Adversarial robustness in AI

Aditya Prasad

January 30, 2023

§1 Motivation

We'll be talking about adversarial robustness today; let's start by providing some context and motivation. One important date is 2012, when the neural network AlexNet made a big splash with its performance on the ImageNet competition. In the time since — as we all know — neural networks have only captured more attention with their performance across an impressive variety of tasks.

At the time that neural networks were rising up, one natural intuition that you might have had is that any model that generalizes well on a task — like neural nets for image recognition – should (at a minimum!) do well on features that are "close" to the training sets. We might call this property **local generalization**. This intuition turns out to fail pretty miserably when it comes to neural networks, though!

So that's one important point of this research: despite generalizing *extremely* well, neural networks really don't have this local generalization property. (In fairness, other learners also suffer from this kind of behavior, such as KNN, but you might have hoped that neural networks wouldn't, given the caliber of their performance on vision tasks.)

§2 "Intriguing Properties of Neural Networks"

Let's start talking about **feature extraction**. Here's some intuition: one way you might try to understand the human brain is to hook up a sensor to a person's brain, have them perform some actions, and see which regions of the brain 'light up' for each action. You would expect that similar actions light up similar regions, and this would help you understand what different regions of the brain are "doing."

The idea with feature extraction for neural networks is similar: we fix a layer i and consider the activation values $\phi_i(x) \in \mathbb{R}^k$ of the k nodes in this layer on input image x. One thing we can then look at is

$$x' \in \arg\max_{x \in \mathcal{I}} \langle \phi_i(x), e_j \rangle$$

for a standard basic vector $e_j \in R^k$. This defines the set of images x' that most stimulate the jth node in layer i, allowing us to determine what structure or patterns this node is "extracting."

Now, when we do this with the standard basis vectors e_j, we indeed find that the images x' in the arg max share some visual commonalities. The surprising finding from this paper is that we also see that behavior when we replace the standard basis vectors e_j with *random* vectors $v \in \mathbb{R}^k$!

Another finding from this paper — really the reason we're talking about it — is that these neural networks for vision tasks are also pretty vulnerable to **adversarial**

examples. These are features that are very nearby to points in the training set but are nevertheless misclassified by the corresponding neural network. For instance, we saw an example where a picture of school bus was imperceptibly altered and then classified as an ostrich! We won't go too deep into the math here (and the underpinnings of this behavior aren't really well understood anyway) but the algorithm for finding these perturbations is pretty straightforward. Supposing your neural network induces the classifier $f : \mathbb{R}^m \to [k]$, simply solve (an approximation of): minimize $\|r\|_2$ subject to $f(x+r) \neq f(x)$ and $x + r \in [0,1]^m$. (We're normalizing pixel values to the interval $[0,1]$ here.)

Overall, two key take-aways:

- Feature extraction on these ImageNet neural networks doesn't do too much, and
- Neural networks have *adversarial examples* – features that are *very* close to the training set but that cause a high-performing neural network to fail!

§3 "Universal Adversarial Perturbations"

In the previous paper, adversarial perturbations were tailored for each particular example. In this paper, we'll see that these perturbations can actually be constructed *uniformly* across all images. That is, given a trained net inducing a classifier f, one can construct a *single* perturbation r such that $f(x) \neq f(x+r)$ for a large fraction of images x (about 80%).

Tangent 3.1

We went on a bit of a tangent here; these are some of the topics we touched on.

- Why do **dominant labels** exist, i.e., why does the network tend to favor certain labels when it is fooled? Could it be that these labels are close to the "average" label, in some sense?
- One important note: most directions of perturbation are *not* adversarial. That is, these kinds of networks are actually quite robust to random noise. Just randomly picking an adversarial vector won't work!
- The previous point is a bit at odds with one of the 'corollaries' of the **manifold hypothesis**, which states that high-dimensional real-world datasets usually lie on very low-dimensional manifolds of feature space.

§4 "Motivating the Rules of the Game for Adversarial Example Research"

This is a position paper with some interesting ideas on how research should be performed, rather than lots of results per se. Here are some of the points it raises:

- We shouldn't necessarily restrict ourselves to "imperceptible" (to humans) feature changes. In something like moderation of pictures/images on social media, we might want to be robust against images that are deliberately crafted to fool the automated filters but are still easily read by humans. (E.g., text for a financial scam placed on an image of a video game.)

  - Sometimes, the adversarial input is created 'from scratch' by the attacker, and it doesn't even make sense to ask whether it is indistinguishable from benign input. Which benign input?

- Adversarial robustness is a game between an attacker and a classifier, and there are actually many possible rules to this game.
  - What kind of behavior does the attacker want? A *targeted* attack, that leads to the classifier to predict a certain incorrect label, or an *untargeted* attack, that just seeks to cause any misclassification.
  - What kind of access does the attacker have to the classifier? *Black-box* access or *white-box* access?
  - How many times is the game played? Who moves first?

Another interesting component of the paper is an example showing that ℓ_p norms for images really don't coincide with human intuition! In particular, there can be a trio of images A, B, C such that $d_p(A, B) = d_p(A, C)$ but B is indistinguishable from A (to a human) and C looks *very* different from A!

Tangent 4.1

Another fun tangent we went on:

- How do these experiments interact with recent (albeit limited) theoretical results showing that neural networks maximize margins?
- Did these papers discuss any kind of *defense* to adversarial perturbations?
  - These papers in particular don't discuss defenses, but there's certainly tons of work on that. It can be a bit of a cat-and-mouse game between research on adversarial attacks and research on defense.
- Can universal perturbations be found in the black-box case?
  - Yes! One idea is to create your own model that mimics the black-box model very well, create universal perturbations on your model, and then these perturbations tend to do well on the original black-box. So there's often not as much of a difference between the white-box and black-box settings as you might expect.