Computational Hardness in Robust Learning

Jihye Choi[1]
[1] University of Wisconsin Madison
jihye@cs.wisc.edu

Abstract

Over recent years, learning classifiers that are robust to adversarial perturbations has emerged as a crucial, yet challenging problem. There are different approaches to reduce the classification error over adversarial examples, but their success is still limited. To understand this phenomenon, a line of works have provided evidence that adversarial vulnerability is inherently due to the computational limitations of the tasks or learning algorithms. In this survey, inspired by the work of Bubeck, Lee, Price and Razenshteyn, we investigate examples of classification tasks where (1) no robust classifier is possible, (2) only inefficient robust classifier is possible, (3) learning robust classifiers requires large sample complexity, or (4) an efficient robust classifier exists, but no efficient learning algorithm is possible. We hope that this survey will encourage the research on developing these ideas to the problems in modern machine learning that involves neural networks, or on studying new learning algorithms for computationally efficient robust classifications.

1 Introduction

Starting with [10] that introduced the term, *adversarial examples*, there is a large body of work that successfully demonstrated that machine learning models are susceptible to small imperceptible perturbations. This failure becomes a severe threat when the trained model is integrated to larger security-critical systems, or used to decide important problems with social and legal consequences. Motivated by this, devising learning algorithms that are robust to adversarial examples has emerged as an important, but challenging problem.

Since [10], huge volume of papers have studied on pursuing adversarial robustness, but we are stuck at a moderate success. For instance, a provable robust accuracy on MNIST dataset is around 96%. To understand why, recent studies started to view adversarial perturbations from the perspective of computational learning theory. That is, the adversarial vulnerabilities are due to computational hardness, and they are unavoidable byproduct of computational limitations of tasks, related to computational complexity or sample complexity.

More specifically, Fawzi et al. [7] derive fundamental upper bounds on the robustness to adversarial perturbations of *any* classification function. [2, 4] provide Boolean classification tasks where learning robust classifier is computationally non-intractable. [9] shows simple classification tasks where the sample complexity required for robust learning is higher than the sample complexity for classical learning by a polynomial factor. [5, 8] demonstrate the feasibility of robust learning, considering both computational and sample complexity.

In relation to analyzing the moderate success of robust learning from the view of computational limitations, Bubeck et al. [2, 4] (henceforth referred to as BLPR) first postulate five possible hypotheses of robust learning:

- **World 1**. Computationally efficient *non-robust* classification is possible, but no robust classification is possible, regardless of computational or sample-efficiency considerations.

Preprint. Work in progress.

- **World 2**. No efficiently computable robust classifier exists. Namely, there exists a robust classifier f, which is intractable to compute (in the sense that the mapping $x \mapsto f(x)$ is hard to compute).
- **World 3**. Computationally efficient robust classification is possible, but requires large sample complexity.
- **World 4**. Computationally efficient robust classifier exists and can be learned with small sample-complexity. However, the learning algorithm is not computationally efficient.
- **World 5**. There exist computationally efficient robust classifier. The learning algorithm is also computationally efficient, and the required sample-complexity is also small.

In this paper, we adopt this taxonomy of worlds we may live in, and organise the referred papers using this framework in Section 3. This line of work suggests that efficient robust classifier exists or may not exist depending on the regime of concept classes, and thereby invokes the need to clearly establish what we want out of robust learning given bounded computational efficiency and sample complexity. We aim to understand these papers in depth and figure out the statistical and computational trade-offs in learning robust classifiers. And by doing so, we clarify where we are at on the way of pursuing robustness in machine learning.

2 Notations

Before to introduce related works in the literature, we begin by formalizing the notion of robust classification and introduce notation that we use throughout this paper.

For simplicity, we consider only binary classification tasks and an input space $\mathcal{X} = \{0,1\}^n$. Given a labeled set of $poly(n)$ i.i.d samples drawn from two distributions $D_0, D_1 \subseteq \{0,1\}^n$, the goal of standard binary classification is to find a classifier $h : \{0,1\}^n \to \{0,1\}$ such that

$$\text{for } b \in \{0,1\}, \quad \Pr_{x \leftarrow D_b}[x \in h^{-1}(b)] > 0.99$$

When we state that there is a *ϵ-robust classifier*, it means the existence of a classifier h such that

$$\Pr_{x \leftarrow D_b}[B(x,\epsilon) \in h^{-1}(b)] > 0.99$$

where $B(x,\epsilon)$ is the Hamming ball of center x and radius $\epsilon \geq 0$.

3 Hardness in Robust Learning

In this section, we demonstrate the works that provide evidence in favor of or against each of the five hypotheses by BLPR.

3.1 World 1: No robust classifier exists

Fawzi et al. [7] provides evidence that supports the hypothesis of world 1. They consider a Lipschitz generative model of the form $\mathcal{X} = g(Z)$ where g is a Lipschitz continuous generative model that maps latent vectors $z \in Z$ to the space of images $\mathcal{X}$ and $z \sim \mathcal{N}(0, I_d)$. Let the Lipschitz parameter of g be L. Assume that we have a classifier on X with roughly balanced classes, i.e., $\mathbb{P}(C_0) \approx \frac{1}{2}$ where C_0 is the set of samples that belong to class 0 (note that it is not restricted to binary classification in the original work [7]). Then, allowing adversarial perturbations of size $O(1)$ makes roughly 90% of the samples susceptible to the perturbations (while the samples are of size $O(\sqrt{d})$. That is, almost everything lies within the constant distance from the boundary of the classifier in Gaussian space, and every point outside the set C_0 can be moved by a constant amount to get in C_0, and through the Lipschitz map g, this is also true in the image domain $\mathcal{X}$.

However, this argument is based on a very strong Lipschitz constraint of g and on minimal perturbations of a Gaussian measure, which might be not helpful in practical settings. It is usually implausible that one can generate a natural image by applying a Lipschitz map to a Gaussian random variable. In modern machine learning, g is a neural network, where L is related to the product of spectral norms of matrices doing linear operations. This implies that L is the order of the number of parameters, which should be prohibitively large to fit real distributions on images. For a more detailed discussion, see [4].

On the other hand, [6] provides evidence against hypothesis 1 on natural learning tasks. Humans seem to be robust classifiers with a small error rate. Humans sometimes bring up optical illusions as an example of adversarial examples indeed, but this small portion of failure cases can be tolerated. Unfortunately, we are still not at clear consensus since it is hard to realize the exponential power of trying all possible perturbations during the test.

3.2 World 2: Only computationally inefficient robust classifiers exist

In the work of Degwekar et al. [5], a theorem is presented, which is in favor of World 2.

Theorem 1. *Assuming one-way functions, there is a classification task over $\{0,1\}^n$ where (1) efficient non-robust classifiers exist, (2) inefficient $O(n)$-robust classifier exists, but (3) no efficient 1-robust classifier exists.*

To prove the theorem 1, they first describe a learning task based on the existence of average-case hard functions. Let g be a random function $\{0,1\}^n \to \{0,1\}$, which is an example of average-case hard functions, and any poly-time algorithm can compute $z \mapsto g(z)$ better than random guessing only by negligible amount. Let (Encode, Decode) be a good error correcting code where Encode algorithm returns a redundant encoding of the input message that the Decode can efficiently recovers the original message allowing a constant fraction of errors. Then, consider two distributions constructed as follows: for $x \leftarrow \{0,1\}^n$ uniformly,

$$D_0 = (0, \text{Encode}(x, g(x))) \quad \text{and} \quad D_1 = (1, \text{Encode}(x, 1 - g(x))) \tag{1}$$

It is obvious that non-robustly distinguishing samples $\sim D_0$ from D_1 is easy because we only need to look at the first coordinate, thus proving (1) of Theorem 1. It is impossible to learn a robust classifier against an adversary that corrupts the first coordinate. In this case, the problem comes down to computing $g(x)$, which cannot be done efficiently, hence this proves (3). The existence of inefficient robust classifier is supported by the fact that one can use Decode algorithm and compute $g(x)$.

This proof is based on no cryptographic assumptions, but with g being a average-case hard function, the samples from D_0, D_1 cannot be generated efficiently. Instead, by relying on a minimal cryptographic assumption, one-way functions, the process of generating the samples becomes simpler by sampling $(z, g(z))$ for random z's, instead of computing $g(z)$ given z.

3.3 World 3: Efficient robust classifier is only learnable with large sample complexity

Schmidt et al. [9] shows positive results on the hypothesis of World 3. They demonstrates that already in a simple natural model, the sample complexity of robust learning can be larger than that of standard learning. As in their work, consider a data model, which is a mixture of two spherical Gaussians with one component per class. Let the two separate Gaussians be $\mu_0 = \mathcal{N}(\theta, \sigma^2 I_d)$ and $\mu_1 = \mathcal{N}(-\theta, \sigma^2 I_d)$ with prior $\theta \sim \mathcal{N}(0, I_d)$. Draw a label $y \in \{0,1\}$ uniformly at random, and sample the data from μ_0 if y is 0, or from μ_1 otherwise.

We have the total variation distance between μ_0, μ_1 as,

$$\begin{aligned} TV(\mu_0, \mu_1) &\le \sqrt{Entropy(\mu_0, \mu_1)} \\ &\le \sqrt{\sum_i^d \frac{\theta_i^2}{\sigma^2}} \;=\; \sqrt{\frac{d}{\sigma^2}} \end{aligned} \tag{2}$$

where as (2) goes to zero, the two Gaussian become indistinguishable. From this equation, we can observe that we can control σ^2 up to d, and even with $\sigma^2 \approx d$, so the amount of overlap between μ_0 and μ_1 is large, we can still distinguish them with error at moat 1%.

Our goal in this task can be states as estimating $theta$ given labeled samples, and this can be done by separating empirical averages from μ_0, μ_1. That is, the only reasonable classifier is max correlation with empirical means: a linear classifier $w = \theta + \frac{\sigma}{\sqrt{n}Z},\ Z \sim \mathcal{N}(0, I_d)$. Then, what is the classification error? Say, draw a new sample of $\theta + \sigma Z'$ where Z' is also a Gaussian, then if $w \cdot (\theta + \sigma Z') \ge 0$, the prediction is correct. So, we want, with high probability, $w \cdot (\theta + \sigma Z') \ge 0$, i.e., $d \gg \sigma\sqrt{d} + \sigma^2 \frac{d}{n}$. When $n = 1$, as long as $\sigma^2 \le \sqrt{d}$, we can get non-robust classifier. However, things get bad when

we consider an adversary A: A obtain a sample from $\theta + \sigma Z'$ and push it into the negative direction of noise Z (bounded in l_∞ norm) you get on the sample. Since our Gaussian spheres are largely overlapping ($\sigma^2 \approx d$), a small perturbation can put a sample from a class to another class. Then, we have, with high probability, $w \cdot (\theta + \sigma Z' - Z) \leq 0$, which means that the classification error is large. Now we get $d \ll \frac{\sigma d}{\sqrt{n}}$ which is equivalent to $n \ll \sigma^2$. This implies that it is impossible to robust-learn if $n \leq \sigma^2$. What is the conclusion of this analysis? Let's consider an example with $\sigma^2 = d$. A single was sufficient for non-robust standard learning; but to learn robustly, we must see at least d samples, i.e., $\Omega(d)$ samples.

This polynomial separation result of [9] says that we need more data in polynomial for robust learning, but not this is not as bad as exponentially more data. Moreover, Bubeck et al. [3] discovers evidence against World 3: assuming that we know the set of distributions can be covered by N Wasserstein balls, $O(logN)$ samples suffices to find an information-theoretically robust classifier (provided that it exists). Going back to the above setting by [9], if you know that the distributions can be described by d parameters, then with roughly d samples, it is possible to find a robust classifier.

3.4 World 4: Finding robust classifiers is possible sample-efficiently and information-theoretically, but not computationally

Supporting World 4 is the work of Bubeck et al. [2, 3] and Degwekar et al. [5]. In BPLR, a cryptographic assumption of factoring is assumed and the following theorem is presented.

Theorem 2 (BPLR). *Assuming factoring, there is a classification task over $\{0,1\}^n$ where (1) it is easy to efficiently learn a non-robust classifier, (2) an efficient $O(1)$-robust classifier exists, but (3) learning any 1-robust classifier is hard.*

To prove Theorem 2, a binary classification problem is introduced, and the primary ingredient of their construction is *trapdoor* pseudorandom generator (PRG). Plain PRG: $\{0,1\}^n \to \{0,1\}^{l(n)}$ is an expanding function (i.e. expansion factor $l(n) \gg n$) whose output is computationally indistinguishable from random strings of length $l(n)$. Formally, for every poly-time distinguisher D that outputs 1 for PRG outputs or 0 otherwise,

$$|\mathbb{P}[D(PRG(x)) = 1] - \mathbb{P}[D(r) = 1]| \leq negl(n)$$

where the first probability is taken over uniform choice of $x \in \{0,1\}^n$ and randomness of D, and the second probability is taken over uniform choice of $r \in \{0,1\}^{l(n)}$ and the randomness of D.
In trapdoor PRG, knowing a key enables efficiently distinguishing the output of the PRG from random strings. That is,

$$|\mathbb{P}[D(TrapPRG(key, x)) = 1] - \mathbb{P}[D(key, r) = 1]| > 0.99$$

Their classification task is to distinguish samples from D_0, D_1 such that $D_0 = \{(0, TrapPRG(x)) : x \leftarrow \{0,1\}^n\}$ and $D_1 = \{(1, r) : r \leftarrow \{0,1\}^{l(n)}\}$. Obviously, non-robust classification is easy by simply looking at the first bit, thus proving Theorem2(1). Consider an adversary that modifies the first bit, then a robust classifier should be able to distinguish PRG outputs from random bit strings. By a volume argument (that is, the support of a PRG is exponentially smaller than the support of the uniform distribution), there exists a $\Omega(\sqrt{n})$-robust classifier. This robust classifier is not efficient if plain PRG is used, by the definition of PRG. However, by introducing trapdoor PRG, they show the existence of an efficient robust classifier. Expecially, Bubeck et al. shows that Blum-Blum-Shub PRG (B.B.S.) [1] has such a trapdoor, and knowing the n-bits factorization allows for efficient distinguishability. This implies that one can find an efficient robust classifier that tolerates constant-sized perturbations (no beyond $O(1)$ because the running time is exponential in the number of perturbed bits), thus proving (2). Of course, this is information-theoretical, and one cannot hope for any robust learning in poly-time due to the cryptographic assumption that integer factorization cannot be solved in poly-time, thus proving (3).

Degwekar et al. [5] relaxes the cryptographic assumption of BPLR into the minimal one, the existence of one-way functions, and allows larger adversarial perturbations.

Theorem 3 (Degwekar et al.). *Assuming one-way functions, there is a classification task over $\{0,1\}^n$ where (1) it is easy to efficiently learn a non-robust classifier, (2) an efficient $O(n)$-robust classifier exists, but (3) it is computationally hard to learn even a 1-robust classifier.*

The key ingredients of their construction to prove Theorem 3 are pseudorandom functions and error correcting codes. Consider a keyed function $F_k : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ for the uniformly chosen key $k \in \{0,1\}^n$. F_k is a pseudorandom function (PRF) if for all probabilistic polynomial-time distinguishers D, there is a negligible function $negl$ such that,

$$|\mathbb{P}[D^{F_k(\cdot)}(1^n) = 1]] - \mathbb{P}[D^{f(\cdot)}(1^n) = 1]| \leq negl(n)$$

where the first probability is taken over uniform choice of $k \in \{0,1\}^n$ and the randomness of D, and the second probability is taken over uniform choice of $f \in Func_n : \{0,1\}^n \to n$ and the randomness of D. On top of such PRF F_k, they construct two distributions D_0 and D_1 using Encode algorithm as follows,

$$D_0 = (0, \text{Encode}(x, F_k(x))) \quad \text{and} \quad D_1 = (1, \text{Encode}(x, 1 - F_k(x))) \tag{3}$$

This is similar to (1) of World 2 with g replaced by F_k. Samples from D_0 and D_1 are easy to classify non-robustly, by simply outputting the first bit. thus proving (1) of Theorem 3. If the first bit is corrupted by an adversary, the robust classification is identical to distinguishing Encode$(x, F_k(x))$ from Encode$(x, 1 - F_k(x))$. As a proof of (2), consider the following scenario:

- Flip a fair coin.
- If the head is up, draw a sample from D_0. If the tail is up, draw from D_1.
- Given the sample, a classifier h decode it using Decode algorithm.
- Given the key k, if the output of Decode is of the form $(x, F_k(x))$, h outputs 0. If the output is the form of $(x, 1 - F_k(x))$, then outputs 1.

The robustness of h against adversarial perturbations is guaranteed by the error correcting code and hence, (2) is proved. However, this only shows the existence of a efficient robust classifier h, but actually learning h is computationally intractable. This is because the process involves predicting PRF, which is impossible in poly-time by the assumption of one-way function, thus proving (3) of Theorem 3.

3.5 World 5: We just have not found the right algorithm yet

In addition to Theorem 3, Degwekar et al. also provides an *optimistic* interpretation of current state of affairs.

Theorem 4. *For any classification task when efficient robust classifiers exist, then either (1) they can be learned, or (2) we can construct cryptography (one-way functions and hence more).*

What does this Theorem 4 tell us? Since cryptographic assumptions are hard to find in the wild, if an efficient robust classifier (e.g., for image classification tasks) exists, then it is efficiently learnable and we just have not found the right algorithm yet.

4 Conclusion

In this paper, we surveyed the recent works that view the reason classifiers are susceptible to adversarial perturbations is due to the computational limitations of the problem itself or the learning algorithms. This viewpoint gives rise to five possible worlds of robust learning, and we went over the related works that provide evidence in favor of or against each of the worlds. If world 1 or world 2 is true, there is actually nothing we can do to improve the robustness. In world 3, we have seen that there is a small polynomial gap between the sample complexity of robust learning and standard learning. Based on this evidence, one might hope that once we get large enough data in the future, this adversarial vulnerability would be resolved naturally. The hypothesis of world 4 is true in general, but this may not be the case for the tasks we care about in practice. The classification tasks presented by Bubeck et al. and Degwekar et al. to support world 4 was basically to distinguish two different types of noises. However, the classification tasks in modern machine learning are about distinguishing structured data. Supporting hypothesis 4 with assumptions on the structure of data, rather than on classifiers may give new insights. Finally, world 5 gives us the most optimistic interpretation of why we are stuck at the moderate success of robust learning, and implies that there are lots of rooms for studying computational efficient learning algorithms for robust classifications.

References

[1] L. Blum, M. Blum, and M. Shub. A simple unpredictable pseudo-random number generator. *SIAM Journal on computing*, 15(2):364–383, 1986.

[2] S. Bubeck, Y. T. Lee, E. Price, and I. Razenshteyn. Adversarial examples from cryptographic pseudo-random generators. *arXiv preprint arXiv:1811.06418*, 2018.

[3] S. Bubeck, E. Price, and I. Razenshteyn. Adversarial examples from computational constraints. *arXiv preprint arXiv:1805.10204*, 2018.

[4] S. Bubeck, E. Price, and I. Razenshteyn. Adversarial examples from computational constraints. In *International Conference on Machine Learning*, pages 831–840, 2019.

[5] A. Degwekar, P. Nakkiran, and V. Vaikuntanathan. Computational limitations in robust classification and win-win results. *arXiv preprint arXiv:1902.01086*, 2019.

[6] G. Elsayed, S. Shankar, B. Cheung, N. Papernot, A. Kurakin, I. Goodfellow, and J. Sohl-Dickstein. Adversarial examples that fool both computer vision and time-limited humans. In *Advances in Neural Information Processing Systems*, pages 3910–3920, 2018.

[7] A. Fawzi, H. Fawzi, and O. Fawzi. Adversarial vulnerability for any classifier. In *Advances in Neural Information Processing Systems*, pages 1178–1187, 2018.

[8] P. Gourdeau, V. Kanade, M. Kwiatkowska, and J. Worrell. On the hardness of robust classification. In *Advances in Neural Information Processing Systems*, pages 7444–7453, 2019.

[9] 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.

[10] 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.