilon(\boldsymbol{x}, \ell_p)} \ell(\boldsymbol{w}; \boldsymbol{x}', y) = \max(0, 1 - y \cdot \langle \boldsymbol{w}, \boldsymbol{x} \rangle + \epsilon \|\boldsymbol{w}\|_q),$$

where $\mathcal{B}_\epsilon(\boldsymbol{x}, \ell_p) = \{\boldsymbol{x}' \in \mathbb{R}^d : \|\boldsymbol{x}' - \boldsymbol{x}\|_p \leq \epsilon\}$ is the ϵ-ball at $\boldsymbol{x}$ in ℓ_p-norm, and q satisfies $1/p + 1/q = 1$.

Note: This result generalizes what we have shown in Problem 1, which implies that for linear models, robust learning against general some specific ℓ_p perturbations $(p \geq 1)$ is essentially looking for a weight parameter with small ℓ_q-norm that maximizes the margin.

End of Homework Assignment 1 (PDF part)
Don't forget to also complete and submit the Jupyter notebook!