What causes adversarial examples? An overview of theoretical explanations

Edgar Dobriban

September 30, 2020

Contents

1 Background

- Neural networks are vulnerable to adversarial examples. An adversarial example is one that is designed to ”fool” the machine learning system, leading to mistakes. Brought to wide attention in (Szegedy et al., 2013; Biggio et al., 2013)

  Many earlier related works exist, for instance Dalvi et al. (2004); Lowd and Meek (2005); Globerson and Roweis (2006). These works consired various formulations and settings such as reverse engineering classifiers, and robust classifiers to feature deletion.

- There are several models for adversarial attacks: white-box attacks assume the attackers know the model, while black-box attackers do not.

- How can we learn models robust to adversarial inputs?

- Let x be the training example, $f(x) := f(x; W)$ be the model, and L be the loss. We can define the adversarial risk minimization problem, see (Madry et al., 2017). Robust optimization references: Soyster (1973); Ben-Tal et al. (2009); Xu et al. (2009). Robust decision-making, Wald-s min-max model: Wald (1939, 1945).

  $$\min_W R(W) := \mathbb{E}_{(x,y)\sim D} \max_{\delta \in S} L(W, x + \delta, y)$$

  Here D is the data distribution. Here now L has parameters the weights W, the feature vector x, and the label y. Also S is a set of allowed perturbations.

  The saddle point problem as the composition of an inner maximization problem and an outer minimization problem.

- Adversarial training

  1. For fixed W, perform PGD on randomly chosen datapoints x, starting from randomly chosen perturbation. PGD for $\max_{\delta \in \Delta} L(W, x + \delta, y)$ is:

     $$\delta^{t+1} = \prod_{\Delta} [\delta^t + \varepsilon \nabla_\delta L(W, x + \delta^t, y)]$$

     Get adversarial perturbations $\delta^*(x)$

  2. Perform one step of GD on W.

     $$W \leftarrow W - \alpha/|B| \sum_{x \in B} \nabla_W L(W, x + \delta^*(x), y)$$

2 Theories of adversarial examples

Here we collect the various theories proposed to explain adversarial learning

2.1 High dimensions

2.1.1 Folklore

- "in high-dimensional settings, there are always adversarial examples" (formalized perhaps in Gilmer et al. (2018)?)

If we find work that formalizes one of them (or if we do that work) let's move them out of here!

Theory	Data	Archit & Train	Experimental test	Notes
High dimensionality	truly high-dim, ”uniform”, deterministic	nonzero pop err	shouldn’t work in low-dim?	real data low-dim?
Oscillation		”dense” level sets; implied by interpo+Bayes>0	how close is random img to dog?	interpo+Bayes >0 not enough in high dim;
Non-Robust features	exist non-rob features $\mathbb{E}yf(x) > 0$, $\mathbb{E}\inf_{\delta} yf(x+\delta) = 0$	use non-rob features		somewhat circular
Low-dim manifold	Unknown low-dim manifold		where are the adv ex in gen-ve model?	untestable in practice?
Condition number	Large cond num of weight mx		is adv ex aligned with top sing vec? appears with large sv?	

Table 1: Summary of theories to explain adversarial examples. For each approach, we list the assumptions on the data, the architecture and training. We also list possible experiments to test it, and other notes/limitations.

2.1.2 Gilmer et al, The relationship between high-dimensional geometry and adversarial examples

Gilmer et al. (2018) describes three existing explanations for the existence of adversarial examples:

> One common hypothesis is that neural network classifiers are too linear in various regions of the input space, (Goodfellow et al., 2014; Luo et al., 2015). Another hypothesis is that adversarial examples are off the data manifold (Goodfellow et al., 2016; Song et al., 2017; Samangouei et al., 2018; Lee et al., 2017). (Lee et al., 2017) argue that large singular values of internal weight matrices may cause the classifier to be vulnerable to small perturbations of the input.

As an alternative explanation, they consider “the high-dimensional geometry of data manifolds combined with the presence of low but non-zero error rates” and study data (x, y) generated as follows (Gilmer et al., 2018, Section 2):

$$y \sim \text{Bern}(1/2), \qquad\qquad x|y \sim \begin{cases} 1 \cdot \text{Unif}(\mathbb{S}^{d-1}) & \text{if } y = 0 \\ R \cdot \text{Unif}(\mathbb{S}^{d-1}) & \text{if } y = 1 \end{cases},$$

i.e., the label y is 0 or 1 with equal probability, and the sample x is uniform on a sphere of radius 1 if $y = 0$ and is uniform on a sphere of radius R if $y = 1$.

Experimentally, (Gilmer et al., 2018, Section 3) learns a classifier for this data in $d = 500$ dimensions and finds that its decision boundary looks correct when restricted to many directions but is quite off when restricted to adversarially chosen ones. It even yields errors on the data manifold. However, the same does not seem to happen when $d = 2$.

(Gilmer et al., 2018, Section 4) considers a tailored "quadratic network" and illustrate that this also has issues.

These examples motivate their main theorem.

Proposition 2.1 (Theorem 5.1 from Gilmer et al. (2018))**.** *Let $E \subset \mathbb{S}^{d-1}$ be a set of misclassified (inner sphere) points for a given model. Then*

$$\mathbb{E}_x R(x, E) \leq O\left[\frac{\Phi^{-1}\{1 - \Pr_x(x \in E)\}}{\sqrt{d}}\right],$$

where $R(x, E) = \inf_{z \in E} \|x - z\|$ is the distance between x and E, and Φ^{-1} is the inverse normal CDF function.

Informally, as long as the misclassified points have nonzero probability, i.e., the (test) error probability $\Pr_x(x \in E) > 0$, the average distance to the error set goes to zero at a rate of $1/\sqrt{d}$. The key idea seems to be that $\mathbb{E}_x R(x, E)$ is maximized by E being a cap of $\mathbb{S}^{d-1}$, i.e.,

$$E = \left\{x \in \mathbb{S}^{d-1} : x_1 > \frac{\alpha}{\sqrt{d}}\right\},$$

for some $\alpha > 0$. For this, the authors cite Figiel et al. (1977) though that paper appears to point to others that may be earlier. Gilmer et al. (2018) then use a Gaussian approximation of x to estimate $\Pr_x(x \in E)$ and $\mathbb{E}_x R(x, E)$ in terms of α.

2.1.3 Restatement

We summarize/describe the high-dimensional geometry perspective of (Gilmer et al., 2018, Theorem 5.1) with the following statements.

Proposition 2.2 (ℓ_2 adversarial examples cover in high dimensions)**.** *Let $f, g : \sqrt{d}\mathbb{S}^{d-1} \to \{0, 1\}$ and $\epsilon > 0$ be arbitrary. If $x \sim \text{Unif}(\sqrt{d}\mathbb{S}_{d-1})$ then*

$$\Pr\left\{\exists x' \in \sqrt{d}\mathbb{S}_{d-1} : \|x - x'\|_2 \leq \epsilon\sqrt{d} \text{ and } f(x') \neq g(x')\right\} \geq 1 - 4\exp(-c\epsilon^2)/\mu, \qquad (1)$$

where $\mu \coloneqq \Pr\{f(x) \neq g(x)\}$ and c is a universal constant.

Proof of Proposition 2.2. The proposition follows from the spherical isoperimetric inequality (Figiel et al., 1977, Theorem 2.1) and a "blow-up" lemma (Lemma 2.3) by noting that $\mu = \Pr(A)$ where $A \coloneqq \{x \in \mathbb{S}^{d-1} : f(x) \neq g(x)\}$ and that $A_{\epsilon\sqrt{d}}$ is the right hand side of (1). □

Lemma 2.3 (Variant of (Vershynin, 2018, Exercise 5.1.9))**.** *Let A be a spherical cap of $\sqrt{d}\mathbb{S}^{d-1}$ and $\epsilon > 0$ be arbitrary. Then*

$$\Pr(A_\epsilon) \geq 1 - 4\exp(-c\epsilon^2/2)/\Pr(A),$$

where $A_\epsilon \coloneqq \{x \in \sqrt{d}\mathbb{S}^{d-1} : \exists x' \in A \text{ with } \|x - x'\|_2 \leq \epsilon\}$ is an ℓ_2 neighborhood of A and c is the absolute constant from (Vershynin, 2018, Lemma 5.1.7).

Proof of Lemma 2.3. If $\Pr(A) \geq 1/2$, then (Vershynin, 2018, Lemma 5.1.7) immediately gives

$$\Pr(A_\epsilon) \geq 1 - 2\exp(-c\epsilon^2) \geq 1 - 4\exp(-c\epsilon^2/2)/\Pr(A).$$

Furthermore if $\epsilon \leq s := \sqrt{(1/c)\ln\{2/\Pr(A)\}}$ then we trivially have

$$\Pr(A_\epsilon) \geq 0 = 1 - 2\exp(-cs^2)/\Pr(A) \geq 1 - 4\exp(-c\epsilon^2/2)/\Pr(A).$$

Hence the main work is extending the result to the case where $\Pr(A) < 1/2$ with $\epsilon > s$. For this, we take the general strategy suggested by (Vershynin, 2018, Exercise 5.1.9).

The key idea is that $\Pr(A_s) > 1/2$ and $A_\epsilon \supset (A_s)_{\epsilon - s}$ so (Vershynin, 2018, Lemma 5.1.7) applies again. We first show $\Pr(A_s) > 1/2$. Without loss of generality, let $A = \{x \in \sqrt{d}\mathbb{S}^{d-1} : x_1 > t/\sqrt{2}\}$ where $t > 0$ since $\Pr(A) < 1/2$. Then sub-gaussianity of x_1 yields $\Pr(A) \leq 2\exp(-ct^2)$ and hence $s = \sqrt{(1/c)\ln\{2/\Pr(A)\}} \geq t$. Thus $A_s \supset \{x \in \sqrt{d}\mathbb{S}^{d-1} : x_1 > 0\}$ so $\Pr(A_s) > 1/2$. The triangle inequality yields $A_\epsilon \supset (A_s)_{\epsilon-s}$ since any point $\epsilon - s$ away from a point that is s away from A is at most $\epsilon - s + s = \epsilon$ away from A. Hence $\Pr(A_\epsilon) \geq \Pr\{(A_s)_{\epsilon-s}\}$. The proof concludes by applying (Vershynin, 2018, Lemma 5.1.7) to A_s to obtain

$$\Pr\{(A_s)_{\epsilon-s}\} \geq 1 - 2\exp\{-c(\epsilon - s)^2\} \geq 1 - 2\exp\{-c(\epsilon^2/2 - s^2)\} = 1 - 4\exp(-c\epsilon^2/2)/\Pr(A),$$

where the second inequality uses $(a+b)^2 \leq 2(a^2 + b^2)$ with $a = \epsilon - s$ and $b = s$. □

As a side remark, note that in the proof one could have instead calculated

$$\Pr(A_{2\epsilon}) \geq \Pr\{(A_s)_{2\epsilon - s}\} \geq 1 - 2\exp\{-c(2\epsilon - s)^2\} \geq 1 - 2\exp\{-c\epsilon^2\},$$

which seems to be the intended result of (Vershynin, 2018, Exercise 5.1.9). Another related result appearing in lecture notes (Kelner, 2009, Theorem 4) gives the bound

$$\Pr(A_\epsilon) \geq 1 - 2\exp(-\epsilon^2/16)/\Pr(A),$$

which is quite similar.

A version of Proposition 2.2 that simply replaces ℓ_2 with ℓ_∞ holds immediately since

$$\begin{aligned}&\Pr\Big\{\exists x' \in \mathbb{S}_{d-1} : \|x - x'\|_\infty \leq \epsilon \text{ and } f(x') \neq g(x')\Big\}\\ &\qquad \geq \Pr\Big\{\exists x' \in \mathbb{S}_{d-1} : \|x - x'\|_2 \leq \epsilon \text{ and } f(x') \neq g(x')\Big\},\end{aligned}$$

by virtue of the fact that the ℓ_∞ unit ball circumscribes the ℓ_2 unit ball. However, things change if we instead inscribe the ℓ_2 ball.

Conjecture 1 (Proposition 2.2 does not hold for ℓ_∞ at a different scaling). *Let $f(x) = \operatorname{sign}(x_1)$, $g(x) = \operatorname{sign}(x_1 - \alpha)$, and $\epsilon > 0$ be arbitrary. Then*

$$\Pr\Big\{\exists x' \in \sqrt{d}\mathbb{S}_{d-1} : \|x - x'\|_\infty \leq \epsilon \textit{ and } f(x') \neq g(x')\Big\} < C_{\mu,\epsilon}, \qquad (2)$$

where $\mu := \Pr\{f(x) \neq g(x)\}$ with $x \sim \operatorname{Unif}(\sqrt{d}\mathbb{S}_{d-1})$, and $C_{\mu,\epsilon}$ is some constant independent of d.

Proof of Conjecture 1. Without loss of generality, assume $\alpha > 0$ and define

$$\begin{aligned} A &\coloneqq \{x \in \sqrt{d}\mathbb{S}^{d-1} : f(x) \neq g(x)\} = \{x \in \sqrt{d}\mathbb{S}^{d-1} : 0 < x_1 < \alpha\}, \\ A_\epsilon &\coloneqq \{x \in \sqrt{d}\mathbb{S}^{d-1} : \exists x' \in A \text{ with } \|x - x'\|_\infty \leq \epsilon\} \subset \{x \in \sqrt{d}\mathbb{S}^{d-1} : -\epsilon < x_1 < \alpha + \epsilon\}. \end{aligned}$$

Observe that $\mu = \Pr(A)$ and the left hand side of (2) is $\Pr(A_\epsilon)$. Then we can lower bound

$$\Pr(A) = \Pr\{x \in \sqrt{d}\mathbb{S}^{d-1} : 0 < x_1 < \alpha\} = 1 - \Pr\{x \in \sqrt{d}\mathbb{S}^{d-1} : x_1 > \alpha\} \geq 1 - \exp(-\alpha^2/2),$$

but we need to upper bound $\Pr(A_\epsilon)$. One bound we could try to use is (Ball, 1997, Lemma 2.3) but it does not seem to be sharp enough to get the desired result. □

2.1.4 Other thoughts

1. The following suggests that the high-dimensional geometry may not be the root cause.

   Sometimes, a much smaller class of perturbations is enough. E.g., rotating and translating an image can lead to adversarial examples. Data augmentations helps a bit, but does not fully solve it. (Fawzi Frossard 2015, Engstrom Tran Tsipras Schmidt Madry 2018)

   See also "Beyond pixel norm-balls: parametric adversaries using an analytically differentiable renderer", ICLR 2019. "Compute adversarial examples by perturbing physical parameters instead of pixel colors. Present (1) adversarial geometry by 3D shape perturbations, and (2) adversarial lighting by scene lighting perturbations."

2. Mahloujifar et al. (2019) Empirically measuring concentration: Fundamental limits on intrinsic robustnesss

   (a) Impossibility results, such as Gilmer et al. (2018), should not make the community hopeless in finding more robust classifiers.

   (b) Concentration of measure is not the sole reason behind the vulnerability of existing classifiers to adversarial examples.

   (c) CIFAR-10 ep = 8/255

   Madry et al. (2017), 12.70%. 52.96%

   Our Bound, 14.22%, 29.21%

   " The AdvRisk reported for our method can be seen as an estimated lower bound of adversarial risk for existing classifiers."

   Feature scattering can get 68.6% rob err. STN (Stability training with noise, 65.5%) [ED: so this is solved?]

2.2 Oscillations

Highly oscillatory fuctions may cause instability and adversarial examples.

2.2.1 Belkin et al, Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate., Theorem 5.1

Belkin et al. (2018) contains a section relating their analysis to the “ubiquity of adversarial examples in interpolated learning” with the following main theorem.

Proposition 2.4 (Theorem 5.1 from Belkin et al. (2018))**.** *Consider data drawn from a compact domain $\Omega \subset \mathbb{R}^d$ and a consistent interpolating classifier $\{\hat{f}_n\}_{n=1}^{\infty}$. If there is non-zero label noise everywhere, i.e.,*

$$\forall x \in \Omega \quad \Pr(f^*(x) \neq Y | X = x) > 0,$$

then

$$\forall \epsilon > 0, \delta \in (0,1) \quad \exists N \in \mathbb{N} \quad s.t. \quad \forall n \geq N$$
$$\Pr\left[\forall x \in \Omega \;\; \exists a \in \Omega : \|x - a\| \leq 2\epsilon \; and \; \hat{f}_n\{(X_1, Y_1), \ldots, (X_n, Y_n)\}(a) \neq f^*(a)\right] \geq \delta$$

Informally, so long as there are sufficiently many training samples with noisy labels, any consistent interpolating classifier will (with high probability) have adversarial examples everywhere.

Let $A_n = \{x \in \Omega : \hat{f}_n(x) \neq f^*(x)\}$ be the set of points at which $\hat{f}_n$ disagrees with the Bayes optimal classifier. Consistency means that, with probability one, $\lim_{n\to\infty} \mu(A_n) = 0$, where μ is the distribution of x.

Could a slightly stronger version be proved that roughly says that any consistent interpolating classifier will (with high probability) have adversarial examples everywhere that there is nonzero label noise?

2.2.2 Flipside: excessive invariance

1. The flipside of excessive variability is excessive invariance. Jacobsen et al. (2018) argue that deep networks are not only too sensitive to task-irrelevant changes of their input (the oscillatory behavior), but are also too invariant to a wide range of task-relevant changes, thus making vast regions in input space vulnerable to adversarial attacks.

2. Jacobsen et al. (2018) constuct invertible/bijective networks $x \to z := F(x)$ such that $z = (z_s, z_n)$, and class labels are based on logits of z_s, while z_n is nuisance.

   They construct metameric examples $x_m = F^{-1}(z_s, \tilde{z}_n)$, by taking signal and nuisance from two different examples.

3. They observe that the nuisance dominates. Visually the examples look like the nuisance class.

   ”We show such excessive invariance occurs across various tasks and architecture types. On MNIST and ImageNet one can manipulate the class-specific content of almost any image without changing the hidden activations.”

   Perturbation robust models are more vulnerable to invariance based adversarial examples.

4. They suggest a fix, to use the Independence Cross-Entropy. Instead of just $\max I(y; z_s)$, maximize $I(y; z_s|z_n)$.

5. Note: these examples seem much harder to come up with than other adversarial examples, because they require a very special architecture (invertible, disentangled).

2.2.3 Other thoughts

1. Initial works like Szegedy et al. (2013) also argued that instability should be the main cause of adversarial examples.

2. Suppose the decision boundary varies a lot, is a complicated manifold. Then, much of the space is close to it. I guess this is the appropriate version of the oscillatory behavior for classification.

   This is the principle behind DeepFool (Moosavi-Dezfooli et al., 2016) [iterative linearization, and stp to boundary]. Also behind universal adversarial perturbations (can use same perturbation to move towards boundary for all examples, iterate & aggregate deepfool over full dataset).

   ”We further explained the existence of such perturbations with the correlation between different regions of the decision boundary.” [they do a PA on matrix of normals to decision boundary/min perturbations]

2.3 Low dimensional manifold

Observe that in many cases, e.g., images, data are believed to lie in a lower-dimensional (often nonlinear) space. We investigate this low intrinsic dimensionality as a source of adversarial examples.

2.3.1 Prior occurrences

- Biggio et al. (2013) study the effect of data manifold on attacks. They argue that to make attacks more successful, they should be on the manifold, so as to make attacks look more similar to real objects. Off the manifold, we can't control the behavior of the classifier, and it may be more random.

2.4 Non-robust features

2.4.1 Ilyas et al, Adversarial examples are not bugs, they are features

1. Ilyas et al. (2019)

2. Somewhat circular: need robust features for robust learning. Combatting the non-robust features can be achieved by ”standard training on robust features.”

3. How construct robust features?

   Start with adversarially trained neural network, extract features at end layer, and modify them slightly. Specifically, suppose we have a robust (adversarially trained) model C. They to construct a distribution $(x, y) \sim D_r$ such that the features f used by C have the same marginal value, i.e., $f(x)y$ has the same mean under the original and new distribution D_r. Moreover, they want the that the features f' not used by C have zero marginal value. (So, they artificially restrict the useful features to be exactly the ones used by C.)

   The construct this by manipulating the individual training data points. For input x, and robust representation $g(x)$ from C, they construct a warped data point x_r that minimizes

$$\|g(x_r) - g(x)\|_2.$$

This roughly leads to similar "marginal values" of g for the original and warped distributions. Then, to ensure that these are all useful features, they have a heuristic sampling approach (sample the starting point of GD from the training data).

4. Then they train a standard classifier on D_r, and show that both its usual and robust accuracy is ok.

   But: if we do the same starting with x, then robust accuracy is low.

   > Overall, our findings corroborate the hypothesis that adversarial examples arise from (non-robust) features of the data itself. By filtering out non-robust features from the dataset (e.g. by restricting the set of available features to those used by a robust model), one can train a robust model using standard training.

5. So my impression is that the robust features are hard to pin down on the real data.

6. Also, their explanations are extremely heuristic. They only theoretically study linear loss (as opposed to logistic, exponential, etc). Also they do not take into account interactions between the features.

2.4.2 Model

1. **Basic Definitions and Notation.** For any classifier $\hat{y} : \mathbb{R}^p \to \mathcal{C}$ the robust risk (with respect to the 0-1 loss) and a norm $\|\cdot\|$ is:

   $$R_{\text{rob}}(\hat{y}, \varepsilon, \|\cdot\|) := \mathbb{E}_{x,y} \sup_{\|\delta\| \le \varepsilon} I\{\hat{y}(x+\delta) \neq y\} = \mathbb{E}_y \Pr_{x|y} \left\{ \exists_{\delta : \|\delta\| \le \varepsilon} \;\; \hat{y}(x+\delta) \neq y \right\}, \tag{3}$$

   where $x \in \mathbb{R}^p$ are the features, $y \in \mathcal{C}$ is the label, $\mathcal{C}$ denotes the set of classes, I is the indicator function, and $\varepsilon \ge 0$ is the perturbation radius.

2. Consider the standard binary classification setting where data is distributed via a mixture of two Gaussians with classes $\mathcal{C} = \{\pm 1\}$:

   $$x|y \sim \mathcal{N}(y\mu, \sigma^2 I_p), \qquad\qquad y = \begin{cases} +1 & \text{with probability } \pi, \\ -1 & \text{with probability } 1-\pi, \end{cases} \tag{4}$$

   where $\mu \in \mathbb{R}^p$ specifies the class means ($+\mu$ and $-\mu$), $\sigma^2 \in \mathbb{R}_{>0}$ is the within-class variance, and $\pi \in [0,1]$ is the proportion of the $y = 1$ class. Note that the means are centered at the origin without loss of generality (wlog). By scaling, we will also take $\sigma^2 = 1$ wlog to simplify the presentation.

   The Bayes optimal classifier for this problem is the linear classifier $\hat{y}^*_{\text{Bay}}(x) = \text{sign}(x^\top \mu - q/2)$ where $q := \ln\{(1-\pi)/\pi\}$ and we define $\ln(0) := -\infty$. Note that scaling the argument of sign by any positive constant does not change the prediction. Denoting the normal cumulative distribution function $\Phi(x) := (2\pi)^{-1/2} \int_{-\infty}^{x} \exp(-t^2/2) dt$ and $\bar{\Phi} := 1 - \Phi$, the corresponding Bayes risk

   $$R_{\text{Bay}}(\mu, \pi) := R_{\text{std}}(\hat{y}^*_{\text{Bay}}) = \pi \cdot \Phi\left(\frac{q}{2\|\mu\|_2} - \|\mu\|_2\right) + (1-\pi) \cdot \bar{\Phi}\left(\frac{q}{2\|\mu\|_2} + \|\mu\|_2\right), \tag{5}$$

   is the smallest attainable standard risk and characterizes problem difficulty.

3. What are the robust features?

2.5 Others

1. A New Defense Against Adversarial Images: Turning a Weakness into a Strength (NeurIPS 2019)

   propose to detect adv ex by two criteria that should be true for natural images: ”easily perturbable by first order methods, but not by random methods”

   circular, but seems to help?

   kind of like a ”hypothesis test”. Under the null hypothesis for natural images, we have the above properties. The alternative is adversarial examples.

   they atttack it with ”Best effort white-box adversary. Based on our proposed detection method, we define a white-box adversary that aims to cause misclassification while passing the detection criteria C1 and C2.”

2. ”Defense Against Adversarial Attacks Using Feature Scattering-based Adversarial Training” - enough to be have good accuracy against perturbed features

3. Li et al. (2019) ”One observes that STN performs slightly worse than TRADES when the size of attacks is small, and becomes better when the size increases. Intuitively, the added random noise dominantly reduces the accuracy for small attack size and becomes beneficial against stronger attacks. It is worth-noting that Algorithm 1 adds almost no computational burden, as it only requires multiple forward passes, and stability training only requires augmenting randomly perturbed examples. On the other hand, TRADES is extremely time-consuming, due to the iterative construction of adversarial examples.”

4. Adversarial training for free

   key premise - same adversarial perturbation δ works for all examples. Learn it along with parameters.

5. .[STN $\sim$ Trades $\sim$ adv train $\sim$ adv train for free, where learn one perturbation for all examples]

6. ME-Net: Towards Effective Adversarial Robustness with Matrix Estimation

   ”first pixels are randomly dropped from the image; then, the image is reconstructed using ME. We show that this process destroys the adversarial structure of the noise”

7. Computational constraints (Bubeck)

8. ”Unlabeled Data Improves Adversarial Robustness”

   Barriers to robustness. Schmidt et al. [41] show a sample complexity barrier to robustness in a stylized setting. We observed that in this model, unlabeled data is as useful for robustness as labeled data. This observation led us to experiment with robust semisupervised learning. Recent work also suggests other barriers to robustness: Montasser et al. [31] show settings where improper learning and surrogate losses are crucial in addition to more samples; Bubeck et al. [5] and Degwekar and Vaikuntanathan [12] show possible computational barriers; Gilmer et al. [16] show a high-dimensional model where robustness is a consequence of any non-zero standard error, while Raghunathan et al. [38] , Tsipras et al. [47] , Fawzi et al. [15] show settings where robust and standard errors are at odds. Studying ways to overcome these additional theoretical barriers may translate to more progress in practice.

   [ED: read Schmidt et al. (2018). Also Fawzi et al. (2018)]

9. ”Jacobsen et al. (2018) provide an alternative viewpoint and argue that the adversarial vulnerability is a consequence of narrow learning, resulting in classifiers that rely only on a few highly predictive features in their decisions.”

References

T. W. Anderson. An Introduction to Multivariate Statistical Analysis. Wiley New York, 2003.

K. M. Ball. An elementary introduction to modern convex geometry. In S. Levy, editor, Flavors of Geometry, volume 31 of Mathematical Sciences Research Institute Publications, pages 1–58. Cambridge University Press, Cambridge, 1997. URL `http://library.msri.org/books/Book31/files/ball.pdf`. Book info at `http://library.msri.org/books/Book31/contents.html`.

M. Belkin, D. J. Hsu, and P. Mitra. Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate. In Advances in Neural Information Processing Systems 31, pages 2300–2311. Curran Associates, Inc., 2018.

A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. Robust optimization, volume 28. Princeton University Press, 2009.

B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli. Evasion attacks against machine learning at test time. In Joint European conference on machine learning and knowledge discovery in databases, pages 387–402. Springer, 2013.

N. Dalvi, P. Domingos, S. Sanghai, and D. Verma. Adversarial classification. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 99–108, 2004.

A. Fawzi, O. Fawzi, and P. Frossard. Analysis of classifiers’ robustness to adversarial perturbations. Machine Learning, 107(3):481–508, 2018.

T. Figiel, J. Lindenstrauss, and V. D. Milman. The dimension of almost spherical sections of convex bodies. Acta Mathematica, 139(0):53–94, 1977. doi: 10.1007/bf02392234.

J. Gilmer, L. Metz, F. Faghri, S. S. Schoenholz, M. Raghu, M. Wattenberg, and I. Goodfellow. The relationship between high-dimensional geometry and adversarial examples, 2018. URL `http://arxiv.org/abs/1801.02774v3`.

A. Globerson and S. Roweis. Nightmare at test time: robust learning by feature deletion. In Proceedings of the 23rd international conference on Machine learning, pages 353–360, 2006.

I. Goodfellow, Y. Bengio, and A. Courville. Deep learning. MIT press, 2016.

I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry. Adversarial examples are not bugs, they are features. In Advances in Neural Information Processing Systems, pages 125–136, 2019.

J.-H. Jacobsen, J. Behrmann, R. Zemel, and M. Bethge. Excessive invariance causes adversarial vulnerability. arXiv preprint arXiv:1811.00401, 2018.

J. Kelner. Lecture 16. URL `https://ocw.mit.edu/courses/mathematics/18-409-topics-in-theoretical-computer-science-an-algorithmists-toolkit-fall-2009/lecture-notes/MIT18_409F09_scribe16.pdf`. From MIT OCW 18.409 An Algorithmist’s Toolkit, 2009.

H. Lee, S. Han, and J. Lee. Generative adversarial trainer: Defense to adversarial perturbations with gan. arXiv preprint arXiv:1705.03387, 2017.

B. Li, C. Chen, W. Wang, and L. Carin. Certified adversarial robustness with additive noise. In Advances in Neural Information Processing Systems, pages 9459–9469, 2019.

D. Lowd and C. Meek. Adversarial learning. In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pages 641–647, 2005.

Y. Luo, X. Boix, G. Roig, T. Poggio, and Q. Zhao. Foveation-based mechanisms alleviate adversarial examples. arXiv preprint arXiv:1511.06292, 2015.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083, 2017.

S. Mahloujifar, X. Zhang, M. Mahmoody, and D. Evans. Empirically measuring concentration: Fundamental limits on intrinsic robustness. In Advances in Neural Information Processing Systems, pages 5210–5221, 2019.

S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2574–2582, 2016.

P. Samangouei, M. Kabkab, and R. Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. arXiv preprint arXiv:1805.06605, 2018.

L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, and A. Madry. Adversarially robust generalization requires more data. In Advances in Neural Information Processing Systems, pages 5014–5026, 2018.

Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. arXiv preprint arXiv:1710.10766, 2017.

A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. Operations research, 21(5):1154–1157, 1973.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199, 2013.

R. Vershynin. High-Dimensional Probability. Cambridge University Press, September 2018. doi: 10.1017/9781108231596.

A. Wald. Contributions to the theory of statistical estimation and testing hypotheses. The Annals of Mathematical Statistics, 10(4):299–326, 1939.

A. Wald. Statistical decision functions which minimize the maximum risk. Annals of Mathematics, pages 265–280, 1945.

H. Xu, C. Caramanis, and S. Mannor. Robustness and regularization of support vector machines. Journal of machine learning research, 10(Jul):1485–1510, 2009.