Adversarial Risk and Robustness: General Definitions and Implications for the Uniform Distribution*

Dimitrios I. Diochnos[†]
University of Virginia
diochnos@virginia.edu

Saeed Mahloujifar[†]
University of Virginia
saeed@virginia.edu

Mohammad Mahmoody[‡]
University of Virginia
mohammad@virginia.edu

Abstract

We study adversarial perturbations when the instances are uniformly distributed over $\{0,1\}^n$. We study both "inherent" bounds that apply to any problem and any classifier for such a problem as well as bounds that apply to specific problems and specific hypothesis classes.

As the current literature contains multiple definitions of adversarial risk and robustness, we start by giving a taxonomy for these definitions based on their direct goals; we identify one of them as the one guaranteeing misclassification by pushing the instances to the *error region*. We then study some classic algorithms for learning monotone conjunctions and compare their adversarial risk and robustness under different definitions by attacking the hypotheses using instances drawn from the uniform distribution. We observe that sometimes these definitions lead to *significantly different* bounds. Thus, this study advocates for the use of the error-region definition, even though other definitions, in other contexts with context-dependent assumptions, may coincide with the error-region definition.

Using the error-region definition of adversarial perturbations, we then study *inherent* bounds on risk and robustness of *any* classifier for *any* classification problem whose instances are uniformly distributed over $\{0,1\}^n$. Using the isoperimetric inequality for the Boolean hypercube, we show that for initial error 0.01, there always exists an adversarial perturbation that changes $O(\sqrt{n})$ bits of the instances to increase the risk to 0.5, making classifier's decisions meaningless. Furthermore, by also using the central limit theorem we show that when $n \to \infty$, at most $c \cdot \sqrt{n}$ bits of perturbations, for a universal constant $c < 1.17$, suffice for increasing the risk to 0.5, and the same $c \cdot \sqrt{n}$ bits of perturbations *on average* suffice to increase the risk to 1, hence bounding the robustness by $c \cdot \sqrt{n}$.

Contents

1 Introduction

In recent years, modern machine learning tools (e.g., neural networks) have pushed to new heights the classification results on traditional datasets that are used as testbeds for various machine learning methods.[1] As a result, the properties of these methods have been put into further scrutiny. In particular, studying the *robustness* of the trained models in various adversarial contexts has gained special attention, leading to the active area of *adversarial* machine learning.

Within adversarial machine learning, one particular direction of research that has gained attention in recent years deals with the study of the so-called *adversarial perturbations* of the test instances. This line of work was particularly popularized, in part, by the work of Szegedy et al. [SZS+14] within the context of deep learning classifiers, but the same problem can be asked for general classifiers as well. Briefly, when one is given a particular instance x for classification, an adversarial perturbation x' for that instance is a new instance with minimal changes in the features of x so that the resulting perturbed instance x' is misclassified by the classifier h. The perturbed instance x' is commonly referred to as an *adversarial example* (for the classifier h). Adversarial machine learning has its roots at least as back as in [LM05, NRH+10]. However, the work of [SZS+14] revealed pairs of images that differed slightly so that a human eye could not identify any real differences between the two, and yet, contrary to what one would naturally expect, machine learning classifiers would predict different labels for the classifications of such pairs of instances. It is perhaps this striking resemblance to the human eye of the pairs of images that were provided in [SZS+14] that really gave this new push for intense investigations within the context of adversarial perturbations. Thus, a very intense line of work started, aiming to understand and explain the properties of machine learning classifiers

---

[1]For example, http://rodrigob.github.io/are_we_there_yet/build/ has a summary of state-of-the-art results.

on such adversarial perturbations; e.g., [GSS15, MFF16, BIL+16, CW17b, MMS+18]. These attacks are also referred to as *evasion* attacks [NRH+12, BFR14, GSS15, CW17b, XEQ18]. There is also work that aims at making the classifiers more robust under such attacks [PMW+16, XEQ18], yet newer attacks of Carlini and Wagner [CW17a] broke many proposed defenses.

Our general goal.   In this work, we study barriers against robust classification of adversarial examples. We are particularly interested in foundational bounds that potentially apply to broad class of problems and distributions. One can study this question from the perspectives of both risk and robustness. In the case of risk, the adversary's goal is to increase the error probability of the classifier (e.g., to reach risk 0.5) by small perturbations of the instances, and in the case of robustness, we are interested in the *average* amount of perturbations needed for making the classifier always fail.

Studying the uniform distribution.   We particularly study adversarial risk and robustness for learning problems where the input distribution is U_n which is uniform over the hypercube $\{0,1\}^n$. We measure the cost of perturbations using the natural metric of Hamming distance. Namely, the distance between the original and perturbed instances $x, x' \in \{0,1\}^n$ is the number of locations that they are different. This class of distributions already include many learning problems of interest. So, by studying adversarial risk and robustness for such a natural distribution, we can immediately obtain results for a broad class of problems. We believe it is crucial to understand adversarial risk and robustness for natural distributions (e.g., U_n uniform over the hypercube) and metrics (e.g., the Hamming distance) to develop a theory of adversarial risk and robustness that can ultimately shed light on the power and limitations of robust classification for practical data sets. Furthermore, natural distributions like U_n model a broad class of learning problems directly; e.g., see [BFJ+94, SM00, JS06, Sel09]. The hope is that understanding the limitations of robust learning for these basic natural distributions will ultimately shed light on challenges related to addressing broader problems of interest.

Related previous work.   The work of Gilmer et al. [GMF+18] studied the above problem for the special case of input distributions that are uniform over unit spheres in dimension n. They showed that for any classification problem with such input distribution, so long as there is an initial constant error probability μ, the robustness under the ℓ_2 norm is at most $O(\sqrt{n})$. Fawzi et al. [FFF18] studied the above question for Gaussian distributions in dimension n and showed that when the input distribution has ℓ_2 norm ≈ 1, then by $\approx \sqrt{n}$ perturbations in ℓ_2 norm, we can make the classifier *change its prediction* (but doing this does not guarantee that the perturbed instance x' will be misclassified). Schmidt et al. [SST+18] proved limits on robustness of classifying uniform instances by specific classifiers and using a definition based on "corrupted inputs" (see Section 2), while we are mainly interested in bounds that apply to any classifiers and guarantee misclassification of the adversarial inputs.

Discussion.   Our negative results of this work, like other (current proved) bounds in the literature for adversarial risk and robustness only apply to specific distributions that do not cover the case of distributions that generate images, voices, or other practically interesting data. We see these results, however, as first steps towards understanding the barriers against robustness. The negative results so far indicate similar phenomena (e.g., relation to isoperimetric inequalities). Thus, as pointed out in [GMF+18], these works motivate a deeper study of such inequalities for real data sets. Finally, as discussed in [FFF18], such theoretical attacks could *potentially* imply direct attacks on real data, *assuming* the existence of smooth generative models for latent vectors with theoretically nice distributions (e.g., Gaussian or uniform over $\{0,1\}^n$) into natural data.

1.1 Our Contribution and Results

As mentioned above, our main goal is to understand inherent barriers against robust classification of adversarial examples, and our focus is on the uniform distribution U_n of instances. In order to achieve that goal, we both do a definitions study and prove technical limitation results.

General definitions and a taxonomy. As the current literature contains multiple definitions of adversarial risk and robustness, we start by giving a taxonomy for these definitions based on their direct goals. More specifically, suppose x is an original instance that the adversary perturbs into a "close" instance x'. Suppose $h(x), h(x')$ are the predictions of the hypothesis $h(\cdot)$ and $c(x), c(x')$ are the true labels of x, x' defined by the concept function $c(\cdot)$. To call x' a successful "adversarial example", a natural definition would compare the predicted label $h(x')$ with some other "anticipated answer". However, what $h(x')$ is exactly compared to is where various definitions of adversarial examples diverge. We observe in Section 2 that the three possible definitions (based on comparing $h(x')$ with either of $h(x)$, $c(x)$ or $c(x')$) lead to three different ways of defining adversarial risk and robustness. We then identify one of them (that compares $h(x)$ with $c(x')$) as the one guaranteeing misclassification by pushing the instances to the *error region*. We also discuss natural conditions under which these definitions coincide. However, these conditions do not hold *in general*.

A comparative study through monotone conjunctions. We next ask: how close/far are these definitions in settings where, e.g., the instances are drawn from the uniform distribution? To answer this question, we make a comparative study of adversarial risk and robustness for a particular case of learning monotone conjunctions under the uniform distribution U_n (over $\{0,1\}^n$). A monotone conjunction f is a function of the form $f = (x_{i_1} \wedge \cdots \wedge x_{i_k})$. This class of functions is perhaps one of the most natural and basic learning problems that are studied in computational learning theory as it encapsulates, in the most basic form, the class of functions that determine which features should be included as relevant for a prediction mechanism. For example, Valiant in [Val09] used this class of functions under U_n to exemplify the framework of evolvability. We attack monotone conjunctions under U_n in order to contrast different behavior of definitions of adversarial risk and robustness.

In Section 3, we show that previous definitions of robustness that are not based on the error region, lead to bounds that do *not* equate the bounds provided by the error-region approach. We do so by first deriving theorems that characterize the adversarial risk and robustness of a given hypothesis and a concept function under the uniform distribution. Subsequently, by performing experiments we show that, on average, hypotheses computed by two popular algorithms (FIND-S [Mit97] and SWAPPING ALGORITHM [Val09]) also exhibit the behavior that is predicted by the theorems. Estimating the (expected value of) the adversarial risk and robustness of hypotheses produced by *other* classic algorithms under specific distributions, or for other concept classes, is an interesting future work.

Inherent bounds for any classification task under the uniform distribution. Finally, after establishing further motivation to use the error-region definition as the default definition for studying adversarial examples in *general* settings, we turn into studying *inherent* obstacles against robust classification when the instances are drawn from the uniform distribution. We prove that for *any* learning problem P with input distribution U_n (i.e., uniform over the hypercube) and for any classifier h for P with a constant error μ, the robustness of h to adversarial perturbations (in Hamming distance) is at most $O(\sqrt{n})$. We also show that by the same amount of $O(\sqrt{n})$ perturbations *in the worst case*, one can increase the risk to 0.99. Table 1 lists some numerical examples.

4

Table 1: Each row focuses on the number of tampered bits to achieve its stated goal. The second column shows results using direct calculations for specific dimensions. The third column shows that these results are indeed achieved in the limit, and the last column shows bounds proved for all n.

Adversarial goals	Types of bounds		
	$n = 10^3, 10^4, 10^5$	$n \mapsto \infty$	all n
From initial risk 0.01 to 0.99	$\approx 2.34\sqrt{n}$	$< 2.34\sqrt{n}$	$< 3.04\sqrt{n}$
From initial risk 0.01 to 0.50	$\approx 1.17\sqrt{n}$	$< 1.17\sqrt{n}$	$< 1.52\sqrt{n}$
Robustness for initial risk 0.01	$\approx 1.17\sqrt{n}$	$< 1.17\sqrt{n}$	$< 1.53\sqrt{n}$

To prove results above, we apply the isoperimetric inequality of [Nig67, Har66] to the error region of the classifier h and the ground truth c. In particular, it was shown in [Har66, Nig67] that the subsets of the hypercube with minimum "expansion" (under Hamming distance) are Hamming balls. This fact enables us to prove our bounds on the risk. We then prove the bounds on robustness by proving a general connection between risk and robustness that might obe of independent interest. Using the central limit theorem, we sharpen our bounds for robustness and obtain bounds that closely match the bounds that we also obtain by direct calculations (based on the isoperimetric inequalities and picking Hamming balls as error region) for specific values of dimension $n = 10^3, 10^4, 10^5$.

2   General Definitions of Adversarial Risk and Robustness

Notation. We use calligraphic letters (e.g., $\mathcal{X}$) for sets and capital non-calligraphic letters (e.g., D) for distributions. By $x \leftarrow D$ we denote sampling x from D. In a classification problem $\mathsf{P} = (\mathcal{X}, \mathcal{Y}, \mathcal{D}, \mathcal{C}, \mathcal{H})$, the set $\mathcal{X}$ is the set of possible *instances*, $\mathcal{Y}$ is the set of possible *labels*, $\mathcal{D}$ is a set of distributions over $\mathcal{X}$, $\mathcal{C}$ is a class of *concept* functions, and $\mathcal{H}$ is a class of *hypotheses*, where any $f \in \mathcal{C} \cup \mathcal{H}$ is a mapping from $\mathcal{X}$ to $\mathcal{Y}$. An *example* is a *labeled instance*. We did not state the loss function explicitly, as we work with classification problems, however all main three definitions of this section directly extend to arbitrary loss functions. For $x \in \mathcal{X}, c \in \mathcal{C}, D \in \mathcal{D}$, the *risk* or *error* of a hypothesis $h \in \mathcal{H}$ is the expected (0-1) loss of (h, c) with respect to D, namely $\mathrm{Risk}(h, c, D) = \Pr_{x \leftarrow D}[h(x) \neq c(x)]$. We are usually interested in learning problems with a fixed distribution $\mathcal{D} = \{D\}$, as we are particularly interested in robustness of learning under the uniform distribution U_n over $\{0, 1\}^n$. Note that since we deal with negative results, fixing the distribution only makes our results stronger. As a result, whenever $\mathcal{D} = \{D\}$, we omit D from the risk notation and simply write $\mathrm{Risk}(h, c)$. We usually work with problems $\mathsf{P} = (\mathcal{X}, \mathcal{Y}, D, \mathcal{C}, \mathcal{H}, \mathbf{d})$ that include a metric $\mathbf{d}$ over the instances. For a set $\mathcal{S} \subseteq \mathcal{X}$ we let $\mathbf{d}(x, \mathcal{S}) = \inf\{\mathbf{d}(x, y) \mid y \in \mathcal{S}\}$ and $\mathcal{B}all_r(x) = \{x' \mid \mathbf{d}(x, x') \leq r\}$. By HD we denote Hamming distance for pairs of instances from $\{0, 1\}^n$. Finally, we use the term *adversarial instance* to refer to an adversarially perturbed instance x' of an originally sampled instance x when the label of the adversarial example is either not known or not considered.

2.1   Different Definitions and Qualitative Comparisons

Below we present formal definitions of adversarial risk and robustness. In all of these definitions we will deal with attackers who perturb the initial test instance x into a *close* adversarial instance x'. We will measure how much an adversary can increase the *risk* by perturbing a given input x into a *close* adversarial example x'. These definitions differ in when to call x' a successful adversarial example. First we formalize the main definition that we use in this work based on adversary's ability to push instances to the error region.

Definition 2.1 (Error-region risk and robustness). *Let* $\mathsf{P} = (\mathcal{X}, \mathcal{Y}, D, \mathcal{C}, \mathcal{H}, \mathbf{d})$ *be a classification problem (with metric $\mathbf{d}$ defined over instances $\mathcal{X}$).*

- **Risk.** *For any $r \in \mathbf{R}_+, h \in \mathcal{H}, c \in \mathcal{C}$, the* error-region risk *under r-perturbation is*

$$\mathsf{Risk}_r^{\mathrm{ER}}(h, c) = \Pr_{x \leftarrow D}[\exists x' \in \mathcal{B}all_r(x), h(x') \neq c(x')].$$

For $r = 0$, $\mathsf{Risk}_r^{\mathrm{ER}}(h, c) = \mathsf{Risk}(h, c)$ becomes the standard notion of risk.

- **Robustness.** *For any $h \in \mathcal{H}, x \in \mathcal{X}, c \in \mathcal{C}$, the* error-region robustness *is the expected distance of a sampled instance to the error region, formally defined as follows*

$$\mathsf{Rob}^{\mathrm{ER}}(h, c) = \mathbf{E}_{x \leftarrow D}\left[\inf\{r \colon \exists x' \in \mathcal{B}all_r(x), h(x') \neq c(x')\}\right].$$

Definition 2.1 requires the adversarial instance x' to be *misclassified*, namely, $h(x') \neq c(x')$. So, x' clearly belongs to the error region of the hypothesis h compared to the ground truth c. This definition is implicit in the work of [GMF$^+$18]. In what follows, we compare our main definition above with previously proposed definitions of adversarial risk and robustness found in the literature and discuss when they are (or when they are not) equivalent to Definition 2.1. Figure 1 summarizes the differences between the three main definitions that have appeared in the literature, where we distinguish cases by comparing the classifier's prediction $h(x')$ at the new point x' with either of $h(x)$, $c(x)$, or $c(x')$, leading to three different definitions.
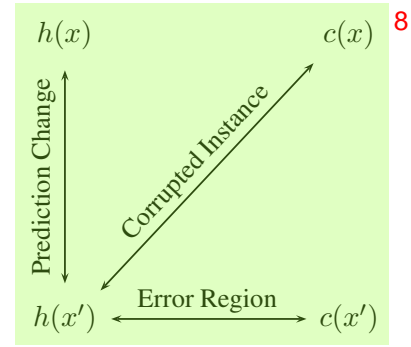

Figure 1: The three main definitions based on what $h(x')$ is compared with.

Definitions based on hypothesis's prediction change (PC risk and robustness). Many works, including the works of [SZS$^+$14, FFF18] use a definition of robustness that compares classifier's prediction $h(x')$ with the prediction $h(x)$ on the original instance x. Namely, they require $h(x') \neq h(x)$ rather than $h(x') \neq c(x')$ in order to consider x' an adversarial instance. Here we refer to this definition (that does not depend on the ground truth c) as *prediction-change* (PC) risk and robustness (denoted as $\mathsf{Risk}_r^{\mathrm{PC}}(h)$ and $\mathsf{Rob}^{\mathrm{PC}}(h)$). We note that this definition captures the error-region risk and robustness if we *assume* the initial correctness (i.e., $h(x) = c(x)$) of classifier's prediction on all $x \leftarrow X$ and "truth proximity", i.e., that $c(x) = c(x')$ holds for all x' that are "close" to x. Both of these assumptions are valid in some natural scenarios. For example, when input instances consist of images that look similar to humans (if used as the ground truth $c(\cdot)$) and if h is also correct on the original (non-adversarial) test examples, then the two definitions (based on error region or prediction change) coincide. But, these assumptions do not hold in *in general*.

We note that there is also a work in the direction of finding adversarial instances that may potentially fool humans that have limited time to decide for their label, as in [ESC$^+$18]. The images of [ESC$^+$18] are sufficiently 'confusing' that answers of the form "I do not know" are very plausible from the humans that are asked. This fuzzy classification that allows "I do not know" answers is reminiscent of the *limited membership query* model of Sloan and Turán [ST94] (which is a worst-case version of the *incomplete membership query* model of Angluin and Slonim [AS94]; see also [AKST97] and [SST10] for further related discussions) as well as of the model of learning from a *consistently ignorant teacher* of Frazier et al. [FGMP96].

Definitions based on the notion of corrupted instance (CI risk and robustness). The works of [MRT15, FMS15, FMS18, AKM18] study the robustness of learning models in the presence of *corrupted inputs*. A more recent framework was developed in [MMS+18, SST+18] for modeling risk and robustness that is inspired by robust optimization [BTGN09] (with an underlying metric space) and model adversaries that corrupt the the original instance in (exponentially more) ways. When studying adversarial perturbations using corrupted instances, we define adversarial risk by requiring the adversarial instance x' to satisfy $h(x') \neq c(x)$. The term "corrupted instance" is particularly helpful as it emphasizes on the fact that the goal (of the classifier) is to find the *true* label of the *original* (uncorrupted) instance x, while we are only given a corrupted version x'. Hence, we refer to this definition as the *corrupted instance* (CI) risk and robustness and denote them by $\mathsf{Risk}_r^{\mathrm{CI}}(h, c)$ and $\mathsf{Rob}^{\mathrm{CI}}(h, c)$. The advantage of this definition compared to the prediction-change based definitions is that here, we no longer need to assume the initial correctness assumption. Namely, only if the "truth proximity" assumption holds, then we have $c(x) = c(x')$ which together with the condition $h(x') \neq c(x)$ we can conclude that x' is indeed misclassified. However, if small perturbations can change the ground truth, $c(x')$ can be different from $c(x)$, in which case, it is no long clear whether x' is misclassified or not.

Stronger definitions with more restrictions on adversarial instance. The corrupted-input definition requires an adversarial instance x' to satisfy $h(x') \neq c(x)$, and the error-region definition requires $h(x') \neq c(x')$. What if we require *both* of these conditions to call x' a true adversarial instance? This is indeed the definition used in the work of Suggala et al. [SPNR18], though more formally in their work, they subtract the original risk (without adversarial perturbation) from the adversarial risk. This definition is certainly a *stronger* guarantee for the adversarial instance. Therefore, we simply refer to risk and robustness under this condition as *strong* adversarial risk and robustness. As this definition is a hybrid of the error-region and corrupted-instance definitions, we do not make a direct study of this definition and only focus on the other three definitions described above.

How about when the classifier h is 100% correct? We emphasize that when h happens to be the same function as c, (the error region) Definition 2.1 implies h has zero *adversarial* risk and infinite adversarial robustness $\mathsf{Rob}^{\mathrm{ER}}(h, c) = \infty$. This is expected, as there is no way an adversary can perturb any input x into a misclassified x'. However, both of the definitions of risk and robustness based on prediction change [SZS+14] and corrupted instance [MRT15, MMS+18] could compute large risk and small robustness for such h. In fact, in a recent work [TSE+18] it is shown that for definitions based on corrupted input, correctness might be *provably at odds* with robustness in some cases. Therefore, even though all these definitions could perhaps be used to approximate the risk and robustness when we do not have access to the ground truth c' on the new point x', in this work we separate the *definition* of risk and robustness from how to compute/approximate them, so we will use Definition 2.1 by default.

2.2 Various Aspects of the Attack Models

We emphasize that the definitions of Section 2.1 are all *information theoretic* and do not address the *efficiency* of the adversary who perturbs the original instance x into x'. Moreover, there are other aspects of the attack that are implicit in the definitions Section 2.1 in terms of what adversary does or does not have access to during the course of the attack. Below, we briefly point out these other aspects.

- **Efficiency.** This aspect of an attack could come in two flavor. One way to mathematically formalize "efficient" attacks is to use polynomial-time attacks as it is done in cryptography. Another way is to

use information theoretic attacks without the efficiency requirements. Security against information theoretic attacks are stronger, while attacks of polynomial-time form are stronger.

- **Information access.** The other aspect of the attack is about *what adversary has access to* during the attack and *how it can access* this information. We separate thes aspects as follows.

  - **What to access.** In general, we can consider attacks that do or do not access to either of the ground truth c, the hypothesis h, or distribution D.
  - **How to access.** If the attack can depend on a function f (e.g., $f = h$ or $f = c$) or a distribution D it can still access this information in various forms. An information theoretic attack can completely depend on the full description of f, while an efficient (polynomial time attack) can use oracle access to f (regardless of efficiency of f itself) or a sampler for D. In fact, if f (or a sampler for a distribution D) has a compact representation, then an efficient attacker can also fully depend on f or D if that representation is given.

Going back to the definitions of Section 2, by "$\exists x' \in \mathcal{B}all_r(x), P(x')$" we simply state the *existence* of a close instance x' with a property $P(x')$ while it might be computationally infeasible to actually *find* such an x'. Moreover, the definitions of Section 2 assume the adversary has full access to f, c, D.

3   A Comparative Study through Monotone Conjunctions

In this section, we compare the risk and robustness under the three definitions of Section 2 through a study of monotone conjunctions under the uniform distribution. Namely, we consider adversarial perturbations of truth assignments that are drawn from the uniform distribution U_n over $\{0,1\}^n$ when the concept class contains monotone conjunctions. As we will see, these definitions diverge in this natural case. Below we fix the setup under which all the subsequent results are obtained.

Problem Setup 1. *Let $\mathcal{C}_n$ be the concept class of all monotone conjunctions formed by at least one and at most n Boolean variables. The target concept (ground truth) c that needs to be learned is drawn from $\mathcal{C}_n$. Let the hypothesis class be $\mathcal{H} = \mathcal{C}_n$ and let $h \in \mathcal{H}$ be the hypothesis obtained by a learning algorithm after processing the training data. With $|h|$ and $|c|$ we denote the* size *of h and c respectively; that is, number of variables that h and c contain.[2] Now let,*

$$c = \bigwedge_{i=1}^{m} x_i \wedge \bigwedge_{k=1}^{u} y_k \qquad and \qquad h = \bigwedge_{i=1}^{m} x_i \wedge \bigwedge_{\ell=1}^{w} z_\ell. \tag{1}$$

We will call the variables that appear both in h and c as mutual, *the variables that appear in c but not in h as* undiscovered, *and the variables that appear in h but not in c as* wrong *(or* redundant*). Therefore in (1) we have m mutual variables, u undiscovered and w wrong. We denote the error region of a hypothesis h and the target concept c with $\mathcal{E}(h, c)$.*

That is, $\mathcal{E}(h, c) = \{x \in \{0,1\}^n \mid h(x) \neq c(x)\}$. The probability mass of the error region between h and c, denoted by μ, under the uniform distribution U_n over $\{0,1\}^n$ is then,

$$\Pr_{x \leftarrow U_n}[x \in \mathcal{E}(h, c)] = \mu = (2^w + 2^u - 2) \cdot 2^{-m-u-w}. \tag{2}$$

---

[2] For example, $h_1 = x_1 \wedge x_5 \wedge x_8$ is a monotone conjunction of three variables in a space where we have $n \geq 8$ variables and $|h_1| = 3$.

In this problem setup we are interested in computing the adversarial risk *and* robustness *that attackers can achieve when instances are drawn from the uniform distribution U_n over $\{0, 1\}^n$.*

Remark 3.1. *Note that μ is a variable that depends on the particular h and c.*

Remark 3.2. *Connecting to our discussion from Section 2.2, adversaries who have oracle access to such hypotheses can identify the variables that appear in them and so reconstruct them efficiently as follows. Since $x_{one} = \langle 1, 1, 1, \ldots, 1 \rangle$ is always positive one can query all the n vectors that have all 1's but one 0, and thus determine all the variables that appear in a particular hypothesis.*

Using the Problem Setup 1, in what follows we compute the adversarial risk and robustness that an arbitrary hypothesis has against an arbitrary target using the *error region (ER)* definition that we advocate in contexts where the perturbed input is supposed to be misclassified and do the same calculations for adversarial risk and robustness that are based on the definitions of *prediction change (PC)* and *corrupted instance (CI)*. The adversarial robustness of a solution using the prediction change and corrupted instance definitions is proportional to the size (number of variables) of the learned solution; experimentally we obtain that it is proportional to about *half* the number of variables that the hypothesis contains. On the other hand, the adversarial robustness of a learned solution using the error region definition is proportional to the *minimum* between the size of the hypothesis and the size of the target; experimentally we obtain that it is proportional to about *half* that minimum value. In other words, for a hypothesis that has many variables and a target that has fairly few variables, the prediction change and corrupted instance definitions imply large robustness for the learned solution, whereas the adversarial robustness as implied by the error region definition is low. This last setup is precisely the case for a large set of target functions when PAC learning monotone conjunctions with the FIND-S algorithm; see Section 3.4

3.1 Error Region Risk and Robustness

Theorem 3.3 (Error region risk - lower bound). *Consider the Problem Setup 1. Then, if $h = c$, we have $\mathrm{Risk}_r^{\mathrm{ER}}(h, c) = 0$, while if $h \neq c$, then, with a perturbation budget of r we can obtain the following lower bounds.*

- *If $0 \le r \le \lfloor m/2 \rfloor$, then $\mathrm{Risk}_r^{\mathrm{ER}}(h, c) \ge \mu \cdot \sum_{j=0}^{r} \binom{m}{j}$, where μ is given by (2).*

- *If $\lfloor \frac{m}{2} \rfloor + 1 \le r = \lfloor \frac{m}{2} \rfloor + \gamma \le \lfloor \frac{m}{2} \rfloor + \lfloor \frac{\min\{u,w\}}{2} \rfloor$, then*

$$\mathrm{Risk}_r^{\mathrm{ER}}(h, c) \ge \frac{1}{4} \cdot 2^{-\min\{u,w\}} \cdot \sum_{\zeta=1}^{\gamma} \binom{\min\{u, w\}}{\zeta}.$$

- *If $\lfloor \frac{m}{2} \rfloor + \lfloor \frac{\min\{u,w\}}{2} \rfloor + 1 \le r \le \min\{|h|, |c|\}$, then $\mathrm{Risk}_r^{\mathrm{ER}}(h, c) \ge \frac{1}{8}$.*

- *If $1 + \min\{|h|, |c|\} \le r$, then $\mathrm{Risk}_r^{\mathrm{ER}}(h, c) = 1$.*

Proof Sketch. Assume that $h \neq c$, since otherwise the risk is 0.

We distinguish cases for the various values that m, u, w can take in (1): (i) $m = 0$, (ii) $m \ge 1$ and $u = 0$ and $w \ge 1$, (iii) $m \ge 1$ and $u \ge 1$ and $w = 0$, and finally, (iv) **the more general case, where $m \ge 1$, $u \ge 1$ and $w \ge 1$.** Below we will prove fully the more involved case, **case (iv)**; the other cases can easily be obtained using (iv) as a guide. Furthermore, we distinguish between two main cases: having a budget $r \le \min\{|h|, |c|\}$ versus having a budget of $r > \min\{|h|, |c|\}$.

Case 1: Budget $0 \le r \le \min\{|h|, |c|\}$. We distinguish cases based on the relationship between the prediction and the true label of a randomly drawn instance.

Case 1A: Budget $0 \le r \le \min\{|h|, |c|\}$ and instance x such that $h(x) \ne c(x)$. When h and c disagree, this means that $x \in \mathcal{E}(h, c)$ and hence without the need of any budget (i.e., $r = 0$) all these truth assignments contribute to the risk of the hypothesis. By (2) we obtain such an x with probability $(2^w + 2^u - 2) \cdot 2^{-m-u-w} = \mu = \mu \cdot \binom{m}{0}$.

Case 1B: Budget $0 \le r \le \min\{|h|, |c|\}$ and instance x such that $h(x) = c(x) = 1$. With probability 2^{-m-u-w}, x satisfies both c and h. Since $h \ne c$ there is at least one variable that appears in either c or h but not in both. Therefore, with a budget of $r \ge 1$ we can flip that one bit in x that corresponds to that variable and the resulting x' will be misclassified.

Case 1C: Budget $0 \le r \le \min\{|h|, |c|\}$ and instance x such that $h(x) = c(x) = 0$. With probability $1 - \mu - 2^{-m-u-w}$, x falsifies both c and h. We distinguish cases further based on the range of the budget r that is provided to us.

Case 1C1: $0 \le r \le \lfloor m/2 \rfloor$. We look at truth assignments that have $1 \le j \le r \le \lfloor m/2 \rfloor$ 0's among the m mutual variables and further have: *(i)* $1 \le \zeta \le u$ undiscovered variables falsified and all the w wrong variables satisfied, and *(ii)* all the u undiscovered variables satisfied and $1 \le \xi \le w$ wrong variables falsified. With a budget of r, the contribution to the adversarial risk by these assignments is,

$$2^{-m-u-w} \sum_{j=1}^{r} \binom{m}{j} \left[\binom{w}{0} \sum_{\zeta=1}^{u} \binom{u}{\zeta} + \binom{u}{0} \sum_{\xi=1}^{w} \binom{w}{\xi} \right]$$

which is, $\left[2^{-m-u} + 2^{-m-w} - 2^{1-m-u-w} \right] \sum_{j=1}^{r} \binom{m}{j}$ and by (2) it is $\mu \cdot \sum_{j=1}^{r} \binom{m}{j}$.

Case 1C2: $\lfloor \frac{m}{2} \rfloor + 1 \le r \le \lfloor \frac{m}{2} \rfloor + \lfloor \frac{\min\{u,w\}}{2} \rfloor$. Assume[3] that $1 \le u \le w \Rightarrow |c| \le |h|$. We look at the truth assignments where we have at most $\lfloor m/2 \rfloor$ 0's among the m mutual variables and further we have γ 0's where $\gamma \ge 1$ among the u undiscovered variables plus at least one 0 among the w wrong variables. With a budget of $r = \lfloor \frac{m}{2} \rfloor + \gamma$ we flip the 0's that exist among the mutual variables plus the γ undiscovered variables that are currently falsified in x and thus we hit the error region. Therefore, the contribution to the adversarial risk by these truth assignments alone is,

$$2^{-m-u-w} \sum_{j=0}^{\lfloor m/2 \rfloor} \binom{m}{j} \sum_{\xi=1}^{w} \binom{w}{\xi} \sum_{\zeta=1}^{\gamma} \binom{u}{\zeta}$$

which is, $\frac{1}{2} \cdot (1 - 2^{-w}) \cdot 2^{-u} \cdot \sum_{\zeta=1}^{\gamma} \binom{u}{\zeta} \ge \frac{1}{4} \cdot 2^{-u} \cdot \sum_{\zeta=1}^{\gamma} \binom{u}{\zeta}$. (In the opposite case, where $1 \le w < u$ we obtain a lower bound of $\frac{1}{4} \cdot 2^{-w} \cdot \sum_{\zeta=1}^{\gamma} \binom{u}{\zeta}$ that explains the statement of the theorem.)

Case 1C3: $\lfloor \frac{m}{2} \rfloor + \lfloor \frac{\min\{u,w\}}{2} \rfloor + 1 \le r \le m + \min\{u, w\}$. Assume that $u \le w$. (Again, the opposite case is symmetric.)

---

[3] The opposite case where $w \le u$ is symmetric. Also, $u, w > 0$ since we explore case (iv), the more general case for the proof, where $m \ge 1$, $u \ge 1$ and $w \ge 1$.

We look at the truth assignments where we have at most $\lfloor m/2 \rfloor$ many 0's among the m mutual variables and further we have at most $1 + \lfloor u/2 \rfloor$ many 0's among the u undiscovered variables plus at least one 0 among the w wrong variables. With a budget of $r \geq \lfloor \frac{m}{2} \rfloor + \lfloor u/2 \rfloor + 1$ we flip the 0's that exist among the mutual variables plus the 0's that exist among the undiscovered variables and thus we hit the error region. Therefore, the contribution to the adversarial risk by these truth assignments alone is,

$$2^{-m-u-w} \sum_{j=0}^{\lfloor m/2 \rfloor} \binom{m}{j} \sum_{\zeta=1}^{1+\lfloor u/2 \rfloor} \binom{u}{\zeta} \sum_{\xi=1}^{w} \binom{w}{\xi}.$$

By Lemma A.1, $\sum_{j=0}^{\lfloor m/2 \rfloor} \binom{m}{j} \geq 2^{m-1}$ as well as $\sum_{\zeta=1}^{1+\lfloor u/2 \rfloor} \binom{u}{\zeta} \geq \sum_{\zeta=0}^{\lfloor u/2 \rfloor} \binom{u}{\zeta} \geq 2^{u-1}$, we obtain the lower bound, $2^{-m-u-w} \cdot 2^{m-1} \cdot 2^{u-1} \cdot (2^w - 1) = \frac{1}{4} \cdot (1 - 2^{-w}) \geq \frac{1}{8}$.

Case 2: Budget $r \geq 1 + \min\{|h|, |c|\}$. The risk 1 since we can hit the error region by making at most $1 + \min\{|h|, |c|\}$ changes in any given truth assignment. This worst case scenario can be observed when h is a specialization of c (or vice versa) and in particular for truth assignments where all the m mutual variables are falsified as well as all the wrong (resp., undiscovered) variables are satisfied. Then, we need to flip the $m = \min\{|h|, |c|\}$ mutual variables plus one more among the wrong (resp., undiscovered) in order to hit the error region. $\qquad\square$

Theorem 3.4 (Error region robustness). *Consider the Problem Setup 1. Then,* $\text{Rob}^{\text{ER}}(h, c) = \infty$ *when* $h = c$, *while if* $h \neq c$ *we have,*

$$\frac{1}{16} \cdot \min\{|h|, |c|\} \leq \text{Rob}^{\text{ER}}(h, c) \leq 1 + \min\{|h|, |c|\}.$$

Proof Sketch. $h = c \Rightarrow \text{Rob}^{\text{ER}}(h, c) = \infty$. Hence, below we will examine the case where $h \neq c$.

As in Theorem 3.3 below we prove fully case (iv) where $m \geq 1$, $u \geq 1$ and $w \geq 1$. Using the analysis that we present below for case (iv) as a guide, we can easily show for the other cases that they also satisfy $\min\{|h|, |c|\}/16 \leq \text{Rob}^{\text{ER}}(h, c) \leq 1 + \min\{|h|, |c|\}.$.

Case $m \geq 1$, $u \geq 1$ and $w \geq 1$. We can ignore the instances that are drawn from the error region. If $x \in \mathcal{E}(h, c)$, then these instances do not increase the robustness of h. By (2) such instances are drawn with probability $\mu = (2^w + 2^u - 2) \cdot 2^{-m-u-w}$.

Lower Bound. We examine the case where $|c| \leq |h|$. The case $|c| > |h|$ can be handled symmetrically. First, $|c| \leq |h| \Rightarrow u \leq w$. We now look at the following falsifying truth assignments in order to obtain a lower bound. Let x falsify $1 \leq j \leq m$ variables that mutually appear between h and c. Further, let x falsify $\zeta \geq 1$ more variables among the u that appear only in c and $\xi \geq 0$ more variables among the w that appear only in h. Then, we can perturb x into an $x' \in \mathcal{E}(h, c)$ by changing $\sigma = j + \min\{\zeta, \xi\}$ coordinates. The contribution to the overall robustness is then,

$$2^{-m-w-u} \sum_{j=1}^{m} \sum_{\zeta=1}^{u} \sum_{\xi=0}^{r} \binom{m}{j} \binom{u}{\zeta} \binom{w}{\xi} (j + \min\{\zeta, \xi\}).$$

11

Letting the above quantity be Q we have,

$$
\begin{aligned}
Q = \ & 2^{-m-w-u} \sum_{j=1}^{m} j\binom{m}{j} \sum_{\zeta=1}^{u} \binom{u}{\zeta} \sum_{\xi=0}^{w} \binom{w}{\xi} \\
& + 2^{-m-w-u} \sum_{j=1}^{m} \binom{m}{j} \sum_{\zeta=1}^{u} \zeta\binom{u}{\zeta} \sum_{\xi=\zeta}^{w} \binom{w}{\xi} \\
& + 2^{-m-w-u} \sum_{j=1}^{m} \binom{m}{j} \sum_{\zeta=1}^{u} \binom{u}{\zeta} \sum_{\xi=0}^{\zeta-1} \xi\binom{w}{\xi}.
\end{aligned}
$$

where, by dropping the last term and applying Lemma A.5 we obtain, $Q \geq 2^{-m-u}(2^u - 1)m2^{m-1} + \frac{u}{8} \cdot 2^{u+w} \cdot 2^{-m-w-u} \sum_{j=1}^{m} \binom{m}{j}$. Thus, $Q \geq \frac{m}{2}(1 - 2^{-u}) + \frac{u}{8}(1 - 2^{-m}) \geq \min\{|h|, |c|\}/16$.

Upper Bound. First, if $x \in \mathcal{E}(h, c)$, then we need to modify $\sigma = 0$ coordinates in x so that the perturbed instance x' is in the error region. Second, if x is a satisfying truth assignment for both h and c, then since h and c differ in at least one variable, making that variable in x equal to 0 will result in an $x' \in \mathcal{E}(h, c)$. Therefore, we need to modify $\sigma = 1$ coordinates in x in this case so that $x' \in \mathcal{E}(h, c)$. Finally, if x is a falsifying truth assignment for both h and c, then the minimum perturbation is obtained by flipping $\sigma \leq (1 + \min\{|h|, |c|\})$ 0's to 1's. Therefore, we need to perturb $\sigma \leq 1 + \min\{|h|, |c|\}$ coordinates in x in every case so that for the perturbed instance x' it holds $x' \in \mathcal{E}(h, c)$. $\square$

3.2 Prediction Change Risk and Robustness

Theorem 3.5 (Prediction change risk). *Consider the Problem Setup 1. Then,* $\mathrm{Risk}_r^{\mathrm{PC}}(h) = 0$ *when* $r = 0$, *while when* $r \geq 1$ *we have,*

$$
\mathrm{Risk}_r^{\mathrm{PC}}(h) = 2^{-|h|} \cdot \sum_{i=0}^{r} \binom{|h|}{i}.
$$

Proof. With a budget of $r = 0$ we can not change the evaluation of h at any instance, so the prediction change adversarial risk is 0.

Now consider the case where we have a budget of $1 \leq r < |h|$. A satisfying truth assignment x (which arises with probability $2^{-|h|}$) requires a budget of only $r = 1$ in order to become falsifying. For a falsifying truth assignment x that has i 0's among the $|h|$ variables that appear in h we need a budget of i. Therefore, for a budget of $1 \leq r < |h|$ we can violate $\sum_{i=1}^{r} \binom{|h|}{i} 2^{n-|h|}$ truth assignments that were originally falsifying.

Of course if we have a budget of $r \geq |h|$, then we can change any x to an x' such that $h(x) \neq h(x')$ and therefore the overall risk is going to be 1. In other words, for the interesting cases where $1 \leq r < |h|$ we have $\mathrm{Risk}_r^{\mathrm{PC}}(h) = 2^{-|h|} + 2^{-n} \cdot \sum_{i=1}^{r} \binom{|h|}{i} 2^{n-|h|} = 2^{-|h|} \sum_{i=0}^{r} \binom{|h|}{i}$. This last formula is also consistent with budget $r \geq |h|$ as then $\mathrm{Risk}_r^{\mathrm{PC}}(h) = 1$. $\square$

Theorem 3.6 (Prediction change robustness). *Consider the Problem Setup 1. Then,*

$$
\mathrm{Rob}^{\mathrm{PC}}(h) = \frac{1}{2} \cdot |h| + 2^{-|h|}.
$$

12

Proof. When x is a satisfying truth assignment for h then it suffices to turn a single 1 into a 0 for one of the variables that appear in h so that the resulting x' is such so that $h(x) \neq h(x')$. When x is a falsifying truth assignment for h with i out of the $|h|$ variables that appear in h being 1, then we need to flip $\sigma = |h| - i$ bits in x for the prediction to change. Therefore, $\text{Rob}^{\text{PC}}(h) = 2^{-n} \cdot 1 \cdot 2^{n-|h|} + 2^{-n} \cdot \sum_{i=0}^{|h|-1} (|h|-i) \cdot \binom{|h|}{i} \cdot 2^{n-|h|} = 2^{-|h|} + 2^{-|h|} \cdot \sum_{j=1}^{|h|} j \cdot \binom{|h|}{j} = 2^{-|h|} + |h|/2$. $\qquad\square$

3.3 Corrupted Instance Risk and Robustness

In the theorem below the important message is that driving the adversarial risk to 1 based on the corrupted instance definition requires budget $\Theta(|h|)$ contrasting the budget of size $\Theta(\min\{|h|, |c|\})$ that is required in Theorem 3.3 in order to drive the adversarial risk to 1 based on the error region definition. The other cases in the statement below refer to intermediate values of budget and explain how we arrive at this conclusion.

Theorem 3.7 (Corrupted instance risk). *Consider the Problem Setup 1. Then, for a budget of size r we have,*

- *if $r = 0$, then* $\text{Risk}_r^{\text{CI}}(h, c) = \mu = (2^w + 2^u - 2) \cdot 2^{-m-u-w}$,

- *if $1 \leq r < w$, then* $\text{Risk}_r^{\text{CI}}(h, c) = 2^{-|c|} + 2^{-|h|} \left(\sum_{j=0}^{r} \binom{|h|}{j} - 2^{-u} \sum_{\xi=0}^{r} \binom{w}{\xi} \right)$,

- *if $w \leq r < |h|$, then* $\text{Risk}_r^{\text{CI}}(h, c) = 2^{-|h|} \cdot \sum_{j=0}^{r} \binom{|h|}{j}$,

- *if $|h| \leq r$, then* $\text{Risk}_r^{\text{CI}}(h, c) = 1$.

Proof. When $x \in \mathcal{E}(h, c)$, all these instances contribute to the adversarial risk, for any budget value (including $r = 0$), since they satisfy $h(x) = h(x') \neq c(x)$. The probability of obtaining such an x is, by (2), $(2^w + 2^u - 2) \cdot 2^{-m-u-w}$.

When x is a satisfying truth assignment for both h and c, then we need to flip $\sigma = 1$ bits. Therefore with a budget of 1, when either h or c (or both) are satisfied, then these truth assignments contribute to the risk an amount of $2^{-m-u} + 2^{-m-w} - 2^{-m-u-w}$.

When x is a falsifying truth assignment for both h and c we have the following two cases.

- When x falsifies $1 \leq j \leq m$ mutual and $0 \leq \xi \leq w$ wrong variables, we need to perturb $\sigma = j + \xi$ bits so that $h(x') \neq c(x)$. Therefore with a budget of r we can increase the risk by an amount of

$$2^{-m-w} \sum_{j=1}^{r} \left(\binom{m+w}{j} - \binom{w}{j} \right).$$

- When no mutual variable is falsified (or there are no mutual variables between h and c), but x falsifies $1 \leq \zeta \leq u$ undiscovered and $1 \leq \xi \leq w$ wrong variables, then we need to perturb $\sigma = \xi$ bits. Hence, with a budget of r we can increase the risk by an amount of $2^{-m-u-w}(2^u - 1) \sum_{\xi=1}^{r} \binom{w}{\xi}$.

We now add everything up. $\qquad\square$

Theorem 3.8 (Corrupted instance robustness). *Consider the Problem Setup 1. Then,*

$$\frac{1}{4} \cdot |h| < \text{Rob}^{\text{CI}}(h, c) < |h| + \frac{1}{2}.$$

Proof. First note that instances in the error region contribute precisely 0 to the overall robustness, regardless if we are looking for a lower bound or an upper bound on the robustness.

For the lower bound consider truth assignments that are falsifying both h and c such that the following variables are falsified: $1 \leq j \leq m$ mutual variables, $0 \leq \zeta \leq u$ undiscovered variables, and $0 \leq \xi \leq w$ wrong variables. So that we can achieve $h(x') \neq c(x)$, we need an x' such that $h(x') = 1$. But then this means that we need to flip $\sigma = j + \xi$ bits in x in this case. As a result, the contribution of these truth assignments to the overall robustness of h is,

$$
\begin{aligned}
Q_1 &= 2^{-m-w-u} \sum_{j=1}^{m} \binom{m}{j} \sum_{\zeta=0}^{u} \binom{u}{\zeta} \sum_{\xi=0}^{w} \binom{w}{\xi} \cdot (j+\xi) \\
&= \frac{m}{2} + \frac{w}{2} \cdot (1 - 2^{-m}) \geq \frac{m}{2} + \frac{w}{4} > \frac{m+w}{4} = \frac{|h|}{4}.
\end{aligned}
$$

For the upper bound, for a truth assignment x that is falsifying both h and c we need to change no more than $|h|$ bits in x so that for the perturbed instance x' it holds $h(x') = 1 \neq 0 = h(x) = c(x)$. As the probability of obtaining such a truth assignment x is strictly less than 1, then the contribution to the overall robustness due to such truth assignments is strictly less than $|h|$. On the other hand, for satisfying truth assignments of both h and c we have $\sigma = 1$ bits that need to change in x and therefore the contribution to the overall robustness is $2^{-m-u-w} \leq 1/2$. $\square$

3.4 Experiments for the Expected Values of Adversarial Robustness

In this part, we complement the theorems that we presented earlier with experiments. This way we are able to examine how some popular algorithms behave under attack, and we explore the extent to which the generated solutions of such algorithms exhibit differences in their (adversarial) robustness on average against various target functions drawn from the class of monotone conjunctions.

Overview of algorithms. The first algorithm is the standard Occam algorithm that starts from the full conjunction and eliminates variables from the hypothesis that contradict the positive examples received; this algorithm is known as FIND-S in [Mit97] because it maintains the most *specific* hypothesis in the *version space* [Mit77] induced by the training data, but has appeared without a name earlier by Valiant in [Val84] and its roots are at least as old as in [BGA57]. The second algorithm is the SWAPPING ALGORITHM from the framework of evolvability [Val09]. This algorithm searches for an ε-optimal solution among monotone conjunctions that have at most $\lceil \lg(3/(2\varepsilon)) \rceil$ variables in their representation using a local search method where hypotheses in the neighborhood are obtained by swapping in and out some variable(s) from the current hypothesis; in particular we follow the analysis that was used in [DT09] and is a special case of the analysis used in [Dio16].

Experimental setup. In each experiment, we first learn hypotheses by using the algorithms under U_n against different target sizes. For both algorithms, during the learning process, we use $\varepsilon = 0.01$ and $\delta = 0.05$ for the learning parameters. We then examine the robustness of the generated hypotheses by drawing examples again from the uniform distribution U_n as this is the main theme of this paper. In particular, we test against the 30 target sizes from the set $\{1, 2, \ldots, 24, 25, 30, 50, 75, 99, 100\}$. For each such target size, we plot the average value, over 500 runs, of the robustness of the learned hypothesis that we obtain. In each run, we repeat the learning process using a random target of the particular size as well as a fresh training sample

and subsequently estimate the robustness of the learned hypothesis by drawing another $10,000$ examples from U_n that we violate (depending on the definition). The dimension of the instances is $n = 100$.

3.4.1 The Algorithm FIND-S

Algorithm FIND-S is shown in Algorithm 1. The initial hypothesis is the full monotone conjunction and as positive examples are received during training, the learner drops from the hypothesis those variables that are falsified in the positive examples. Following the Problem Setup 1, the concept class $\mathcal{C}_n$ contains $|\mathcal{C}_n| = 2^n - 1$ monotone conjunctions since the empty conjunction is not part of the class. Similarly, the hypothesis class is $\mathcal{H} = \mathcal{C}_n$. So, in particular for our experiments we have $|\mathcal{H}| = 2^{100} - 1$ since $n = 100$. Therefore, we use a sample of size $m = \lceil \frac{1}{\varepsilon} \ln(|\mathcal{H}|/\delta) \rceil = 7232$ which is enough (see, e.g., [BEHW87]) for learning any finite concept class of size $|\mathcal{H}|$ up to risk ε, with probability at least $1 - \delta$, under any distribution and in particular under the uniform distribution U_n as in our experiments.

Algorithm 1: An Occam algorithm for efficiently PAC learning monotone conjunctions

Input: dimension $n \in \mathbb{N}^*$, $\delta \in (0,1)$, $\varepsilon \in (0,1)$
Output: a hypothesis $h \in \mathcal{C}_n$

1   $h \leftarrow \wedge_{i=1}^{n} x_i$;
2   Draw $m = \lceil \frac{1}{\varepsilon} \cdot \ln(|\mathcal{H}|/\delta) \rceil$ examples;
3   **for** $j \leftarrow 1$ *up to* m **do**
4   **if** *j-th example is positive* **then** remove from h all the variables that are falsified in the example ;
5   **return** h;

As we proved in Theorems 3.6 and 3.8 the robustness of the learned hypothesis is *proportional to the size of the hypothesis* regardless of the size of the target when using the definitions of *prediction change* and *corrupted instance* (and in fact, experimentally the robustness in both cases is about $|h|/2$). However, the robustness of the hypothesis using the *error region* approach is *proportional to the minimum between the size of the hypothesis and the size of the target*. Now note that the way FIND-S constructs solutions, it holds that $|c| = \min\{|h|, |c|\}$. As such, when the target is sufficiently large (e.g., $|c| \geq 20$), then the learned hypothesis using FIND-S is almost always the initial full conjunction of size 100 and therefore the robustness based on the *error region* approach is proportional to $|c|$ and not proportional to 100 (which is the size of h) as it is the case of the robustness that is based on the *prediction change* and *corrupted instance* definitions. In other words, the robustness obtained by the *error region* approach is significantly different from the robustness obtained by the *prediction change* and *corrupted instance* approaches for a significant range of target functions. Figure 2 presents the values of the three robustness measures when using the FIND-S algorithm to form a hypothesis in our setup ($\varepsilon = 0.01$, etc.). Then, Figure 3 presents the average size of the learned hypotheses and Figure 4 presents the average error of the hypotheses learned.

3.4.2 The SWAPPING ALGORITHM

The SWAPPING ALGORITHM appeared in Valiant's framework of evolvability [Val09]. We give a brief description of this algorithm without going deeply in our discussions regarding fascinating details of the framework of evolvability. Loosely speaking, in evolvability the idea is to develop an evolutionary mechanism that allows near-optimal hypotheses to form by letting such hypotheses interact with the environment. Here we describe the evolutionary mechanism as if it is deployed by some learner who knows how the evolutionary mechanism operates on hypotheses (organisms), how these hypotheses interact with the environment, and how this interaction affects the hypothesis formation from one iteration (generation) to another.
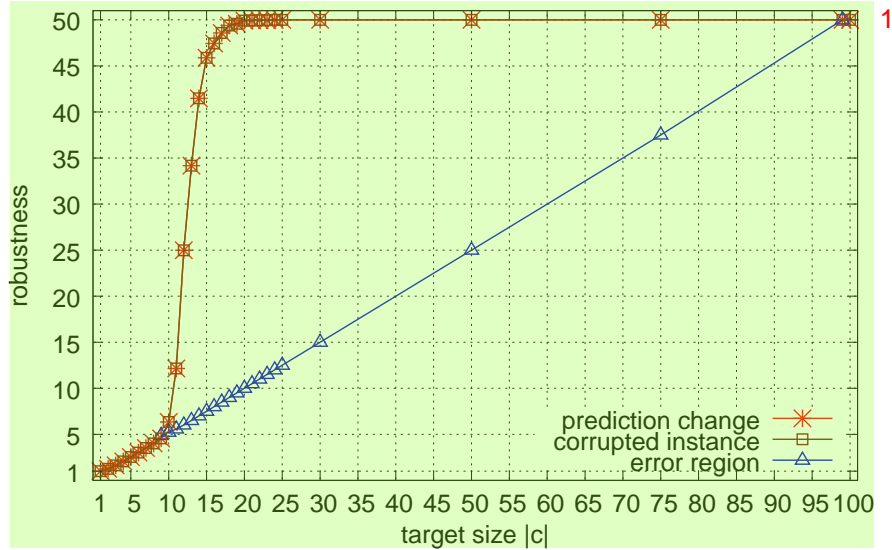
Figure 2: Experimental comparison of the different robustness measures. The values for PC and CI almost coincide and they can hardly be distinguished. The value for ER robustness is completely different compared to the other two. Note that ER robustness is ∞ when the target size $|c|$ is in $\{1,\ldots,8\} \cup \{100\}$ and for this reason only the points between 9 and 99 are plotted. When $|c| \geq 20$, almost always the learned hypothesis is the initialized full conjunction. The reason is that positive examples are very rare and our training set contains none. As a result no variable is eliminated from the initialized hypothesis h (full conjunction). Hence, when $|c| \geq 20$ we see that PC and CI robustness is about $\max\{|h|,|c|\}/2 = |h|/2$, whereas ER is roughly $\min\{|h|,|c|\}/2 = |c|/2$.

First of all, the algorithm is a *local-search* method. At any given time the algorithm is maintaining a hypothesis that is a monotone conjunction of some of the Boolean variables. For any given hypothesis that the learner has formed, a *mutator* function (i) defines a *neighborhood* $N(h)$[4] with hypotheses that can potentially replace the current hypothesis of the learner, (ii) *scores* the candidates in the neighborhood based on their *predictive performance* on samples of certain size, and (iii) uses a *decision rule* to determine which hypothesis among the ones in the neighborhood will be selected as the hypothesis that the learner will have during the next iteration (generation).

Neighborhood. The neighborhood N, can, in general, be decomposed into $N = N^+ \cup N^- \cup N^\pm \cup \{h\}$. The neighborhood N^+ contains the hypotheses that have one more variable compared to the current hypothesis h, the neighborhood N^- contains the hypotheses that have one less variable compared to the current hypothesis h, and the neighborhood $N^\pm$ contains the hypotheses that are obtained by *swapping* one of the variables that appear in h with one of the variables that do not yet appear in h. As an example, let $h = x_1 \wedge x_2$ and $n = 3$. Then, $N^- = \{x_1, x_2\}$, $N^+ = \{x_1 \wedge x_2 \wedge x_3\}$, and $N^\pm = \{x_1 \wedge x_3, x_2 \wedge x_3\}$. As we can see, the current hypothesis h is always in the neighborhood, so the learner can always retain its current guess for the next iteration.[5]

---

[4]We will simply write N instead of $N(h)$, since the neighborhood always depends on the current hypothesis h.

[5]Additions and deletions can be seen as swaps with the constant 1. This swapping nature of the algorithm justifies its name that was given in [DT09].
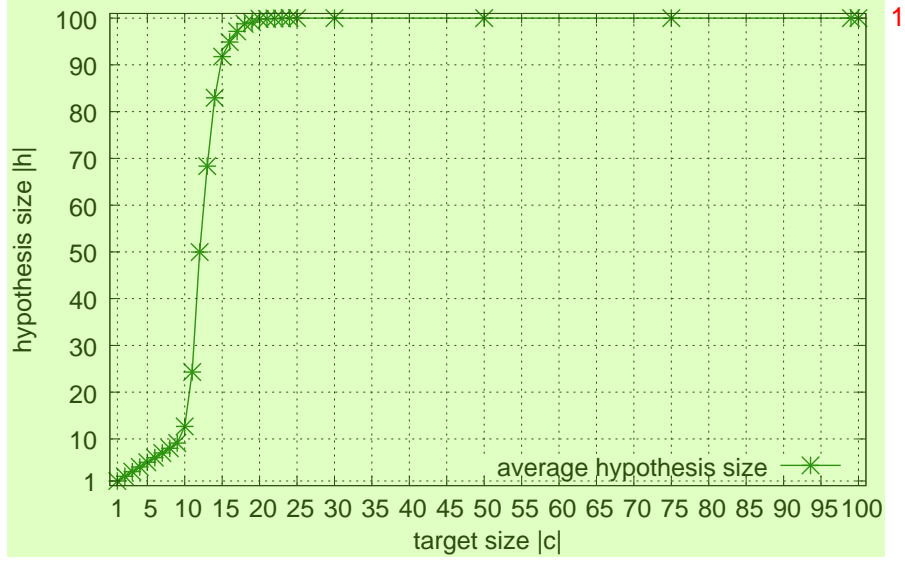
16

Figure 3: Average size, over 500 runs, for the computed hypothesis using FIND-S for learning monotone conjunctions with a sample of size $7,232$ under the uniform distribution U_n.

Weights. Each hypothesis in $N^+ \cup N^- \cup \{h\}$ is assigned the same weight so that all these weights add up to $1/2$. Each hypothesis in $N^{\pm}$ is assigned the same weight so that the weights of all these hypotheses in $N^{\pm}$ also add up to $1/2$.

Scoring. The predictive power of each hypothesis in N is approximated, with high probability, within ϵ_s of its true value, by testing its *predictive performance* on a sufficiently large sample.

Partitioning. The hypotheses in N are partitioned into three sets based on their *predictive performance*. For a real constant t, called *tolerance*, and a base performance value ν_h for the current hypothesis h, the hypotheses that exhibit performance strictly larger than $\nu_h + t$ form the *beneficial* group $Bene$, the hypotheses that exhibit performance within $\nu_h \pm t$ form the *neutral* set $Neut$, and finally the hypotheses that exhibit performance strictly less than $\nu_h - t$ form a *deleterious* set.

Decision rule for mutator. If the set $Bene$ is non-empty, then a hypothesis is selected from this set, otherwise a hypothesis from the set $Neut$ is selected. Note that $Neut$ is always nonempty as the current hypothesis is always there. When a hypothesis has to be selected among many from a set (whether $Bene$ or $Neut$), it is selected with probability proportional to its weight compared to the total weight of the hypotheses in the set.

Hypothesis class $\mathcal{H}$. For a threshold $q = \lceil \log_2(3/(2\varepsilon)) \rceil$ it can be shown [DT09] that, for any target monotone conjunction from our concept class $\mathcal{C}_n$ in the Problem Setup 1, among the monotone conjunctions that have up to q variables, there is at least one such conjunction that has risk at most ε. In particular, the SWAPPING ALGORITHM will (i) identify precisely targets with up to q variables (i.e., generate a hypothesis that achieves risk 0), or (ii) form a hypothesis with q variables that has risk at most ε for targets that have size larger than q. Due to this property of the algorithm, a typical hypothesis class is $\mathcal{H} = \mathcal{C}_n^{\leq q}$ that contains
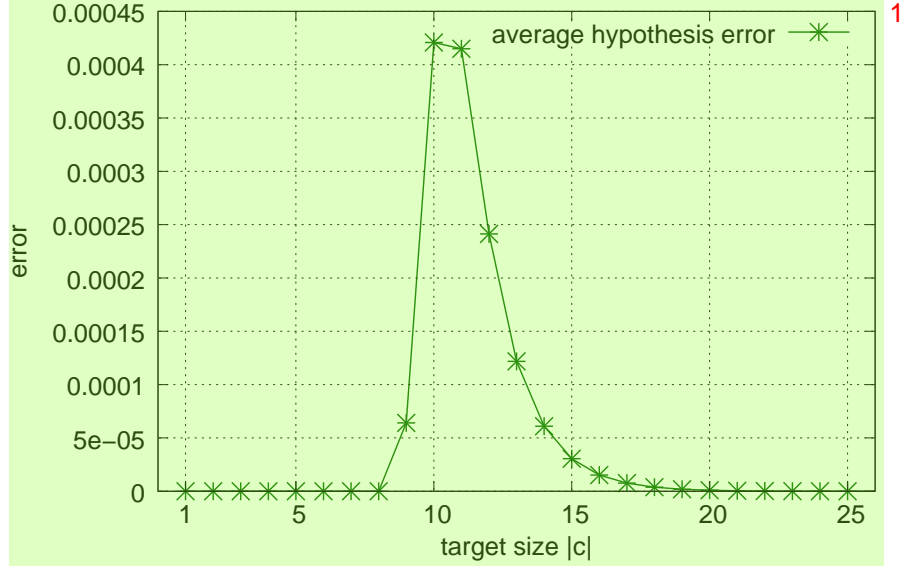
Figure 4: Average error, over 500 runs, of the computed hypotheses using FIND-S for learning monotone conjunctions with a sample of size $7,243$ under the uniform distribution U_n. For target sizes $|c| \in \{1, \ldots, 8\} \cup \{100\}$, the learner always identifies precisely the target and thus the error is precisely 0 in that regime. (For the case $|c| = 100$ the learner is lucky, as the initial guess is the target itself and thus the hypothesis never changes as examples are being presented to the learner.) For target sizes $|c|$ between 14 and 99 the average error is nonzero but less than 10^{-4} and decays as the size of the target increases. For clarity we only plot target sizes between 1 and 25.

the monotone conjunctions that have 0 up to q variables. Therefore, we formally deviate from the hypothesis class $\mathcal{H} = \mathcal{C}_n$ found in our Problem Setup 1 - even though the algorithm can still run with this hypothesis class and moreover in the end the algorithm will return a hypothesis that satisfies $h \in \mathcal{C}_n$ as in the Problem Setup 1. This deviation on the hypothesis class is done on one hand for simplicity because it is easier to explain the evolutionary mechanism, and on the other hand because such a selection of a hypothesis class technically gives an algorithm in the *non-realizable* case of learning - which might be interesting in its own right. For more information on these details we refer the reader to [DT09] and [Dio16].

The mutator function for the SWAPPING ALGORITHM is shown in Algorithm 2. The function SetWeight assigns weights to the hypotheses as explained in the paragraph about assigning weights above. The function Select returns a hypothesis from the set in its argument as explained in the paragraph above regarding the decision rule for the mutator. The function Perf computes empirically the predictive performance of the hypotheses within accuracy ϵ_s with probability at least $1 - \delta_s$.

As mentioned above, for the solutions obtained by the SWAPPING ALGORITHM it holds that $|h| = \min\{|h|, |c|\}$. Thus, while we can observe some differences for a small range of target functions regarding the robustness of the generated hypotheses based on the various definitions, these differences are smaller compared to what we obtain with the FIND-S algorithm. Figure 5 presents the values of the three robustness measures when using the swapping algorithm from the framework of evolvability in order to form a hypothesis. The robustness measures based on the *prediction change* and *corrupted instance* definitions behave very similarly against the solutions obtained by the SWAPPING ALGORITHM, just like they did for the solutions obtained by the FIND-S algorithm.

18

Algorithm 2: Mutator function for uniform distribution

Input: dimension n, $\delta \in (0,1)$, $\varepsilon \in (0,1)$, $h \in \mathcal{H}$
Output: a new hypothesis h'

1   $q \leftarrow \lceil \log_2(3/(2\varepsilon)) \rceil$;
2   **if** $|h| > 0$ **then** Generate N^- **else** $N^- \leftarrow \emptyset$;
3   **if** $|h| < q$ **then** Generate N^+ **else** $N^+ \leftarrow \emptyset$;
4   **if** $|h| \le q$ **then** Generate $N^\pm$ **else** $N^\pm \leftarrow \emptyset$;
5   $Bene \leftarrow \emptyset$; $Neutral \leftarrow \{h\}$;
6   $t \leftarrow 2^{-2q}$; $\epsilon_s \leftarrow 2^{-2q}$; $\delta_s \leftarrow \delta/(6q^2n)$;
7   $\texttt{SetWeight}(h, h, N^-, N^+, N^\pm)$; $\nu_h \leftarrow \texttt{Perf}(h, \epsilon_s, \delta_s)$;
8   **for** $x \in N^+, N^-, N^\pm$ **do**
9     $\texttt{SetWeight}(x, h, N^-, N^+, N^\pm)$; $\nu_x \leftarrow \texttt{Perf}(x, \epsilon_s, \delta_s)$;
10   **if** $\nu_x > \nu_h + t$ **then** $Bene \leftarrow Bene \cup \{x\}$;
11   **else if** $\nu_x \ge \nu_h - t$ **then** $Neutral \leftarrow Neutral \cup \{x\}$;
12   **if** $Bene \ne \emptyset$ **then return** $\texttt{Select}(Bene)$ **else return** $\texttt{Select}(Neutral)$;
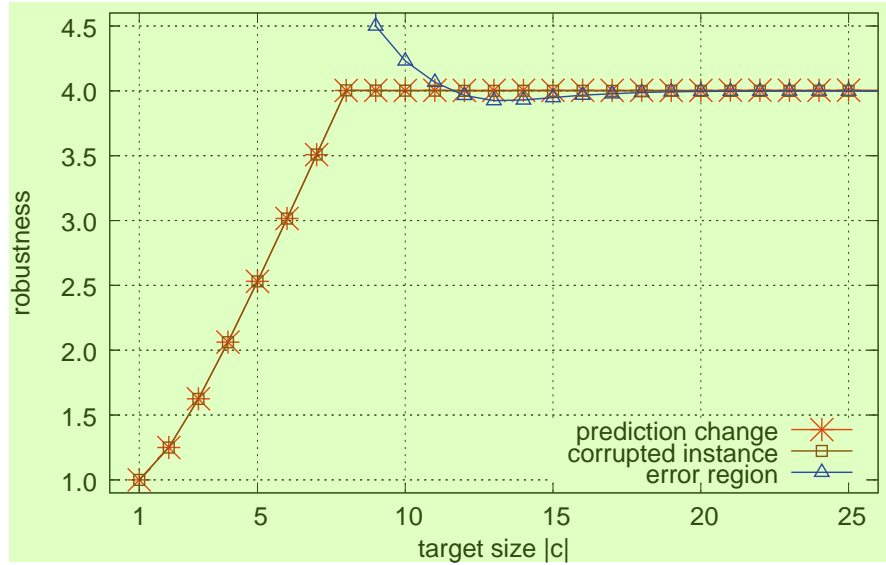

Figure 5: Experimental comparison of the different robustness measures regarding the hypotheses that are obtained by the SWAPPING ALGORITHM. The values for all three measures almost coincide when the target has size $|c| \ge 20$ and for clarity we only plot the values until $|c| = 25$. Note that ER robustness is ∞ when the target size $|c|$ is in $\{1, \ldots, 8\} \cup \{100\}$ and for this reason these points are not plotted (similarly to Figure 2). Further, when $9 \le |c| \le 19$, we can observe that ER robustness behaves slightly differently compared to PC and CI and eventually in the regime $20 \le |c| \le 99$ it is much closer to CI with PC being consistently slightly larger than both of them (and this is explained by the fact that truth assignments that belong to the disagreement region still need to be perturbed). Finally, our earlier bounds predict that the values will be roughly the same as this time $\min\{|h|, |c|\} = |h|$ contrasting the observation from Figure 2.

We see that the robustness due to the *error region* is different for some targets of certain size compared to the robustness values obtained when using the *prediction change* and the *corrupted instance* definitions, but this time the differences are smaller compared to what we observe for the solutions obtained by the FIND-S algorithm (and of course excluding the cases where the error region robustness is infinite). However this

is more or less expected since, on one hand for the learned solution it is always true that $\min\{|h|, |c|\}$ coincides with $|h|$ which is the factor governing the robustness under the *prediction change* and *corrupted instance* definitions, regardless of the size of the target $|c|$, and on the other hand, again due to the fact that $|h| = \min\{|h|, |c|\}$, the robustness of the generated solutions have significantly smaller values compared to the robustness of the generated solutions obtained by FIND-S.

4   Inherent Bounds on Risk and Robustness for the Uniform Distribution

After showing the different behavior of various definitions of adversarial risk and robustness through a study of monotone conjunctions under the uniform distribution, in this section, we state and prove our main theorems about *inherent barriers* against achieving *error region* adversarial risk and robustness of *arbitrary* learning problems whose instances are distributed uniformly over the n-dimension hypercube $\{0,1\}^n$.

   We first define a useful notation for the size of the (partial) Hamming balls and state three lemmas and two corollaries based on the notation.

Definition 4.1. *For every* $n \in \mathbb{N}$ *we define the (partial) "Hamming Ball Size" function* $\mathsf{BSize}_n \colon [n] \times [0,1) \to [0,1)$ *as follows*

$$\mathsf{BSize}_n(k, \lambda) = 2^{-n} \cdot \left(\sum_{i=0}^{k-1} \binom{n}{i} + \lambda \cdot \binom{n}{k} \right).$$

Note that this function is a bijection and we use $\mathsf{BSize}^{-1}(\cdot)$ *to denote its inverse. When* n *is clear from the context, we will simply use* $\mathsf{BSize}(\cdot, \cdot)$ *and* $\mathsf{BSize}^{-1}(\cdot)$ *instead.*

Lemma 4.2. *For* $\mu \in [0,1]$ *we have* $\mu \geq \mathsf{BSize}\left(\frac{n - \sqrt{-2 \cdot \ln(\mu) \cdot n}}{2} + 1, 0 \right)$. *Also, if* $(k, \lambda) = \mathsf{BSize}^{-1}(\mu)$ *then*

$$k \geq \frac{n - \sqrt{-2 \cdot \ln(\mu) \cdot n}}{2} + 1.$$

Proof. Let $k' = \frac{n - \sqrt{-2 \cdot \ln(\mu) \cdot n}}{2} + 1$. Now consider n uniform random variables $X_1, \ldots, X_n$ over $\{0,1\}$. We have

$$\mathsf{BSize}(k', 0) = \Pr[X_1 + \cdots + X_n \leq k' - 1].$$

Then, by Hoefding inequality we have

$$\Pr[X_1 + \cdots + X_n \leq k - 1] \leq e^{-n \cdot (1 - \frac{2k-2}{n})^2 / 2} = \mu.$$

Therefore we have,

$$\mu \geq \mathsf{BSize}(k', 0)$$

which proves the first part of the Lemma. Also the second part of the lemma immidiately follows because we have

$$\mu = \mathsf{BSize}(k, \lambda) \geq \mathsf{BSize}(k', 0)$$

which implies $k \geq k'$. □

Lemma 4.3 (Followed by the Central Limit Theorem [Bil08])**.** *For all $\lambda \in [0,1)$ and $a \in \mathbb{R}$ we have*

$$\lim_{n \to \infty} |\mathsf{BSize}(n/2 + a \cdot \sqrt{n}, \lambda)| = \Phi(2a)$$

where Φ is the CDF of the standard normall distribution.

Lemma 4.4 ([LPV03])**.** *If $1 \le k \le \lfloor n/2 \rfloor$, then $\mathsf{BSize}(k,0) < \binom{n}{k} \cdot 2^{2m-1-n}/\binom{2m}{m}$ where $m = \lfloor \frac{n}{2} \rfloor$.*

Proof. Case of $n = 2m$ is Lemma 3.8.2 in [LPV03]. Case of $n = 2m+1$ follows from the case for $n = 2m$ and Pascal's equality. $\qquad\square$

    Using the above lemma, we can give a worst-case and a "limit-case" bound for the Hamming ball.

Corollary 4.5. *If $1 \le k \le \lfloor n/2 \rfloor$, then $\mathsf{BSize}(k,0) < \binom{n}{k} \cdot \sqrt{\frac{n}{2^{2n+1}}}$.*

Proof. By Lemma A.2 we know that $\binom{2m}{m} \ge \frac{2^{2m-1}}{\sqrt{m}}$. Therefore, by Lemma 4.4 we have $\mathsf{BSize}(k,0) < \binom{n}{k} \cdot \sqrt{\frac{n}{2^{2n+1}}}$. $\qquad\square$

Corollary 4.6. *For any $k \in \mathbb{N}$, $\lim_{n \to \infty} \frac{\mathsf{BSize}(k,0)}{\sqrt{n} \cdot \binom{n}{k} \cdot 2^{-n}} \le \sqrt{\frac{\pi}{8}}$.*

Proof. Let $m = \lfloor \frac{n}{2} \rfloor$. By Lemma 4.4 we have

$$\frac{\mathsf{BSize}(k,0)}{\sqrt{n} \cdot \binom{n}{k} \cdot 2^{-n}} \le \frac{2^{2m-1}}{\sqrt{2m}\binom{2m}{m}}$$

which implies $\lim_{n \to \infty} \frac{\mathsf{BSize}(k,0)}{\sqrt{n} \cdot \binom{n}{k}} \le \lim_{m \to \infty} \frac{2^{2m-1}}{\sqrt{2m}\binom{2m}{m}} = \sqrt{\frac{\pi}{8}}$. Where the last equality follows from Lemma A.3. $\qquad\square$

    The following theorem, gives a general lower bound for the adversarial risk of any classification problem for uniform distribution U_n over the hypercube $\{0,1\}^n$, depending on the original error.

Theorem 4.7. *Suppose $\mathsf{P} = (\{0,1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \mathsf{HD})$ is a classification problem. For any $h \in \mathcal{H}, c \in \mathcal{C}$ and $r \in \mathbb{N}$, let $\mu = \mathsf{Risk}(h,c) > 0$ be the original risk and $(k,\lambda) = \mathsf{BSize}^{-1}(\mu)$ be a function of the original risk. Then, the error-region adversarial risk under r-perturbation is at least*

$$\mathsf{Risk}_r^{\mathrm{ER}}(h,c) \ge \mathsf{BSize}(k+r, \lambda).$$

    Before proving Theorem 4.7 we state and prove two corollaries. The proof of Theorem 4.7 appears in section 4.1.

Corollary 4.8 (Error-region risk for all n)**.** *Suppose $\mathsf{P} = (\{0,1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \mathsf{HD})$ is a classification problem. For any hypothesis h, c with risk $\mu \in (0, \frac{1}{2}]$ in predicting a concept function c, we can increase the risk of (h,c) from $\mu \in (0, \frac{1}{2}]$ to $\mu' \in [\frac{1}{2}, 1]$ by changing at most*

$$r = \sqrt{\frac{-n \cdot \ln \mu}{2}} + \sqrt{\frac{-n \cdot \ln(1 - \mu')}{2}}$$

bits in the input instances. Namely, by using the above r, we have $\mathsf{Risk}_r^{\mathrm{ER}}(h,c) \ge \mu'$. Also, to increase the error to $\frac{1}{2}$ we only need to change at most $r' = \sqrt{\frac{-n \cdot \ln(\mu)}{2}}$ bits.

Proof. Let $(k, \lambda) = \mathsf{BSize}^{-1}(\mu)$. By Theorem 4.7, we know that

$$\mathsf{Risk}_r^{\mathrm{ER}}(h, c) \geq \mathsf{BSize}(k + r, \lambda).$$

By Lemma 4.2 we know that $k \geq \frac{n - \sqrt{-2 \cdot \ln(\mu) \cdot n}}{2}$. Therefore we have

$$
\begin{aligned}
\mathsf{Risk}_r^{\mathrm{ER}}(h, c) &\geq \mathsf{BSize}(k + r, \lambda) \\
&\geq \mathsf{BSize}\left(\frac{n + \sqrt{-2 \cdot \ln(1 - \mu') \cdot n}}{2}, \lambda\right) \\
&\geq 1 - \mathsf{BSize}\left(\frac{n - \sqrt{-2 \cdot \ln(1 - \mu') \cdot n}}{2} + 1, 0\right) \\
(\text{By Lemma 4.2}) \quad &\geq \mu'
\end{aligned}
$$

$\square$

Similarly, for the case of reaching error $\frac{1}{2}$ we have

$$\mathsf{Risk}_{r'}^{\mathrm{ER}}(h, c) \geq \mathsf{BSize}(k + r', \lambda) \geq \mathsf{BSize}\left(\frac{n}{2}, \lambda\right) \geq \frac{1}{2}.$$

Example. Corollary 4.8 implies that for classification tasks over U_n, by changing at most $3.04\sqrt{n}$ number of bits in each example we can increase the error of an hypothesis from 1% to 99%. Furthermore, for increasing the error just to 0.5 we need half of the number of bits, which is $1.52\sqrt{n}$.

Also, the corollary bellow, gives a lower bound on the limit of adversarial risk when $n \mapsto \infty$. This lower bound matches the bound we have in our computational experiments.

Corollary 4.9 (Error-region risk for large n). *Let and* $\mathsf{P} = (\{0, 1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \mathsf{HD})$ *be a classification problem nd* $\mu \in (0, 1]$ *and* $\mu' \in (\mu, 1]$. *Then for any* $h \in \mathcal{H}, c \in \mathcal{C}$ *such that* $\mathsf{Risk}(h, c) \geq \mu$ *we have* $\mathsf{Risk}_r(h, c) \geq \mu'$ *for*

$$r \approx \sqrt{n} \cdot \frac{\Phi^{-1}(\mu') - \Phi^{-1}(\mu)}{2} \text{ when } n \mapsto \infty$$

where Φ is the CDF of the standard normal distribution.

Proof. For simplicity suppose μ is exactly the risk (rather than a lower bound for it). Let $(k, \lambda) = \mathsf{BSize}^{-1}(\mathsf{Risk}(h, c))$. By Lemma 4.3, for $n \mapsto \infty$ we have

$$\mu = |\mathsf{BSize}(k, \lambda)| \approx \Phi\left(\frac{2k - n}{\sqrt{n}}\right).$$

Therefore, $k \approx n/2 + \Phi^{-1}(\mu) \cdot \sqrt{n}/2$. By Theorem 4.7 and another application of Lemma 4.3,

$$
\begin{aligned}
\mathsf{Risk}_r(h, c) &\geq |\mathsf{BSize}(k + r, 0)| \\
&\approx \left|\mathsf{BSize}\left(n/2 + \frac{\Phi^{-1}(\mu)}{2} \cdot \sqrt{n} - \frac{\Phi^{-1}(\mu)}{2} \cdot \sqrt{n} + \frac{\Phi^{-1}(\mu')}{2} \cdot \sqrt{n}, 0\right)\right| \\
&= \left|\mathsf{BSize}\left(n/2 + \frac{\Phi^{-1}(\mu')}{2} \cdot \sqrt{n}, 0\right)\right| \\
&\approx \Phi\left(\Phi^{-1}(\mu')\right) \\
&= \mu'
\end{aligned}
$$

$\square$

Example. Corollary 4.9 implies that for classification tasks over U_n, when n is large enough, we can increase the error from 1% to 99% by changing at most $2.34\sqrt{n}$ bits, and we can we can increase the error from 1% to 50% by changing at most $1.17\sqrt{n}$ bits in test instances.

The following theorem shows how to upper bound the adversarial *robustness* using the original risk.

Theorem 4.10. *Suppose* $\mathsf{P} = (\{0,1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \mathsf{HD})$ *is a classification problem. For any* $h \in \mathcal{H}$ *and* $c \in \mathcal{C}$, *if* $\mu = \mathsf{Risk}(h,c)$ *and* $(k,\lambda) = \mathsf{BSize}^{-1}(\mu)$ *depends on the original risk, then the error-region robustness is at most*

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq \sum_{r=0}^{n-k+1} (1 - \mathsf{BSize}(k+r,\lambda)).$$

Following, using Theorem 4.10, we give an asymptotic lower bound for robustness before proving the Theorem. The proof of Theorem 4.10 appears in section 4.1.

Corollary 4.11. *Suppose* $\mathsf{P} = (\{0,1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \mathsf{HD})$ *is a classification problem. For any hypothesis* h *with risk* $\mu \in (0, \frac{1}{2}]$, *we can make* h *to give always wrong answers by changing* $r = \sqrt{-n \cdot \ln \mu / 2} + \mu \cdot \sqrt{n/2}$ *number of bits on average. Namely, we have*

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq \sqrt{\frac{-n \cdot \ln \mu}{2}} + \mu \cdot \sqrt{\frac{n}{2}}.$$

And the following Corollary gives a lower bound on the robustness in limit.

Proof. Let $(k,\lambda) = \mathsf{BSize}^{-1}(\mu)$. By Theorem 4.10, we have

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq \sum_{r=0}^{n-k+1} 1 - \mathsf{BSize}(k+r,\lambda) \leq \sum_{r=0}^{n-k+1} 1 - \mathsf{BSize}(k+r,0).$$

On the other hand, by Lemma A.4 we know that $\sum_{i=1}^{n+1} \mathsf{BSize}(i,0) = 1 + \frac{n}{2}$. Therefore we have,

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq n - k + 1 - \left(1 + \frac{n}{2} - \sum_{i=1}^{k-1} \mathsf{BSize}(i,0)\right) = \frac{n}{2} - k + \sum_{i=0}^{k-1} \mathsf{BSize}(i,0).$$

Therefore we have,

$$
\begin{aligned}
\mathsf{Rob}^{\mathrm{ER}}(h,c) &\leq \frac{n}{2} - k + \sum_{i=1}^{k-1} \mathsf{BSize}(i,0) \\
(\text{By Lemma 4.5}) \quad &\leq \frac{n}{2} - k + \sum_{i=1}^{k-1} \binom{n}{i} \cdot 2^{-n} \cdot \sqrt{\frac{n}{2}} \\
&\leq \frac{n}{2} - k + \mathsf{BSize}(k,0) \cdot \sqrt{\frac{n}{2}} \\
&\leq \frac{n}{2} - k + \mu \cdot \sqrt{\frac{n}{2}}.
\end{aligned}
\tag{3}
$$

On the other hand, by using Lemma 4.2 we know that $k \geq \frac{n - \sqrt{-2 \cdot \ln(\mu) \cdot n}}{2} + 1$. Therefore, we have

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq \sqrt{\frac{-n \cdot \ln \mu}{2}} + \mu \cdot \sqrt{\frac{n}{2}}.$$

$\square$

Corollary 4.12. *For any $\mu \in (0,1]$, classification problem $\mathsf{P} = (\{0,1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \mathsf{HD})$, and any $h \in \mathcal{H}, c \in \mathcal{C}$ such that $\mathsf{Risk}(h,c) \geq \mu$, we have*

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq \frac{\Phi^{-1}(\mu)}{2} \cdot \sqrt{n} + \mu \cdot \sqrt{\frac{\pi \cdot n}{8}} \quad \text{when} \quad n \mapsto \infty,$$

where Φ is the CDF of the standard normal distribution.

Proof. Similar to inequality 3 in the proof of Corollary 4.11, for large enough n we have,

$$
\begin{aligned}
\mathsf{Rob}^{\mathrm{ER}}(h,c) &\leq \frac{n}{2} - k + \sum_{i=1}^{k-1} \mathsf{BSize}(i,0) \\
\text{(By Corollary 4.6)} \quad &\leq \frac{n}{2} - k + \sum_{i=1}^{k-1} \binom{n}{i} \cdot 2^{-n} \cdot \sqrt{\frac{\pi \cdot n}{8}} \\
&\leq \frac{n}{2} - k + \mathsf{BSize}(k,0) \cdot \sqrt{\frac{\pi \cdot n}{8}} \\
&\leq \frac{n}{2} - k + \mu \cdot \sqrt{\frac{\pi \cdot n}{8}}
\end{aligned}
$$

Now, by Lemma 4.3, for large enough n we have $k \approx n/2 + \Phi^{-1}(\mu) \cdot \sqrt{n}/2$. Therefore we have

$$\mathsf{Rob}^{\mathrm{ER}}(h,c) \leq \frac{\Phi^{-1}(\mu)}{2} \cdot \sqrt{n} + \mu \cdot \sqrt{\frac{\pi \cdot n}{8}}.$$

$\square$

Example. By changing $1.53\sqrt{n}$ number of bits *on average* we can increase the error of an hypothesis from 1% to 100%. Also, if $n \mapsto \infty$, by changing only $1.17\sqrt{n}$ number of bits *on average* we can increase the error from 1% to 100%.

4.1 Proving Theorems 4.7 and 4.10

We start by proving Theorem 4.7. Before that we define several notations.

Definition 4.13 (Volume, expansion, internal boundary and external boundary). *Let $\mathcal{A} \subseteq \{0,1\}^n$.*

- *The volume of $\mathcal{A}$ is simply its probability $\mathsf{vol}(\mathcal{A}) = |A|/2^n$.*
- *The r-expansion of $\mathcal{A}$ is $\mathcal{A}_r = \{x \in \{0,1\}^n \mid \exists a \in A, \mathsf{HD}(a,x) \leq r\}$*
- *The external boundary of $\mathcal{A}$ is defined as $\mathcal{EB}(\mathcal{A}) = \mathcal{A}_1 \setminus \mathcal{A}$.*
- *The internal boundary of $\mathcal{A}$ is $\mathcal{IB}(\mathcal{A}) = \{a \mid a \in \mathcal{A}, \exists x \in \{0,1\}^n \setminus \mathcal{A}, \mathsf{HD}(a,x) \leq 1\}$.*

The following isoperimetric inequality by Nigmatullin [Nig67] improves the famous vertex isoperimetric inequality of Harper [Har66] in a way that is more suitable for us to use.

Lemma 4.14 ([Nig67, Har66]). *For any set $A \subset \{0,1\}^n$ where $(k,\lambda) = \mathsf{BSize}^{-1}(\mathsf{vol}(A))$, we have*

$$|\mathcal{IB}(A)| \geq \binom{n}{k-1} + \lambda \cdot \left(\binom{n}{k} - \binom{n}{k-1} \right).$$

24

The following simple lemma shows how to turn lower bounds for internal boundary into a lower bound for external boundary.

Lemma 4.15 (External boundary vs. internal boundary). *Suppose* $b : [0,1] \to [0,1]$ *is an internal boundary lower bound. Namely, For any given set* $\mathcal{A} \subseteq \{0,1\}^n$ *we have,*

$$\mathsf{vol}(\mathcal{IB}(\mathcal{A})) \geq b(\mu(\mathcal{A})).$$

Then for any set $\mathcal{A} \subseteq \{0,1\}^n$ *we have,*

$$\mathsf{vol}(\mathcal{EB}(\mathcal{A})) \geq \inf\{b(\mu') \mid \mu' - b(\mu') \geq \mathsf{vol}(\mathcal{A})\}.$$

Proof. Suppose $\mathcal{A} \subseteq \{0,1\}^n$ is such that $\mathsf{vol}(\mathcal{A}) \geq \mu$, and let $\mathcal{A}' = \mathcal{A} \cup \mathcal{EB}(\mathcal{A})$. First we note that $\mathcal{IB}(\mathcal{A}') \subseteq \mathcal{EB}(\mathcal{A})$, because any point $a' \in \mathcal{IB}(\mathcal{A}')$ cannot be also in $\mathcal{A}$, as otherwise, a' would not be an internal boundary point in $\mathcal{A}'$. Now, let $\mu' = \mathsf{vol}(\mathcal{A}) = \mathsf{vol}(\mathcal{A}) + \mathsf{vol}(\mathcal{EB}(\mathcal{A}))$. Because $b(\cdot)$ is an internal boundary lower bound, and because $\mathcal{IB}(\mathcal{A}') \subseteq \mathcal{EB}(\mathcal{A})$, therefore $b(\mu') \leq \mathsf{vol}(\mathcal{EB}(\mathcal{A}))$. Thus, it holds that $\mu' - b(\mu') \geq \mu' - \mathsf{vol}(\mathcal{EB}(\mathcal{A})) = \mathsf{vol}(\mathcal{A}) \geq \mu$. By the definition of inf, it holds that $\inf\{b(\mu') \mid \mu' - b(\mu') \geq \mathsf{vol}(\mathcal{A})\} \leq b(\mu')$, which together with $b(\mu') \leq \mathsf{vol}(\mathcal{EB}(\mathcal{A}))$ implies that $\inf\{b(\mu') \mid \mu' - b(\mu') \geq \mathsf{vol}(\mathcal{A})\} \leq \mathsf{vol}(\mathcal{EB}(\mathcal{A}))$. $\square$

Now we use Lemma 4.15 to a derive a lower bound on the volume of exterior of a set.

Lemma 4.16. *For any set* $A \subset \{0,1\}^n$, *if* $(k,\lambda) = \mathsf{BSize}^{-1}(|A|/2^n)$, *then we have*

$$|\mathcal{EB}(A)| \geq \binom{n}{k} + \lambda \cdot \left(\binom{n}{k+1} - \binom{n}{k}\right).$$

Proof of Lemma 4.16. Using Lemma 4.14 and Lemma 4.15 we have

$$|\mathcal{EB}(A)| \geq \inf\left\{\binom{n}{k-1} + \lambda \cdot \left(\binom{n}{k} - \binom{n}{k-1}\right) \mid \mathsf{BSize}(k-1,\lambda) \geq |A|\right\}.$$

Since $\mathsf{BSize}(\cdot)$ is a monotone function,

$$|\mathcal{EB}(A)| \geq \binom{n}{k} + \lambda \cdot \left(\binom{n}{k+1} - \binom{n}{k}\right). \qquad \square$$

We now prove Theorem 4.7.

Proof of Theorem 4.7. Let set $\mathcal{E}$ be the error region of hypothesis h with respect to concept c. Namely, $\mathcal{E} = \{x \in \{0,1\}^n \mid h(x) \neq c(x)\}$. Let $(k_{\mathcal{E}}, \lambda_{\mathcal{E}}) = \mathsf{BSize}^{-1}(|\mathcal{E}|/2^n)$. By Lemma 4.16,

$$|\mathcal{EB}(\mathcal{E})| \geq \binom{n}{k_{\mathcal{E}}} + \lambda_{\mathcal{E}} \cdot \left(\binom{n}{k_{\mathcal{E}}+1} - \binom{n}{k_{\mathcal{E}}}\right).$$

Now, by induction on r, we prove that $\mathsf{vol}(\mathcal{E}_r) \geq \mathsf{BSize}(k_{\mathcal{E}} + r, \lambda_{\mathcal{E}})$. For $r = 0$ the statement is trivially true. Now for $r = i$, we assume $\mathsf{vol}(\mathcal{E}_i) \geq \mathsf{BSize}(k_{\mathcal{E}} + i, \lambda_{\mathcal{E}})$. For $r = i + 1$ we have,

$$\begin{aligned}
\mathsf{vol}(\mathcal{E}_{i+1}) &= \mathsf{vol}(\mathcal{Ball}_{i+1}(\mathcal{E})) \\
&= \mathsf{vol}(\mathcal{Ball}_i(\mathcal{E})) + \mathsf{vol}(\mathcal{EB}(\mathcal{Ball}_i(\mathcal{E}))) \\
\text{(By induction)} \quad &\geq \mathsf{BSize}(k_{\mathcal{E}} + i, \lambda_{\mathcal{E}}) + \mathsf{vol}(\mathcal{EB}(\mathcal{Ball}_i(\mathcal{E}))) \\
\text{(By Lemma 4.16)} \quad &\geq \mathsf{BSize}(k_{\mathcal{E}} + i, \lambda_{\mathcal{E}}) + \binom{n}{k_{\mathcal{E}}+i} + \lambda_{\mathcal{E}} \cdot \left(\binom{n}{k_{\mathcal{E}}+i+1} - \binom{n}{k_{\mathcal{E}}+i}\right) \\
&= \mathsf{BSize}(k_{\mathcal{E}} + i + 1, \lambda_{\mathcal{E}}).
\end{aligned}$$

25

Therefore, the induction hypothesis is correct and we have $\text{vol}(\mathcal{E}_r) \geq \text{BSize}(k_\mathcal{E} + r, \lambda_\mathcal{E})$ for all $r \in \mathbb{N}$. Note that $\text{vol}(\mathcal{E}_r) = \text{Risk}_r^{\text{ER}}(h, c)$, so we have

$$\text{Risk}_r^{\text{ER}}(h, c) \geq \text{BSize}(k_\mathcal{E} + r, \lambda_\mathcal{E}).$$

$\square$

Now we state a useful lemma that connects risk to robustness.

Lemma 4.17. *Suppose* $\mathsf{P} = (\{0,1\}^n, \mathcal{Y}, U_n, \mathcal{C}, \mathcal{H}, \text{HD})$ *is a classification problem. For any* $h \in \mathcal{H}$ *and* $c \in \mathcal{C}$*, if* $(k, \lambda) = \text{BSize}^{-1}(\mu)$ *for* $\mu = \text{Risk}(h, c) > 0$*, then we have,*

$$\text{Rob}^{\text{ER}}(h, c) = \sum_{r=0}^{n-k+1} \left(1 - \text{Risk}_r^{\text{ER}}(h, c)\right).$$

We now use the above lemma to prove Theorem 4.10.

Proof. Let $\mathcal{E}$ be the error region of (h, c). Let $\text{Pr}_{a \leftarrow D}[\text{HD}(a, \mathcal{E}) = r] = p_r$. Then, it holds that

$$\text{Rob}^{\text{ER}}(h, c) = \sum_{r \geq 0} r \cdot p_r = \sum_{r \geq 0} \sum_{i > r} p_i = \sum_{r \geq 0} (1 - \text{vol}(\mathcal{E}_r)) = \sum_{r \geq 0} (1 - \text{Risk}_r^{\text{ER}}(h, c)).$$

$\square$

Proof of Theorem 4.10. Based on Lemma 4.17,

$$\text{Rob}^{\text{ER}}(h, c) = \sum_{r=0}^{n-k+1} (1 - \text{Risk}_r^{\text{ER}}(h, c)).$$

Now Theorem 4.7 implies,

$$\text{Rob}^{\text{ER}}(h, c) \leq \sum_{r=0}^{n-k+1} 1 - \text{BSize}(k + r, \lambda).$$

$\square$

5 Conclusions

We discussed different definitions for adversarial perturbations. We saw that definitions that do not rely on the *error region* do not guarantee misclassification for the perturbed (adversarial) instance in every case. Apart from the conceptual differences of the definitions, this was also shown through an extensive study of attacking monotone conjunctions under the uniform distribution U_n (over $\{0,1\}^n$). In this study we were able to separate the adversarial robustness of hypotheses generated by popular algorithms, both in an asymptotic theoretical sense as well as in the average case by observing that the results of experiments also verify in practice the theoretical, worst-case, discrepancies on the robustness of the generated hypotheses. Subsequently, using the error region definition as the main definition for generating adversarial instances we were able to study and provide bounds for the adversarial risk and robustness of any classifier that is attacked when instances are drawn from the uniform distribution U_n.

There are however many interesting questions for future work. One such example is theoretical and experimental investigation of popular algorithms in order to determine the adversarial risk and robustness of the generated solutions in various cases. This may involve specific distributions or classes of distributions

and of course extend the investigation to concept classes beyond that of monotone conjunctions. Such investigations appear to be important because we can gain more insight on the geometry and the properties of the learned solutions and the extent to which such solutions are resistant to adversarial perturbations. For example, the solutions generated by FIND-S are consistently at least as resistant to adversarial perturbations (and for the largest part, significantly more resistant) as the solutions obtained by the SWAPPING ALGORITHM.

Another question is of course to generalize the results that we have for the error region definition under the uniform distribution, to distributions beyond uniform. What kind of tools and techniques are required? In the case that such bounds contain the uniform distribution as a special case, how do such bounds compare to the bounds that we explored, given that our investigation is specific to the uniform distribution? Furthermore, our results of Section 4 are information theoretic. Are there equivalent computationally efficient attacks by adversaries that can achieve such bounds?

Getting along, our results, like all other current provable bounds in the literature for adversarial risk and robustness, only apply to specific distributions that do not cover the case of image distributions. These results, however, are first steps, and indicate similar phenomena (e.g., relation to isoperimetric inequalities). Thus, though challenging, these works motivate a deeper study of such inequalities for specific distributions. In addition, the work of Fawzi et al. [FFF18] has a discussion on how attacks on "nice" distributions can potentially imply attacks on real data. That would be the case assuming the existence of a generative model that transforms "nicely"-distributed random latent vectors into natural data. Therefore, eventually an interesting line of work would be to investigate scenarios that also apply to the case of image classification and other very popular every-day classification tasks, highlighting further the importance of the investigations suggested in the previous paragraph.

References

[AKM18] Idan Attias, Aryeh Kontorovich, and Yishay Mansour. Improved generalization bounds for robust learning. *arXiv preprint arXiv:1810.02180*, 2018.

[AKST97] Dana Angluin, Mārtiņš Kriķis, Robert H. Sloan, and György Turán. Malicious Omissions and Errors in Answers to Membership Queries. *Machine Learning*, 28(2-3):211–255, 1997.

[AS94] Dana Angluin and Donna K. Slonim. Randomly Fallible Teachers: Learning Monotone DNF with an Incomplete Membership Oracle. *Machine Learning*, 14(1):7–26, 1994.

[BEHW87] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Occam's Razor. *Information Processing Letters*, 24(6):377–380, 1987.

[BFJ+94] Avrim Blum, Merrick L. Furst, Jeffrey C. Jackson, Michael J. Kearns, Yishay Mansour, and Steven Rudich. Weakly learning DNF and characterizing statistical query learning using Fourier analysis. In *STOC*, pages 253–262, 1994.

[BFR14] Battista Biggio, Giorgio Fumera, and Fabio Roli. Security evaluation of pattern classifiers under attack. *IEEE transactions on knowledge and data engineering*, 26(4):984–996, 2014.

[BGA57] Jerome S. Bruner, Jacqueline J. Goodnow, and George A. Austin. *A study of thinking*. John Wiley & Sons, New York, NY, USA, 1957.

[Bil08] Patrick Billingsley. *Probability and measure*. John Wiley & Sons, 2008.

[BIL+16] Osbert Bastani, Yani Ioannou, Leonidas Lampropoulos, Dimitrios Vytiniotis, Aditya V. Nori, and Antonio Criminisi. Measuring Neural Net Robustness with Constraints. In *NIPS*, pages 2613–2621, 2016.

[BTGN09] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi S. Nemirovski. *Robust Optimization*. Princeton Series in Applied Mathematics. Princeton University Press, October 2009.

[CW17a] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14. ACM, 2017.

[CW17b] Nicholas Carlini and David A. Wagner. Towards Evaluating the Robustness of Neural Networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 39–57, 2017.

[Dio16] Dimitrios I. Diochnos. On the Evolution of Monotone Conjunctions: Drilling for Best Approximations. In *ALT*, pages 98–112, 2016.

[DT09] Dimitrios I. Diochnos and György Turán. On Evolvability: The Swapping Algorithm, Product Distributions, and Covariance. In *SAGA*, pages 74–88, 2009.

[ESC+18] Gamaleldin F. Elsayed, Shreya Shankar, Brian Cheung, Nicolas Papernot, Alex Kurakin, Ian J. Goodfellow, and Jascha Sohl-Dickstein. Adversarial Examples that Fool both Computer Vision and Time-Limited Humans. *NIPS*, 2018.

[FFF18] Alhussein Fawzi, Hamza Fawzi, and Omar Fawzi. Adversarial vulnerability for any classifier. *arXiv preprint arXiv:1802.08686*, 2018.

[FGMP96] Michael Frazier, Sally A. Goldman, Nina Mishra, and Leonard Pitt. Learning from a Consistently Ignorant Teacher. *Journal of Computer and System Sciences*, 52(3):471–492, 1996.

[FMS15] Uriel Feige, Yishay Mansour, and Robert Schapire. Learning and inference in the presence of corrupted inputs. In *Conference on Learning Theory*, pages 637–657, 2015.

[FMS18] Uriel Feige, Yishay Mansour, and Robert E Schapire. Robust inference for multiclass classification. In *Algorithmic Learning Theory*, pages 368–386, 2018.

[GMF+18] Justin Gilmer, Luke Metz, Fartash Faghri, Samuel S Schoenholz, Maithra Raghu, Martin Wattenberg, and Ian Goodfellow. Adversarial spheres. *arXiv preprint arXiv:1801.02774*, 2018.

[GSS15] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *ICLR*, 2015.

[Har66] Lawrence H Harper. Optimal numberings and isoperimetric problems on graphs. *Journal of Combinatorial Theory*, 1(3):385–393, 1966.

[JS06] Jeffrey C. Jackson and Rocco A. Servedio. On Learning Random DNF Formulas Under the Uniform Distribution. *Theory of Computing*, 2(8):147–172, 2006.

[LM05] Daniel Lowd and Christopher Meek. Adversarial learning. In *KDD*, pages 641–647, 2005.

[LPV03] László Lovász, József Pelikán, and K Vesztergombi. *Discrete Mathematics: Elementary and Beyond.* Springer Science & Business Media, 2003.

[MFF16] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. In *CVPR*, pages 2574–2582, 2016.

[Mit77] Tom M. Mitchell. Version spaces: A candidate elimination approach to rule learning. In *IJCAI*, pages 305–310, 1977.

[Mit97] Thomas M. Mitchell. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997.

[MMS+18] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083; to appear in International Conference on Learning Representations (ICLR)*, 2018.

[MRT15] Yishay Mansour, Aviad Rubinstein, and Moshe Tennenholtz. Robust probabilistic inference. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 449–460. Society for Industrial and Applied Mathematics, 2015.

[Nig67] R. G. Nigmatullin. Some metric relations in the unit cube (in russian). *Diskretny Analiz 9, Novosibirsk*, pages 47–58, 1967.

[NRH+10] Blaine Nelson, Benjamin I. P. Rubinstein, Ling Huang, Anthony D. Joseph, and J. D. Tygar. Classifier Evasion: Models and Open Problems. In *PSDM*, pages 92–98, 2010.

[NRH+12] Blaine Nelson, Benjamin IP Rubinstein, Ling Huang, Anthony D Joseph, Steven J Lee, Satish Rao, and JD Tygar. Query strategies for evading convex-inducing classifiers. *Journal of Machine Learning Research*, 13(May):1293–1332, 2012.

[PMW+16] Nicolas Papernot, Patrick D. McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*, pages 582–597, 2016.

[Sel09] Linda Sellie. Exact learning of random DNF over the uniform distribution. In *STOC*, pages 45–54, 2009.

[SM00] Yoshifumi Sakai and Akira Maruoka. Learning Monotone Log-Term DNF Formulas under the Uniform Distribution. *Theory of Computing Systems*, 33(1):17–33, 2000.

[SPNR18] Arun Sai Suggala, Adarsh Prasad, Vaishnavh Nagarajan, and Pradeep Ravikumar. On Adversarial Risk and Training. *arXiv preprint arXiv:1806.02924*, 2018.

[SST10] Robert H. Sloan, Balázs Szörényi, and György Turán. *Learning Boolean Functions with Queries*, pages 221–256. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2010.

[SST+18] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *arXiv preprint arXiv:1804.11285*, 2018.

[ST94] Robert H. Sloan and György Turán. Learning with Queries but Incomplete Information (Extended Abstract). In *COLT*, pages 237–245, 1994.

[SZS+14] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2014.

[TSE+18] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness May Be at Odds with Accuracy. *arXiv preprint arXiv:1805.12152*, 2018.

[Val84] Leslie G. Valiant. A Theory of the Learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[Val09] Leslie G. Valiant. Evolvability. *Journal of the ACM*, 56(1):3:1–3:21, 2009.

[XEQ18] Weilin Xu, David Evans, and Yanjun Qi. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. *arXiv preprint arXiv:1704.01155. To appear in Network and Distributed System Security Symposium (NDSS)*, 2018.

A   Some Useful Facts

Lemma A.1. *Let $n \in \mathbb{N}^*$. Then,*

$$\sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} = \sum_{i=\lceil n/2 \rceil}^{n} \binom{n}{i} \geq 2^{n-1}.$$

Proof. If n is even, then $n/2$ is an integer and moreover $\lfloor n/2 \rfloor = n/2 = \lceil n/2 \rceil$. Therefore the sequence $\binom{n}{\lceil n/2 \rceil}, \binom{n}{\lceil n/2 \rceil+1}, \ldots, \binom{n}{n}$ has $1+n/2$ terms. The expansion $\binom{n}{0}, \binom{n}{1}, \ldots, \binom{n}{n}$ has $n+1$ terms and moreover $\binom{n}{j} = \binom{n}{n-j}$ for $j \in \{0, 1, \ldots, n/2 - 1\}$. Let $A = \sum_{j=0}^{n/2-1} \binom{n}{j} = \sum_{k=n/2+1}^{n} \binom{n}{k}$. Then, $A + \binom{n}{n/2} + A = 2^n \Rightarrow A = 2^{n-1} - \binom{n}{n/2}/2$. But this implies that $\sum_{i=\lceil n/2 \rceil}^{n} \binom{n}{i} = \binom{n}{n/2} + A = 2^{n-1} + \binom{n}{n/2}/2 > 2^{n-1}$.

If n is odd, then the sequence $\binom{n}{\lceil n/2 \rceil}, \binom{n}{\lceil n/2 \rceil+1}, \ldots, \binom{n}{n}$ has $1 + n - \lceil n/2 \rceil = 1 + \lfloor n/2 \rfloor = (1+n)/2$ terms. Further, again by symmetry we have $B = \sum_{j=0}^{\lfloor n/2 \rfloor} \binom{n}{j} = \sum_{k=\lceil n/2 \rceil}^{n} \binom{n}{k}$. Then, $B + B = 2^n \Rightarrow B = 2^{n-1}$. $\square$

Lemma A.2. *For any $m \in \mathbb{N}$ we have $\binom{2m}{m} \geq \frac{2^{2m-1}}{\sqrt{m}}$.*

Lemma A.3. *We have $\lim_{m \to \infty} \frac{2^{2m}}{\binom{2m}{m} \cdot \sqrt{m}} = \sqrt{\pi}$.*

Lemma A.4. *Let $k \in \mathbb{N}$. Then, $\sum_{i=0}^{k} i \cdot \binom{k}{i} = k \cdot 2^{k-1}$.*

Lemma A.5. *Let $1 \leq u \leq w$. Let $1 \leq \zeta \leq u$ and $1 \leq \xi \leq w$. Then,*

$$\frac{u}{8} \cdot 2^{u+w} \leq \sum_{\zeta=1}^{u} \binom{u}{\zeta} \sum_{\xi=\zeta}^{w} \binom{w}{\xi} \min\{\zeta, \xi\}$$

Proof. We have,

$$\sum_{\zeta=1}^{u}\binom{u}{\zeta}\sum_{\xi=\zeta}^{w}\binom{w}{\xi}\min\{\zeta,\xi\} \geq \sum_{\zeta=1}^{\lceil u/2\rceil}\zeta\binom{u}{\zeta}\sum_{\xi=\lceil w/2\rceil}^{w}\binom{w}{\xi}$$

$$\geq 2^{w-1}\sum_{\zeta=1}^{\lceil u/2\rceil}u\binom{u-1}{\zeta-1} \qquad \text{(Lemma A.1)}$$

$$= \frac{u}{2}\cdot 2^{w}\cdot\sum_{\zeta=0}^{\lceil u/2\rceil-1}\binom{u-1}{\zeta}$$

$$= \frac{u}{2}\cdot 2^{w}\cdot\sum_{\zeta=0}^{\lfloor(u-1)/2\rfloor}\binom{u-1}{\zeta}$$

We now use Lemma A.1 again and obtain the statement. □