Existence, computability and certification of optimal classifiers under adversarial example attacks

Raphael Ettedgui

PSL
UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à Université Paris-Dauphine

Existence, computability and certification of optimal classifiers under adversarial example attacks

Soutenue par
Raphaël ETTEDGUI
Le 15 Décembre 2022

École doctorale n°543
Ecole doctorale SDOSE

Spécialité
Informatique

Composition du jury :

Florence D'Alché Professeur, TELECOM PARIS	*Présidente*
Amaury HABARD Professeur, PARIS DAUPHINE-PSL	*Rapporteur*
Aurélien BELLET Chargé de recherche, INRIA PARIS	*Rapporteur*
Alexandre ALLAUZEN Professeur, PARIS DAUPHINE-PSL	*Examinateur*
Yann CHEVALEYRE Professeur, PARIS DAUPHINE-PSL	*Directeur de thèse*
Jamal ATIF Professeur, PARIS DAUPHINE-PSL	*Co-encadrant de thèse*

Dauphine | PSL
UNIVERSITÉ PARIS

A Mr Kausch, professeur incroyable
qui m'a transmis la passion des mathématiques

Merci pour tout...

Remerciements

Je remercie mes directeurs de thèse, Yann Chevaleyre et Jamal Atif, pour leur encadrement, et leur présence tout au long de ces trois années de thèse. C'est grâce à eux que cette expérience m'a autant plu et a été aussi enrichissante. Yann a été un soutien constant tant sur les plan mathématique et administratif que comme coach, m'a appris énormément de choses sur le métier de chercheur, et je le remercie infiniment pour cela. Merci à Jamal de m'avoir transmis sa rigueur intellectuelle, la passion du travail d'équipe, et d'avoir toujours été là pour soutenir mes travaux, même lorsque je m'écartais des sentiers battus. J'ai eu beaucoup de chance de travailler avec ces deux personnes incroyables.

Je tiens à remercier tout particulièrement aussi Rafael Pinot, qui a été comme un troisième encadrant de thèse pour moi. Il m'a formé et accompagné pendant notre travail commun sur mon premier article, a bien voulu faire des réunions régulières lorsque j'étais perdu entre plusieurs directions de recherche, et nos échanges mathématiques ont été précieux pour chacun de mes travaux. Son sérieux, sa générosité et sa clarté de pensée ont été des inspirations tout au long de ma thèse. Merci pour tout !

Cette thèse a été l'occasion de collaborations avec nombre de chercheurs talentueux, qui sont également des gens formidables avec qui j'ai pris un grand plaisir à travailler. Merci à Alexandre Araujo, Guillaume Carlier, Laurent Meunier, Geovanni Riszk, Emma Müller et Luca Ganassali pour toutes ces heures que nous avons passées à échanger, construire et découvrir ensemble à la pointe de la craie. Merci à tous les autres doctorants du labo pour les conversations déjantées, les groupes de lectures et les verres qui suivaient. Merci à l'ensemble des équipes administratives qui permettent au laboratoire de fonctionner aussi bien qu'il le fait.

Ces trois années ont été marquées en grande partie par la pandémie de Covid-19, durant laquelle j'ai consacré beaucoup de temps à diverses analyses mathématiques de la situation. Je remercie les très nombreuses personnes avec qui j'ai échangé autour du projet Covimpact, qui ont rendu possible de terminer un projet aussi gigantesque en quelques mois à peine. Je tiens à remercier tout particulièrement Shryka, dessinatrice talentueuse qui s'est occupée du design des personnages pour le site, Emma, qui m'a accompagné tout au long du projet et aidé à coacher les équipes, Clément, qui m'a formé en Réact et permis de

concevoir un site complexe aussi rapidement, ainsi que Laura, pour les brainstormings divers et son soutien.

Un grand merci à mes parents, Nathalie et Gabriel, qui ont toujours été là pour moi et m'ont permis d'être qui je suis aujourd'hui. Merci à ma famille de coeur, la RI, Ivan, Lennart, Charlotte, Auriane et JP, toujours là au quotidien quelles que soient les difficultés, et toujours si bienveillants et compréhensifs. Merci pour toutes ces heures à refaire le monde en vocal tout en détruisant des fourmis géantes, pour les fous rires en jeu de rôle et les innombrables memes sur le roulisme et la compagnie du poulet lumineux. Puissent nos IRL être nombreuses et toujours aussi déjantées ! =D

Merci à la petite famille de yèvres, Valou, Laura, Hélo et Flo, qui ont partagé mon quotidien pendant la pandémie. Je chéris énormément cette période, nos sessions cuisine et étirements kaamelott, qui m'ont été d'un grand soutien durant ma thèse. Merci à Laetitia, ma brain-mate avec qui c'est toujours un bonheur de refaire le monde et parler de science comme de business ! Merci à mes quasi frère et soeur, Maylis et Rémi (ou Biscotte et Camembert ? :p), à tous les amis de la danse, thanks Irina for the many conversations on our respective Phds and our mutual passion for science, Kathrin for your support and wonderful kindness, Hélène et Matt pour ces moments magiques à Strasbourg, à Paris et en corse (Matt, I'm still ahead in duolingo by the way :p), Pauline pour tous nos entraînements et longues discussions, Jess et Mika (et Spookiiiiie) pour votre gentillesse et les photos de chat en soutien, Santo et Laurent pour tous ces bons moments, toutes nos conversations, dans des villas comme dans des trains, Camille pour nos sessions de messages vocaux interminables (mais toujours passionnants), qui m'ont tenu compagnie pendant toute la période de rédaction, Bébert et Clem pour toutes les danses en switch et en vol, toutes nos conversations et bientôt plein d'autres ! Merci à Nat, Caro, Agnès, Joshua et Rachel, Maëlys, Léo, Marine, Simons et tous les autres qui rendent la communauté wcs aussi formidable. J'ai beaucoup de chance de vous connaître. :)

Abstract

This thesis investigates the question of adversarial examples in machine learning. These small, imperceptible perturbations of an input can be crafted to fool any state-of-the art classifier, and transfer to other models that solve the same task. Such vulnerabilities constitute a major barrier to the use of machine learning for critical applications, from self-driving cars to automated surgeries. Currently, no research team has found any "optimal" classifier, whose performance can be mathematically certified against any attack, and new defenses continue to be outperformed by new attacks. This raises two major question :

Q1: *Is it only possible to obtain a classifier that performs optimally under any attack?*

Q2: *If so, how much specific information would we need to certify its performance?*

We investigate **Q1** through the lens of game theory. We show that for the $0/1$ loss, there can be no pure Nash equilibriums. This means that no deterministic classifier can performs optimally against every attack : even with infinite processing power, we could only have guarantees against some specific set of attacks. When using convex surrogates, such equilibria can exist, but will always be unstable, i.e. impossible to compute in practice.

This encourages the use of randomization. We show that starting from any deterministic base classifier, it is possible to design a randomized mixture that strictly outperforms it under attack. We then provide an algorithm to output such a mixture, showing both theoretical guarantees and empirical results on CIFAR10 and CIFAR100. We then show that randomization in the form of noise injection increases the stability of equilibria at the cost of natural accuracy. We also show conditions on the existence of equilibria when the attacker is allowed randomization.

We then investigate **Q2** by conducting an analysis of randomized smoothing certification. We quantify the gap between current, single-noise certificates, and the best theoretically possible one, and show that it explodes as dimension increases when the curvature of the decision boundary is high. This shows that more information is needed to bypass the current impossibility results. We then focus on the class of noised-based certificates, and introduce a new framework to collect information from several noise distributions at the same time. By separating the information gathering from the smoothing, this additional information requires no further loss in accuracy. We show that this allows to approximate the perfect certificate arbitrarily well, at the expense of high computational cost. We finally study how to use invariances, symmetries and prior information to reduce that cost, and provide a randomization-based certificate that can be computed independently of the dimension.
We conclude this work with open research leads and perspectives.

Résumé

Cette thèse porte sur la question des attaques adversariales en machine learning. Ces perturbations, imperceptibles à l'oeil humain, sont conçues pour induire les classifieurs en erreur, et transfèrent à nombre de modèles similaires. Les meilleurs réseaux actuels sont vulnérables à ces attaques, ce qui constitue un obstacle majeur à l'utilisation du ML pour des applications critiques comme les voitures autonomes ou les drones. A l'heure actuelle, aucune équipe de chercheurs n'a identifié de classifieur "optimal", dont les performances seraient garanties contre toutes attaques, et les nouvelles défenses continuent d'être vaincues par de nouvelles attaques. Cela soulève deux questions :

Q1: *Existe-t-il un classifieur optimal contre toutes les attaques à la fois ?*

Q2: *Si oui, de quel degré d'information a-t-on besoin pour garantir ses performances ?*

Nous étudions **Q1** sous l'angle de la théorie des jeux. Nous montrons qu'il n'existe pas d'équilibre de Nash pur pour la $0/1$ loss, et que par conséquent aucun classifieur déterministe ne peut être optimal contre toutes les attaques à la fois, même avec une puissance de calcul infinie. Avec des surrogates convexes, des équilibres purs peuvent exister, mais ne peuvent être stable; autrement dit, ils seraient impossibles à calculer en pratique. Tout cela encourage l'usage de la randomisation.

Nous montrons qu'à partir d'un classifieur déterministe, il est possible de créer une mixture qui performe strictement mieux sous toute attaque. Nous présentons un algorithme permettant d'obtenir cette mixture, et fournissons des garanties théoriques et des résultats empiriques sur CIFAR10 et 100. Nous montrons ensuite que l'injection de bruit stabilise les équilibres, au prix d'une augementation du risque naturel. Nous étudions également les conditions d'existence d'équilibres de Nash lorsque l'attaquant est randomisé.

Nous étudions ensuite **Q2** en analysant la certification pour Randomized Smoothing. Nous quantifions l'écart entre les certificats actuels à bruit unique, et le certificat parfait, et montrons que cela explose lorsque la dimension augmente, aux points où la frontière de décision a une forte courbure locale. Cela montre la nécessité d'utiliser davantage d'information pour dépasser les résultats d'impossibilité actuels. Nous introduisons une méthode permettant de collecter de l'information de plusieurs bruits à la fois, indépendamment du smoothing et donc sans perte d'accuracy naturelle. Nous montrons que cela permet d'approximer le certificat parfait avec une précision arbitraire, au prix d'un fort coût en calcul. Nous étudions ensuite comment exploiter les invariances, symmétries et l'information a priori pour réduire ce coût, et présentons un certificat basé sur des bruits à centres aléatoires, pouvant être calculé indépendamment de la dimension du problème. Nous concluons ensuite cette thèse par des questions de recherche ouvertes, ainsi que quelques perspectives sur l'avenir du domaine.

Contents

List of Figures

List of Tables

Notations and Symbols

We use bold lower-case to denote vectors and functions with multidimensional outputs and standard lower-case to denote scalars and real-value functions. Depending on the context, we either use calligraphic font or upper-case to denote ensembles – most of the times calligraphic, sometimes upper-case to denote sub-sets or elements of a set of sets.

Algebra

$\mathbb{R}$	Set of real numbers	
$\mathbb{N}$	Set of natural integers	
$\mathbb{R}^d$	Set of d-dimensional real-valued vectors	
I_d	$d \times d$ identity matrix	
$[a]$	Set of integers between 1 and a	$[a] := \{1, \dots, a\}$
$\Delta(K)$	K dimensional simplex	$\Delta(K) := \{\boldsymbol{z} \in \mathcal{R}^K \text{ st } \|\boldsymbol{z}\|_1 = 1\}$
$\|\boldsymbol{v}\|_p$	ℓ_p-norm of $\boldsymbol{v} \in \mathbb{R}^d$ for $p \in [1, +\infty)$	$\|\boldsymbol{v}\|_p = \left(\sum_{i=1}^d \|\boldsymbol{v}_i\|^p\right)^{1/p}$
$\|\boldsymbol{v}\|_\infty$	Infinite norm of $\boldsymbol{v} \in \mathbb{R}^d$	$\|\boldsymbol{v}\|_\infty = \max_{i \in [d]}(\|\boldsymbol{v}_i\|)$
$B_p(\boldsymbol{v}, \alpha)$	ℓ_p ball with center $\boldsymbol{v} \in \mathcal{R}^d$ and radius $\alpha \geq 0$	$\{\boldsymbol{u} \text{ st } \|\boldsymbol{u} - \boldsymbol{v}\|_p \leq \alpha\}$
$B_p(\alpha)$	ℓ_p ball with center 0 and radius $\alpha \geq 0$	$\{\boldsymbol{u} \text{ st } \|\boldsymbol{u}\|_p \leq \alpha\}$
$\text{Vol}(B)$	Volume of the sub-space $B \subset \mathcal{R}^d$	

Probability

$\mathcal{A}(\mathcal{Z})$	σ-algebra of an arbitrary space $\mathcal{Z}$
$\mathcal{P}(\mathcal{Z})$	Set of probability distribution over $(\mathcal{A}(\mathcal{Z}), \mathcal{Z})$
$\mathcal{F}_{\mathcal{Z},\mathcal{Z}'}$	Set of measurable functions from $\mathcal{Z}$ to $\mathcal{Z}'$
$\boldsymbol{\psi}\#\rho$	Push-forward of $\rho \in \mathcal{P}(\mathcal{Z})$ by $\boldsymbol{\psi} \in \mathcal{F}_{\mathcal{Z} \times \mathcal{Z}'}$
$\mathbb{E}[.]$	Expectation of a random event
$\mathbb{P}[.]$	Probability of a random event
$\mathcal{N}(.,.)$	Gaussian distribution
$\text{Lap}(.,.)$	Laplace distribution
Φ	cdf of the standard Gaussian distribution $\mathcal{N}(0,1)$

Classification and Learning theory

$\mathcal{X}$	Input space
d	Dimension of the input space
$\mathcal{Y}$	Output space
$\mathcal{D}$	Ground-truth distribution
$\mathcal{S}$	Training sample
$\mathcal{H}$	Hypothesis space
$\mathcal{L}$	Loss function

Functions

$\mathbb{1}\{.\}$	Indicator function of an event	$\mathbb{1}\{A\} = 1$ if A is true, 0 otherwise
$\text{sign}(x)$	Sign function applied on x	$\text{sign}(x) = 1$ if $x > 0$, -1 if $x < 0$ and 0 if $x = 0$

Abbreviations

a.k.a.	**a**lso **k**nown **a**s
cdf	**c**umulative **d**ensity **f**unction
C & W	**C**arlini and **W**agner (attack)
e.g.	*exempli gratia*
Eq.	**Eq**uation
ERM	**E**mpirical **R**isk **M**inimization
FGM	**F**ast **G**radient **M**ethod (attack)
i.e.	*id est*
i.i.d.	**i**dentically and **i**ndependently **d**istributed
PGD	**P**rojected **G**radient **D**escent (attack)
resp.	**resp**ectively
s.t.	**s**uch **t**hat
SRM	**S**tructural **R**isk **M**inimization
std	**st**andard **d**eviation
w.r.t.	**w**ith **r**espect **t**o

1 Introduction

Contents

Machine learning algorithms are now at the center of many aspects of our everyday lives, from self driving cars to shops that automatically track your purchases. However, they sill exhibit some glaring vulnerabilities, that a malicious agent may use to fool even the most successful classifiers to this date. The industry is thus the scene of a race between attackers and defenders, whose stakes are the viability of machine learning itself. In this chapter, we present the context in which this thesis takes place, as well as our main motivations in section 1.1. We will then present, in section 1.2 the global problem that we will tackle, namely *How to design robust classifiers under adversarial perturbations?* Finally, we summarize the main contributions of this thesis in section 1.3.

1.1 Context & Motivation

1.1.1 Machine Learning and neural networks

Machine learning is a category of algorithms, that leverage historical data to accurately predict new outcomes without being explicitly programmed to do so. They are parametric functions, made of two main blocks : a *representation* of the data, extracting from the input

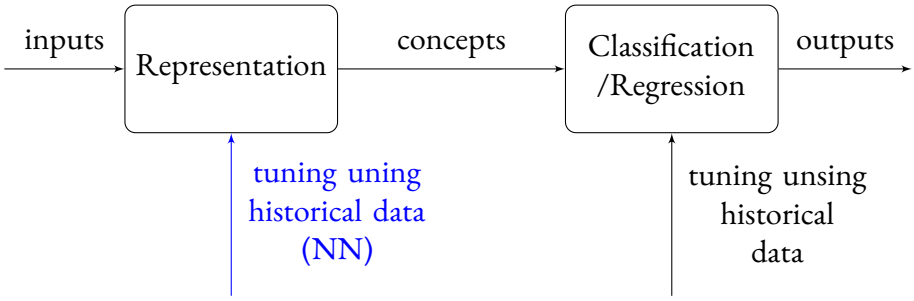


Figure 1.1: Decomposition of a machine learning algorithm. Standard algorithms only learn the classification function, whereas neural networks also learn the representation

the necessary concepts to solve the problem, and a *classification* or *regression* algorithm, that weight these concepts to predict new outputs (fig. C.2). When the data is well represented, simple algorithms like logistic regression are enough to make prediction in seemingly complex problems such as whether to recommend a cesarean section ([Mor-Yosef et al., 1990]).

Deriving "good" representations of the data however, is a much bigger challenge. Indeed, raw data is often impossible to analyze directly, even when using the most complex classification aglorithms. There is for example no direct correlation between the relative intensity of a pixel in an image, and the presence or absence of a dog in that image. For many applications with structured data, such as recommendation engines [Khanal et al., 2020] or spam filtering [Dada et al., 2019], representations were crafted manually by experts, using prior knowledge on the problem at hand. However, as tasks became more complex, working on high-level, unstructured data such as images and sounds, the representation design quickly became the bottleneck of the process. Hence came the idea of learning these representations from data as well.

Neural networks exist as early as 1943 with [McCulloch and Pitts, 1943] and the multi-layer perceptron. They consist on several layers, learning successive representations of the input, and expressing them in terms of other, more simple representations. For example, a cat can be described in terms of eyes, nose, ears and fur, which can all be described in terms of geometric shapes, themselves an arrangement of lines and angles, until the bottom layer directly manipulates the pixels of the image. Given enough depth, i.e. layers of representations, neural networks provide enough complexity to approximate any continuous function.

Convolutional neural networks were then introduced for image recognition by [LeCun et al., 1998], using prior information on the problem (namely translation invariance) to constrain the optimization. However, it is only with the progress of graphical programmable

units (GPUs) and specific architectures built to exploit their processing capabilities, that neural networks truly began to shine, as [Russakovsky et al., 2015] managed to outperform the state of the art on Imagenet by more than 10 points, using AlexNet, a convolutional network with over 60 million parameters. Today, deep neural networks achieve state-of-the art performances in various domains such as image recognition [Simonyan and Zisserman, 2014], natural language processing ([Zhang et al., 2018], and [Fedus et al., 2021] with over one trillion parameters), and speech recognition [Hinton et al., 2012], exhibiting remarkable generalization capabilities on new data.

1.1.2 Vulnerabilities on deep learning models

Although deep neural networks perform exceptionally well when evaluated in practice, their way of making predictions remains opaque, as the representation learned by the intermediate layers is not easily interpretable in human speech. The inability to provide an explanation for the decisions made by such algorithms raises the question of whether special cases may exist, that are not captured by test datasets, and where the algorithms would make illogical decisions.

In 2013, [Szegedy et al., 2014] identified such a vulnerability : deep neural networks can be fooled by small, imperceptible perturbations of the input, that are specifically crafted by a malicious attacker. These adversarial example attacks were then shown to fool algorithms in critical applications such as self-driving cars ([Sitawarin et al., 2018], [Morgulis et al., 2019]), speech recognition ([Alzantot et al., 2018], [Qin et al., 2019]) or malware detection systems ([Li et al., 2019]). Worse, those attacks are simple to implement, and can be reproduced by any individual with limited computational power. Furthermore, they can be computed with no information on the algorithm, as they *transfer* : an attack designed against one classifier will usually perform well against another that solves the same problem ([Liu et al., 2016]).

1.2 Position of the problem

In this thesis, we will focus on the problem of *supervised classification under adversarial perturbations*. We will start by recalling the standard classification problem, then formalize the presence of an attacker, and state the main questions that we will work on answering.

1.2.1 The standard binary classification problem

Let $\mathcal{X}$ be our input space (usually $\mathbb{R}^d$ for images), and $\mathcal{Y} = \{-1, 1\}$ be our output space. We consider that our algorithm will encounter images drawn from some unknown distribution $\mathcal{D}$ over $(\mathcal{X}, \mathcal{Y})$, that we call "ground truth distribution". The goal of the

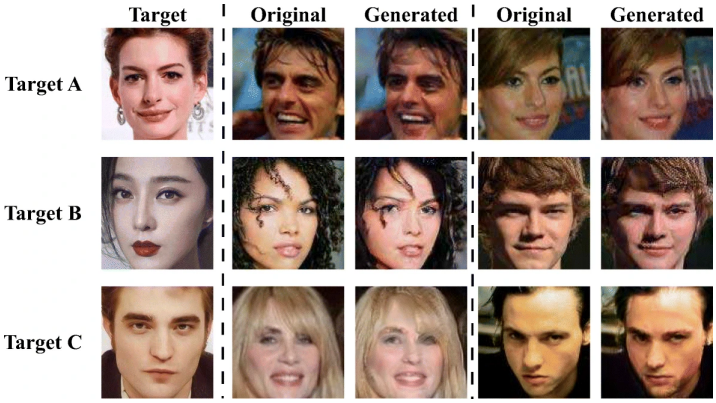


Figure 1.2: An illustration of adversarial example attacks from [Yang et al., 2021]. The column on the left contains the target faces. For every "original" image, an adversarial example is generated to be indistinguishable by a human, but classified by the algorithm as the target.

classification task is to associate each input $x \in \mathcal{X}$ to the unique label $y \in \mathcal{Y}$ that correctly describes it, using a classifier, i.e. a function $h : \mathcal{X} \to \mathcal{Y}$.

To evaluate and compare different classifiers, we use a *loss function* $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ that measures how far our prediction is from the true label. Ideally, we would use the zero-one loss, which counts the number of incorrect predictions made.

$$L_{0/1}(h(x), y) = \mathbb{1}_{h(x) \neq y} \tag{1.1}$$

We then select a hypothesis class $\mathcal{H}$, which is the set of functions that we consider to be pertinent for the problem at hand (for example continuous, linear, lipschitz ...). Our classification task then comes down to an optimization problem : find a classifier h that minimizes the risk, i.e. the average loss over the distribution $\mathcal{D}$:

$$\inf_{h \in \mathcal{H}} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [L(h(x), y)] \tag{1.2}$$

However, in practice we do not have access to the true distribution. We only have some historical data, in the form of a training set $S = \{(x_1, y_1), \dots (x_n, y_n)\}$, constituted of points drawn i.i.d. from $\mathcal{D}$. To approximate the solution from eq. (1.1), we thus replace $\mathcal{D}$ by the empirical distribution associated with $\mathcal{S}$, to obtain the *empirical risk minimization problem*:

$$\inf_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} L(h(x_i), y_i) \tag{1.3}$$

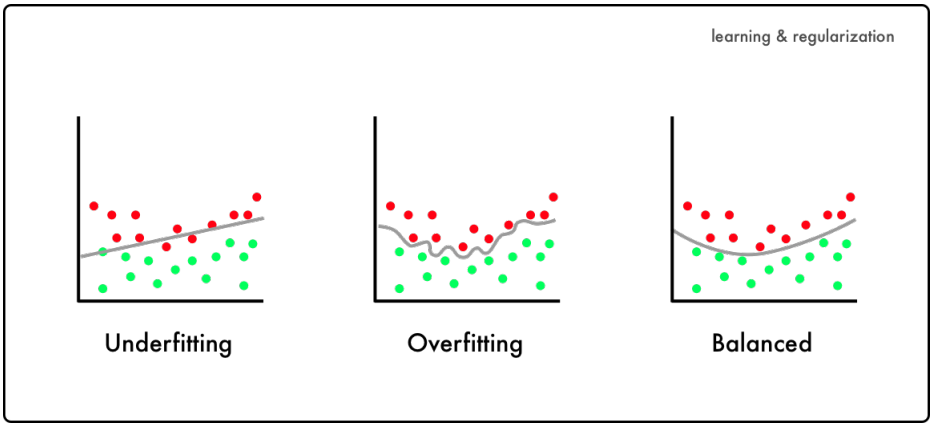


Figure 1.3: The generalization problem. Our classifier must fit the data well enough to capture its specificity and avoid underfitting, while avoiding too much complexity that would capture the randomness of the sampling (overfitting).

However, this does not provide us any guarantee that our empirical risk minimizer will also minimize the true risk. There is the risk of *overfitting*, i.e. tailoring our classifier so well to the empirical distribution that we capture part of the randomness of the sampling, instead of the underlying distribution (see fig. 1.3). We say that an algorithm *generalize* well if it maintains good performances even on unknown data.

One possible solution for algorithms to generalize, is to use a very large sample size. As n becomes large, the empirical distribution gets closer to the underlying one, and the law of large numbers provides some concentration bounds that ensures that the empirical risk is not too far from the risk. However, increasing the training size comes with additional computational cost, and is often not enough when the dimension of the problem is very high. As d increases, we need exponentially more samples for our algorithm to generalize properly (this is one of the aspects of the curse of dimensionality).

Another possible solution to this problem is to add prior information about the true distribution, by choosing a hypothesis class $\mathcal{H}$ that exhibits a "good" complexity for this type of data. For example, linear regressions are a type of classification problem where we expect the relationship between images and labels to be simple, and so restrict our hypothesis class to the set of all linear functions. We can also penalize complex hypothesis by introducing a *regularization*.

To study the generalization, we need some measure for the complexity of the hypothesis class. This can be defined in several way, such as VC-dimension [Vapnik and Chervonenkis, 2015] or the Rademacher complexity [Bartlett and Mendelson, 2002]. We use this complexity measure to bound the the generalization gap, i.e. the difference between the empirical risk and the true risk for any function $h \in \mathcal{H}$.

The combination of complexity bounds and concentration bounds from large sample size allow us to in practice use the empirical risk minimization as a satisfying proxy to the classification problem.

1.2.2 Classification under attack

In this setting, an external agent, called *the Attacker*, is allowed to perturbate any input $x \in \mathcal{X}$, with the goal of inducing a misclassification. More formally :

- For every input $x \in \mathcal{X}$ of class y, the Attacker can choose to replace it by a transformed input $\phi(x) = x + \tau$, with $\tau \in \mathcal{X}$;
- The Attacker has perfect information on the underlying distribution $\mathcal{D}$ and knows the true class of each input. Thus, he is in fact designing two functions : $\boldsymbol{\phi} = (\phi_1, \phi_{-1})$. This means that when an input (x, y) is sampled from $\mathcal{D}$, the Attacker will perturbate it into $(\phi_y(x), y)$.
- The score of the Attacker is the misclassification risk, to which we subtract an additional term $\mathbb{E}[c(x, \phi_y(x))]$ that represents the average cost of this attack. The Attacker's goal is to maximize that score.

Typically, $c(x, z) = \begin{cases} 0 & \text{if } \|x - z\|_p \leq \epsilon \\ +\infty & \text{otherwise} \end{cases}$, which represents the imperceptibility constraint : to be valid, the perturbation must be small enough to be invisible for a human eye. Note that since human sight is a complex process, involving many notions such as contrast and spatial-sensitivity (see for example [Cavonius and Estevez, 1975]), this is only a sufficient condition for imperceptibility, rather than perfectly encompassing the phenomenon. The choice of the norm (typically ℓ_2 or ℓ_∞) that is used in that constraint is important in practice, as they exhibit very different behaviors in large dimension. However, in our theoretical analysis, we will focus on existence and stability results, and so only use general properties of the constraint functions, which encompass all norms.

Note that this constraint takes a whole different meaning in fields such as text recognition, where imperceptibility cannot be defined in a pixel-wise manner ([Wang et al., 2022]).

We are thus confronted to a min-max problem, where both the Defender and the Attacker try to optimize the following function :

$$\mathcal{R}_c(h, \boldsymbol{\phi}) = \mathbb{E}[l(h(\phi_y(x)), y) - c(x, \phi_y(x))] \tag{1.4}$$

The adversarial nature of the problem (where two rational agents play against each other) makes it natural to investigate it under the prism of Game Theory.

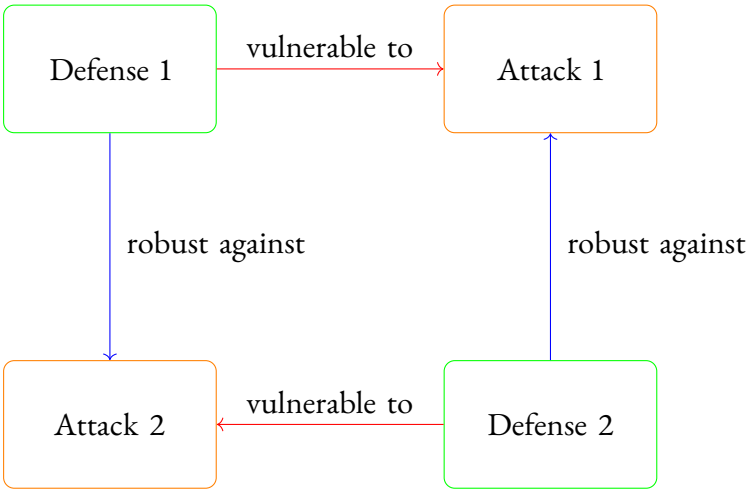


Figure 1.4: The cat and mouse games of adversarial attacks and defenses. New attacks are breaking state-of-the art defenses, then outperformed by different, more specific defenses which are themselves more vulnerable to other attacks.

1.2.3 A game theory perspective on adversarial example attacks

When this thesis began, the literature around adversarial examples revolved mainly around designing efficient attacks or defenses, evaluating them against each other, with the hope of improving the overal "strength" of attacks and defenses. The implicit hypothesis behind this line of research was that there exists some perfect defense and/or perfect attack, which could be reached with small iterative improvements, either in technique or computational power.

This theory started to be questioned with [Tramer et al., 2020], which considerably shook the domain, by showing the importance of the specificity of attacks and defenses. A very elaborate defense can be broken by an old and simple attack, that is just slightly adapted to the specificity of the classifier. Worst, when defenses are designed to be robust against a certain form of attacks, they are often vulnerable to several other types. The situation can be summarized by fig. 1.4 : attacks and defenses follow a cat-and-mouse situation, outperforming each other in a seemingly unendless cycle.

This raises the following question :

Q1: *Does a perfect defense or attack exist, and is it possible to compute ? Or is the chain of attacks and defenses cyclic, and will never lead to a final solution ?*

This question has a natural formulation in Game theory in terms of *Nash equilibria*, i.e. stable states in the game, where no player has an incentive to modify its behaviour. We will

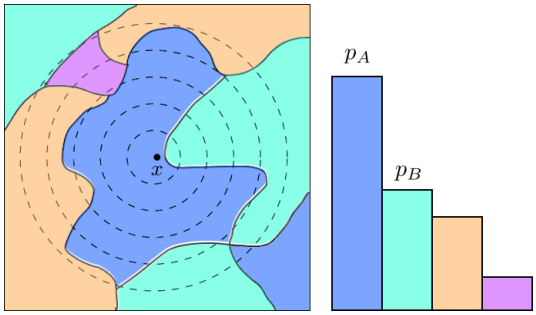


Figure 1.5: Illustration of Randomized Smoothing. We sample around the input according to some noise distributions, compute the relative probabilities of each class and return the most probable one.

investigate this problem in chapter 3, and provide a definitive answer in the deterministic regime.

We will then study whether such equilibria may actually arise in practice, through the study of *stability*. This can be described as the "margin of error" that a computing algorithm would have to identify equilibria, when trying to reach the optimal solution.

1.2.4 Randomized smoothing and certification

In the second part of this thesis, we will tackle the question of how to certify the performance of classifiers under limited information. For that, we will focus on a framework called *randomized smoothing*, which provides guarantees of robustness by collecting information on the classifier from noise-based queries. We will conduct an analysis of current certification algorithms, and show that the current limitations that weight on the method can be lifted using information from several noise distributions at the same time. We then introduce a new framework to derive the corresponding robustness guarantees.

Randomized Smoothing The Randomized smoothing of a classifier h at point x is a new classifier, which returns the most probable value of $h(x+u)$ when u is sampled from a noise distribution q_0. In practice, this is done using Monte-Carlo sampling, simulating many $u_i \sim q_0$ and counting the number of occurrences of each class for $h(x+u_i)$ (see fig. 1.5).

When using gaussian noise, this method allows to compute a certificate that is independent of the dimension, and uses no information on the classifier outside of the values $p_y = \mathbb{P}_{u\sim q_0}[h(x+u) = y], y = \pm 1$. Furthermore, it performs remarkably well on image classification, being the current state-of-the art of provable defenses ([Salman et al., 2019]).

Recent papers ([Yang et al., 2020], [Kumar et al., 2020]) have introduced *impossibility results*. Namely, there is a tradeoff between robustness and performance : as the dimension of the problem increases, keeping a constant level of robustness requires a drastic loss of natural accuracy. That method is hence currently considered as "doomed" in the community.

However, these impossibility results all rely on the current method of certification, which uses the same noise distribution q_0 to gather the information p_y, and to smooth the classifier. This leads to the natural question :

Q2: *Are these impossibility results intrinsical to Randomized smoothing, or are they just a byproduct of the certification method ?*

We will answer that question by exploring the limitations of current certificates, and introducing a framework that allows to bypass the impossibility results.

1.3 Main contributions of the thesis

This thesis is constituted of two main parts. The first one is a theoretical analysis of the adversarial example problem under the prism of game theory and optimal transport. We answer **Q1** by showing that no stable state can exist in the deterministic regime, and so that the quest for a perfect attack and defense cannot be successful through empirical exploration only. We then explore randomization as a promising research lead, showing that it brings stability to the game and increase the Defender's performance.

The second part is a study of Randomized smoothing certification. We answer **Q2** by exploring the limitations of current certificates, and showing that the gap between them and the perfect certificate grows arbitrarily large with the dimension of the problem. We then introduce a framework to obtain new certificates, and show that this allows to bypass the impossibility results.

A game theory perspective on adversarial attacks and defenses

We formulate the global problem of adversarial attacks and defenses as a two-players zero-sum game. On one side, the Defender solves a standard classification problem on the perturbated distribution, whereas on the other side the Attacker solves an optimal transport problem, trying to move the conditional distributions toward each other. Our contributions are the following:

1. We show that when both players play deterministic strategies, no stable pure Nash equilibrium can exist in the game. Furthermore, under any realistic cost function for the Adversary, no Nash equilibrium can exist at all;

2. We show several benefits of using randomization for the Defender. First, it is possible to outperform any deterministic classifier using a simple randomized one, that we provide an algorithm to design. Then, we show that noise injection increases the stability of Nash equilibria, and quantify the accuracy/stability tradeoff that this induces on some toy distributions.

3. We generalize the Attacker's problem into an optimal transport one, and provide general conditions for the existence of Nash equilibria for convex surrogate loss functions.

A theoretical study of Randomized Smoothing certification

We provide a general framework to evaluate and compare certificates. Using that, we study the limitations of current schemes, and introduce a new method for bypassing them.

1. We show, using easy to interpret decision boundaries, that single-noise certificates are blind to the local curvature of the decision boundary. We quantify the gap between single-noise certificates and the perfect one, and show that it explodes as the dimension of the problem increases for points of high local curvature, thus explaining the impossibility results.

2. We evaluate on neural networks the importance of this underestimation, by providing a sufficient conditions for the certificate to be locally suboptimal. We show that this is the case for at least 40% of the points on CIFAR10, suggesting that huge gains in robust accuracy are possible.

3. We introduce a new framework for designing certificates, that leverages the Generalized Neyman-Pearson lemma to gather information from several noise distributions at the same time. We show that this allows to get as close as desired from the perfect certificate with no loss of natural accuracy, at the cost of high computational overhead.

4. We then provide some insights on how to reduce the computational issues. We show that computing the Neyman-Pearson set can be reduced to a combinatorial problem when gathering information from uniform noises, and that Monte-Carlo sampling can be done in one dimension when using Gaussian noise. Finally, we introduce the notion of high-probability certification, and show that introducing

randomization in the certification process allows us to obtain certificates that are independent of the dimension.

Outline of the thesis

The rest of the thesis is organized as follows : Chapter 2 and 4 present overviews of the adversarial attacks game and randomized smoothing respectively, Chapter 3 and 5 contain our main contributions as summarized above, and Chapter 6 presents a summary of the thesis, as well as several discussions and open problems. Finally, the appendix contains additional results obtained in parallel of this thesis, such as on the consistency of loss functions in the adversarial setting, and on designing efficient contact tracing schemes against epidemics.

1.3.1 Research papers

During this thesis, we have contributed to several research papers, some being published and others still in the process of writing :

1st author

- "Randomization Matters : how to defend against strong adversarial attacks" *(ICML 2020)*
  Rafaël Pinot*, Raphaël Ettedgui*, Geovani Rizk, Yann Chevaleyre, Jamal Atif;
- "Towards evading the theoretical limitations of Randomized smoothing : a theoretical analysis" *(arxiv pre-print)*
  Raphaël Ettedgui, Alexandre Araujo, Rafaël Pinot, Yann Chevaleyre, Jamal Atif
- "Stability of Nash Equilibria in the adversarial examples game" *(ongoing work)*
  Raphaël Ettedgui, Yann Chevaleyre, Jamal Atif
- "Deep dive on contact tracing for epidemics" *(ongoing work)*
  Raphaël Ettedgui, Emma Müller

Other papers

- "Towards Consistency in Adversarial Classification" *(ICML 2022)*
  Laurent Meunier, Raphaël Ettedgui, Rafaël Pinot, Yann Chevaleyre, Jamal Atif

2 Background on adversarial examples and game theory

The field of adversarial examples has received a considerable interest in the last few years. The game theory and optimal transport perspective, which were emergent at the beginning of this thesis, are now active research areas, with several important papers published each year. In this chapter, we will provide an overview of that field. We summarize the standard classification problem in Section 2.1, then the state of the arts of adversarial attacks in Section 2.2 and defenses in Section 2.3. We provide an introduction to game theory in Section 2.4, optimal transport in Section 2.5 and summarize the recent papers that link these two concepts to adversarial attacks in Section 2.6.

2.1 The classification problem and Neural Networks

In this section, we will recall the definitions given in the introduction, and provide some important result about classification theory.

2.1.1 Statistical learning theory

Let $\mathcal{X}$ be our input space, and $\mathcal{Y} = \{-1, 1\}$ our output space. For more generality, we will consider classifiers as functions $h : \mathcal{X} \to \mathbb{R}$. This means that instead of just returning a label (which would correspond in this case to $\text{sign}(h)$), they also indicate their *degree of confidence* in the prediction. In this context, a loss function will be a function $L : (\mathbb{R} \times \mathcal{Y}) \to \mathbb{R}_+$, the most classical being the zero-one loss.

Hypothesis set Our research of "good" classifiers will often be guided by prior beliefs on how a solution to the task at hand should behave. Hence, we restrict our optimization problem to some given subset of all measurable functions, which we call the *hypothesis set* $\mathcal{H}$.

$$L_{0/1}(h(x), y) := \mathbb{1}_{\text{sign}(h(x)) \neq y} \tag{2.1}$$

Definition 1 (Standard classfication problem)**.** *Given a ground-truth distribution $\mathcal{D}$ over $(\mathcal{X}, \mathcal{Y})$, the classification problem is, as in the introduction, defined by the Risk Minimization (RM) problem, i.e. :*

$$\inf_{h \in \mathcal{H}} \mathcal{R}_L(h) \tag{RM}$$

where $\mathcal{R}_L(h) := \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[L(h(x), y)]$

Bayes optimal classification. For the $0/1$ loss, if we have access to the ground-truth distribution $\mathcal{D}$, we can easily compute the optimal classifier. It simply returns the most probable class given the input.

Definition 2 (Bayes optimal classifier)**.** *The Bayes optimal classifier is any function h_{bayes} such that almost surely, for $x \in \mathcal{X}$, we have :*

$$\operatorname{sign}\big(h_{bayes}(x)\big) = \arg\max_{i=\pm 1} \mathbb{P}[y = i|x] \tag{2.2}$$

By definition, it is the classifier with the lowest possible probability of misclassification, for the $0/1$ loss.

However, the $0/1$ loss is non-convex, and so optimizing it is an NP-hard problem. Although we use it for theoretical analysis, in practice we use surrogate losses as a proxy to optimize the $0/1$. This however only works when the loss function satisfies a property called consistency ([Bartlett et al., 2006], [Steinwart, 2007]) : its minimizing sequences must also minimize the $0/1-$loss.

Definition 3 (Consistency of a loss function)**.** *We say that the loss function L is consistent with regard to the $0/1$ loss if for every sequence of classifier h_n, we have :*

$$\mathcal{R}_L(h_n) \xrightarrow{n\to\infty} \inf_h \mathcal{R}_L(h) \Rightarrow \mathcal{R}_{L_{0/1}}(h_n) \xrightarrow{n\to\infty} \inf_h \mathcal{R}_{L_{0/1}}(h) \tag{2.3}$$

This means that any minimizing sequence for the surrogate loss is also a minimizing sequence for the $0/1$.

We will conduce a more through study of consistency in the standard and adversarial case in Appendix B. For now, we simply deduce from this result that in the standard case,

the $0/1$ loss can be optimized via gradient descent on a convex, consistent surrogate loss such as the hinge or cross-entropy losses.

2.1.2 Training and testing : the generalization gap

In practice, as the ground-truth distribution is not known, we use historical data to build a proxi distribution, that we call the *empirical distribution*. Let $\mathcal{S} = \{(x_1, y_1), \dots, (x_n, y_n)\}$ where the (x_i, y_i) are drawn i.i.d. from $\mathcal{D}$. Then the empirical distribution is the average of the Dirac distributions corresponding to the sampled points :

$$\frac{1}{n}\sum_{i=1}^{n} \delta_{x=x_i}$$

We thus evaluate the perfomance of our classifiers on this empirical distribution, which leads to the *Empirical Risk Minimization* (ERM) problem :

$$\inf_{h\in\mathcal{H}} \mathcal{R}_L^n(h) \text{ where } \mathcal{R}_L^n(h) = \sum_{i=1}^{n} L(h(x_i), y_i) \tag{ERM}$$

Let $\hat{h}_n$ be the minimizer of Equation (ERM), and $\hat{h}$ be the minimizer of Equation (RM). To evaluate the *generalization gap*, i.e. how much risk we gained by using our empirical distribution as a proxy, we must bound the quantity :

$$\mathcal{R}_L(\hat{h}_n) - \mathcal{R}_L(\hat{h})$$

Note that the quantity of interest is the risk, not the empirical risk, which is just a proxy. We can bound this gap using what is called the *generalization error* :

Definition 4 (Generalization Error)**.** *The generalization error for loss L and sample $\mathcal{S}$ is defined by :*

$$\mathcal{E}_L^n = \sup_{h\in\mathcal{H}} |\mathcal{R}_L(h) - \mathcal{R}_L^n(h)| \tag{2.4}$$

Proposition 1 (Bounding the generalization gap)**.**

$$\mathcal{R}(\hat{h}_n) - \mathcal{R}_L(\hat{h}) \leq 2\mathcal{E}_L^n \tag{2.5}$$

Proof.

$$\mathcal{R}(\hat{h}_n) - \mathcal{R}_L(\hat{h}) = \mathcal{R}(\hat{h}_n) - \mathcal{R}_L^n(\hat{h}_n) + \mathcal{R}_L^n(\hat{h}_n) - \mathcal{R}_L^n(\hat{h})$$

$$\leq \mathcal{R}(\hat{h}_n) - \mathcal{R}_L^n(\hat{h}_n) + \mathcal{R}_L^n(h_*) - \mathcal{R}_L^n(\hat{h})$$
$$\leq 2 \sup_{h \in \mathcal{H}} |\mathcal{R}_L(h) - \mathcal{R}_L^n(h)|$$

□

This means that if we can bound, for every classifier, the difference between the risk and the empirical risk, we can control the generalization gap as a whole. This bound is of course not tight, but usually sufficient for our needs.

To bound the generalization error (and thus the generalization gap), we need a way of evaluating the complexity of our class of functions, i.e. how well it can fit precise data. Among the classical measures of complexity are the VC-Dimension ([Blumer et al., 1989]) and the *Rademacher complexity* ([Bartlett and Mendelson, 2002]). We will describe the second one here, as it is more useful to analyze rich classes of functions, such as neural networks.

Definition 5 (Rademacher Complexity)**.** *The empirical Rademacher complexity of class $\mathcal{H}$, for some sample $\mathcal{S} = \{z_1, \ldots, z_n\}$ is defined as :*

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{H}) := \mathop{\mathbb{E}}_{i=1\ldots n, \sigma_i \sim U(\{\pm 1\}} \left[\sup_{h \in \mathcal{H}} \left(\frac{1}{n} \sum_{i=1}^{n} \sigma_i h(z_i) \right) \right] \tag{2.6}$$

The σ_i are called Rademacher variables, i.e. uniformly distributed on $\{\pm 1\}$ (i.e. $\mathbb{P}[\sigma_i = \pm 1] = \frac{1}{2}$).

The Rademacher complexity of a class of functions represents, in a way, how well it can be used to fit some random noise. Intuitively, classes with high Rademacher complexity will tend to generalize worst, and be more prone to overfitting, whereas classes with small Rademacher complexity will have a very small expressive power and be prone to underfitting on complex problems.

Theorem 1 (Rademacher bound, [Bartlett and Mendelson, 2002]). *Let $\mathcal{H}$ be some class of functions, L a loss functions bounded by $M > 0$, and $\mathcal{S} = \{z_1, \dots, z_n\}$ a sample. Let $F(L, \mathcal{H}) := \{(x, y) \mapsto L(h(x), y) | h \in \mathcal{H}\}$ be the loss class, i.e. all possible combinations of the loss function and the hypothesis. Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, we have :*

$$\mathcal{E}_L^{\mathcal{S}} \leq 2M\mathfrak{R}_{\mathcal{S}}(F(L, \mathcal{H})) + 3M\sqrt{\frac{\ln 2/\delta}{2n}} \tag{2.7}$$

2.1.3 An important hypothesis class : neural networks

In this thesis, we will mainly use neural networks in our implementations. Here, we give an overview of that class of functions, as well as the different architectures that we will use. In this section, we will consider $\mathcal{X} = \mathbb{R}^d$ for some $d > 0$.

Definition 6 (Linear classifier). *A linear classifier of parameter $\theta \in \mathbb{R}^d$ is defined by :*

$$h_\theta : x \mapsto \theta^\intercal x \tag{2.8}$$

Classifying using linear functions consists in divinding the space in two subsets by a hyperplane. These models are easy to study and train, but lack expressivity, as most real datasets are not linearily separable. Furthermore, composing two linear functions gives another linear function, adding no complexity to the class.

This is usually solved using *representations* of the data : we first transform our input in some space that is well chosen to preserve most of the variance of the dataset, while being linearily separable. The main idea of neural networks is to learn these representations using a composition of linear classifiers and non-linear *activation functions*, typically the Rectified Linear Unit ([Nair and Hinton, 2010]).

Definition 7. *The Rectified Linear Unit (ReLu) is defined on any space V by :*

$$\forall x \in V, \sigma(x) = \max(0, x)$$

Currently, the Leaky ReLu $\max(\alpha, x)$ for some small $\alpha > 0$ is most commonly used, to avoid zero-valued gradients and accelerate the convergence of networks.

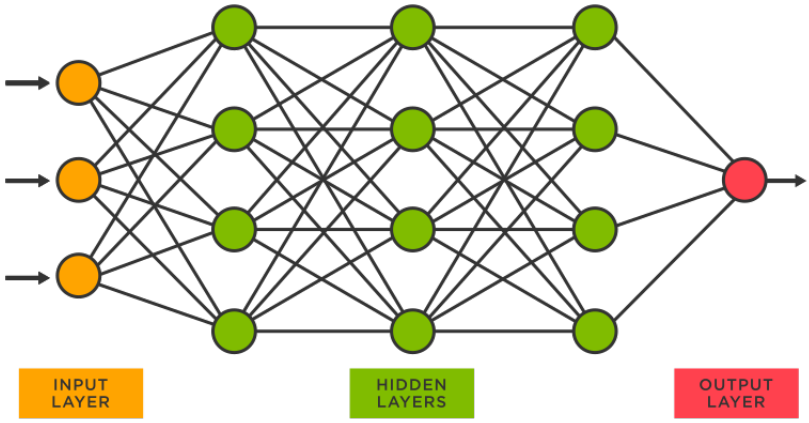


Figure 2.1: Representation of a neural network.

Definition 8 (Neural networks)**.** *A feed-forward neural network is a composition of activation functions and linear functions, of the form :*

$$\sigma \circ h_{\theta_1} \circ \cdots \circ \sigma \circ h_{\theta_m}$$

With $\theta_1, \ldots, \theta_m \in \mathbb{R}^d$.

We call each block $\sigma \circ h_{\theta_i}$ a *layer* of the network, and usually represents the overall network as several blocks linked to each other. The intermediate layers are called *hidden*, since the user usually never know their values, contrary to the first and last layer. They correspond to the representation learning, converting the input into some concept space that is better fit for the classification problem.
We therefore often view the neural network as a graph whose nodes are for each layer the canonical base of the current representation space, and whose edges are weighted by the linear coefficients.

On top of this basic structure, several training methods are used to avoid overfitting and allow networks to generalize better :

- **Dropout** is a way of adding noise during the training, by zeroing at each step some of the nodes that are chosen at random. This ensures some form of smoothness of the function learned by the network, as it cannot be too sensitive to small changes in the inputs.

- **Regularization** consists in adding a term in the loss function that penalizes the norm of the parameters θ_i. This corresponds to adding prior beliefs on the class

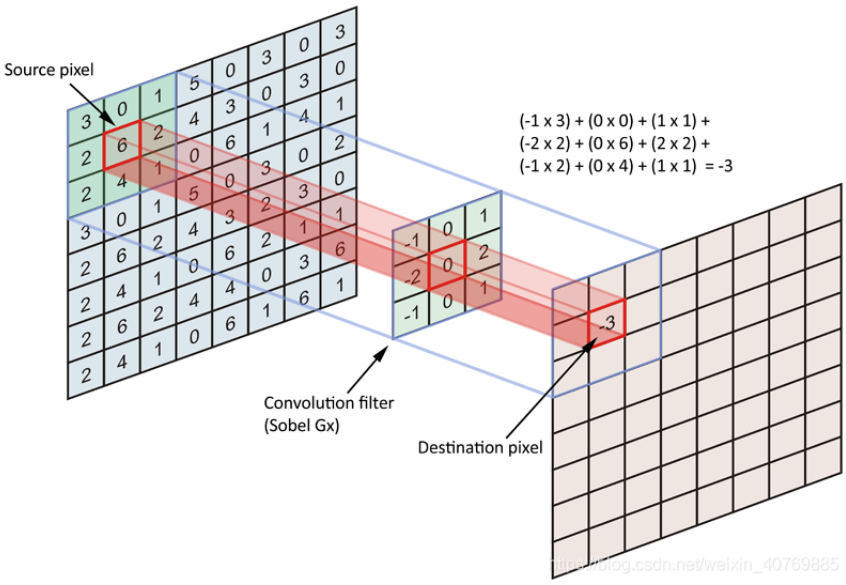


Figure 2.2: Illustration of the convolution mechanism for images.

of functions, putting more probability mass on the most "simple" functions. For example, an ℓ_1 regularization will typically provide sparse parameter vectors, whereas the ℓ_2 will give small, non-zero values. A smoothness prior will lead to smoother functions, etc.

Convolutions are a way of exploiting the translation-invariant properties of some input, typically images. The idea is to apply a regularization to an ordinary neural network, to only keep weights in a small window around each pixel, which we slide along the input so that the result does not depend on the position in the image (see Figure 2.2).

Residual neural networks The strength from neural networks comes from their expressive power, that is supposed to grow with their depth, as we use more and more parameters. But for that to be the case, the consecutive function classes must be "nested", i.e. each layer makes the network strictly more expressive (see Figure 2.3). To bypass that issue, [He et al., 2016] introduced the idea of residual networks : each block learns $f(x) - x$ instead of $f(x)$, to ensure that the identity mapping can always be learned, and so that every function learned by the previous blocks can still be learned after adding a layer.

This leads to the architecture we mainly use in this thesis for experiments, namely *ResNets* and its variants.

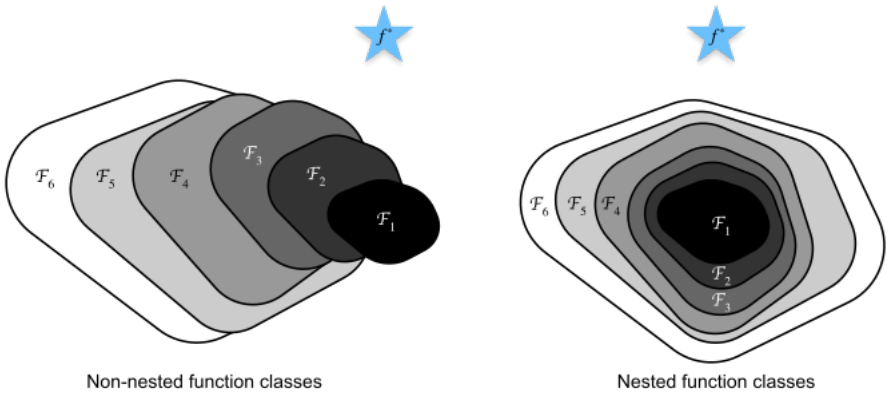


Figure 2.3: The difference between a non-nested and nested function class. In the first case, we loose part of the expressive power when adding a layer, while in the second case we strictly gain. Figure from [Zhang et al., 2018]

CIFAR10 and CIFAR100 datasets We use the CIFAR datasets ([Krizhevsky and Hinton, 2009] to evaluate our models. They are both comprised of the same 60000 32×32 images with 3 channels, for a total dimension of 3072.These images are labelled in 10 and 100 classes respectively for CIFAR10 and CIFAR100. This gives a total of 6000 per class for the first one, and only 600 for the second one, which is thus harder to classify. Current state-of-the art models achieve 99.5% natural accuracy on CIFAR10 ([Dosovitskiy et al., 2020]), and 96.08% on CIFAR100 ([Foret et al., 2020]).

2.2 The problem of adversarial examples in machine learning

Let us now give an overview of the adversarial example attacks against deep neural networks.

2.2.1 Adversarial example attacks

An adversarial example is a small, imperceptible perturbation of an input, which fools the classifier with high confidence. They were first introduced by [Szegedy et al., 2014], with the classical example shown in Figure 2.4. In a more mathematical way, given a classifier h and an input x, this amounts to finding $\delta \in \mathcal{X}$ that is "small enough to be humanly imperceptible", and such that $h(x + \delta) \neq h(x)$.

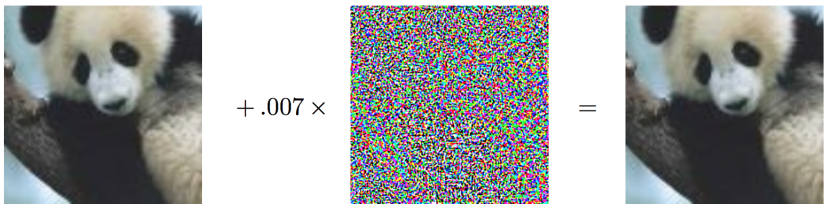


Figure 2.4: An example of adversarial example attack.

Definition 9 (Adversarial example Attack)**.** *Let h be some classifier, L some loss function, $x \in \mathcal{X}$, and $\mathcal{C}(x) \subset \mathcal{X}$ be some constraint set that represent imperceptibility. Then the adversarial example problem is :*

$$\sup_{\delta \in \mathcal{C}(x)} L(h(x+\delta), y) \tag{2.9}$$

Note that this problem may not exhibit any solution. However, this is only used as a proxy for the $0/1$ loss-version of the problem, which is much harder to optimize :

$$\text{Find } \delta \in \mathcal{C}(x) \text{ such that } h(x+\delta) \neq h(x) \tag{2.10}$$

And an approximation of the optimum in Equation (2.9) (such as one obtained via gradient descent) is usually enough to induce a misclassification.

Imperceptibility As we mentionned in Chapter 1, defining the imperceptibility condition is a hard problem, that is often bypassed by using a constraint $\|\delta\| \leq \epsilon$ for some norm $\|.\|$ and some $\epsilon > 0$. The most standard constraints used in the literature rely on the ℓ_p (for various p) and ℓ_∞ norms. Note that as the dimension of the problem grows, the norms behave in increasingly different ways, their unit balls overlapping on an ever smallest part of the space. It follows that empirical robustness results rely heavily on the choice of the norm.
For our theoretical results, we will most of the time rely on general properties of the norm functions, but will sometimes need to specify when we use the ℓ_2 or ℓ_∞ norms, especially for implementations.
The constraint set thus becomes $\mathcal{C}(x) = B_p(x, \epsilon)$ where $p \in (0, \infty]$.

Black and white box settings An important question is the amount of information to which the attacker has access. Typically, most attacks require some knowledge on the parameters of the models, to compute some form of gradient descent. We call *white-box setting* the case where the attacker has perfect information, and *black-box setting* the case where it has no information outside of the format of the inputs and outputs.

Transferability of attacks However, these two frameworks appear to be equivalent in practice, due to the existence of the *transferability* phenomenon. Attacks that are computed on a given classifier will often transfer to all similar models ([Tramèr et al., 2017], [Zhong and Deng, 2020]). It follows that we can focus our theoretical study on the white-box setting without much loss of generality (on top of following the principle of maximal precaution from cryptography).

2.2.2 Different types of attack models

To solve Equation (2.9), two methods are essentialy being used : minimizing the loss directly, and then ensuring that the perturbation is small enough, or incorporating the constraint to the loss in the form of a Lagrangian, and optimizing this new condition via gradient descent.

Fast gradient sign method [Goodfellow et al., 2015] introduced the most direct kind of attack, which is called FGSM or "fast gradient sign method". This consists in simply taking a step in the direction of maximal increase of the loss :

$$x^{t+1} = x^t + \epsilon \frac{\nabla_x L(h(x^t), y)}{\|\nabla_x L(h(x^t), y)\|} \tag{FGSM}$$

Projected Gradient Descent This can be further extended by taking several gradient steps, and enforcing the imperceptibility constraint by projecting at each step on $B_p(x, \epsilon)$. This attack framework is called Projected Gradient Descent, was one of the earliest benchmarks to evaluate empirical defenses against ([Madry et al., 2018], [Kurakin et al., 2018]). This can be summarized in the following way :

Definition 10 (Projected Gradient descent (PGD)). *The PGD algorithm for classifier h and loss L at point (x, y) is defined by the sequence $x^0 = x$ and for any $t > 1$:*

$$x^{t+1} := \Pi_{B_p(x,\epsilon)}\left(x^t + \epsilon \frac{\nabla_x L(h(x^t), y)}{\|\nabla_x L(h(x^t), y)\|}\right) \tag{2.11}$$

Where $\Pi_{B_p(x,\epsilon)}$ is the projection operator on $B_p(x, \epsilon)$.

Relaxing the constraint Another fruitful approach consists in switching the place of the constraint and the minimization problem. This means minimizing the norm of the perturbation, under the constraint that it induces a misclassification. This has the benefit of being easier to optimize by stochastic gradient descent using the Lagrangian formulation :

$$\inf_{\delta} \|\delta\|_p + c \cdot f(x + \delta) \tag{2.12}$$

Where f is some objective function, chosen such that $f(x+\delta) \leq 0 \iff L(h(x), y) \geq \alpha$ for some chosen threshold α (which is selected empirically to be a good proxy for $h(x + \delta) \neq y$). This framework was developed by [Carlini and Wagner, 2017b], who use $p = 2$ for their eponymous C&W attack.

Note that this frameworks does not ensures that $\|\delta\|_p \leq \epsilon$. This can be compensated by adding a projection on $B_p(x, \epsilon)$ at the last step, although in practice that is never necessary, as the minimization problem leads to very small values of $\|\delta\|_p$.

Generative Adversarial networks A more recent approach to design adversarial example attacks is the use of GANs ([Xiao et al., 2018]). These networks are designed to generate synthetic samples, under the hypothesis that artificial data is good if it is hard to distinguish from real data. The core idea is to train two networks in parallel :

- **The Discriminator** performs a standard classification task, by separating real data points from fake ones.
- **The Generator** inputs some noise, and outputs a sample. It is evaluated by the response of the Discriminator, performing well if the Discriminator cannot distinguish the output sample from a real one.

This leads to a zero-sum game between both algorithms, and the objective functions are designed so that a stable state (or Nash equilibria, as we will define later) exists, and the learning procedure converges to it. In the case of Adversarial example attacks, the

generator takes as input the original image instead of noise, and the discriminator simply evaluates if the attack induces a misclassification or not from the classifier h.

2.2.3 The adversarial classification problem

The standard classification problem from Equation Equation (RM) thus becomes :

Definition 11 (Adversarial classification problem)**.** *Given a ground-truth distribution $\mathcal{D}$ over $(\mathcal{X}, \mathcal{Y})$, the adversarial classification problem is defined by the Adversarial Risk Minimization (ARM), i.e. :*

$$\inf_{h \in \mathcal{H}} \mathcal{R}_L^{adv}(h) \tag{ARM}$$

where $\mathcal{R}_L^{adv}(h) := \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \left[\sup_{z \in B_p(x,\epsilon)} L(h(z), y) \right]$

This means that the classifier is always evaluating the worst-case scenario, with the attack that causes the bigger increase in the loss. Equation (ARM) is the framework most commonly used in the literature ([Pydi and Jog, 2020a], [Diochnos et al., 2018], [Meunier et al., 2021]), but lacks flexibility, as it does not encompasses how most real-world attacker follow different, usually relaxed constraints than $\|\delta\| \leq \epsilon$, as we saw in the previous subsection. In chapter Chapter 3, we will introduce a more general framework to study the adversarial classification problem, that take all of these alternative constraints into account.

2.2.4 Hypothesis on the origins of attacks

The explanation of adversarial examples remains an open problem, although several different explanations have been offered :

- **Overfitting** [Szegedy et al., 2014], when introducing the first examples of adversarial attacks, postulated that this comes from current models being over-parametrized. Deep neural networks can approximate almost any function, and so it would appear logical that some overfitting happens, which is just not captured by our current testing sets. Recent results ([Nakkiran et al., 2021]) have further shown the existence of a double descent phenomenon for most deep learning algorithms (including adversarial training), where the generalization capabilities of the model first decreases then increases when the sample size grows.

- **Local linearity** [Goodfellow et al., 2015] rejected that hypothesis, claiming that the transferability of attacks makes it impossible for these attacks to be an artifact of training, and so of the noise in the training set. They postulate that the origin of adversarial examples lies in the locally linear behavior of neural networks, showing that in large dimension linear models always suffer from these kind of perturbations.
- **Complexity of the image classes** [Shafahi et al., 2018] analyze whether the dimensionality of the dataset contributes to the prevalence of adversarial examples. They conclude that there is no direct relation between the dimension of the problem and adversarial susceptibility, but that high-complexity image classes are way harder to defend using adversarial training, hinting at this complexity as a potential origin of adversarial examples.
- **Shape of the decision boundary** [Moosavi-Dezfooli et al., 2019] claim the opposite hypothesis : they show that several defense mechanism (such as adversarial training) lead to a local smoothing of the decision boundary, and especially a smaller curvature. This would indicate that adversarial examples stem from high local curvature. Our own results (see Chapter 5) point toward this hypothesis as well, as we show the importance of the local curvature on randomized smoothing certifications, and that two known robustness-inducing mechanism, namely noise injection and adversarial training, reduce the local curvature of the decision boundary.

2.3 State of the art defenses

At the moment, no classifier achieves an industry-enabling performance against adversarial attacks. In this section, we will give an overview of the current defense mechanisms, as well as their limitations. They can be divided into two main categories :

- **Empirical defenses** are designed by trial and error, and evaluated empirically against real-world attacks;
- **Provable defenses** rely on a mathematical framework, and offer some guarantees of robustness.

2.3.1 Adversarial training

One of the first defense mechanisms was introduced by [Goodfellow et al., 2015]. As adversarial examples show a problem of generalization, and so a dissonance between the empirical distribution and the underlying ground truth, the natural solution is to introduce some adversarial examples in the training set. This procedure was initially very

costly (since every adversarial attack needs to be crafted individually), but [Shafahi et al., 2019] showed that it is possible to drastically reduce that cost by re-using the gradient computations efficiently.

Figure 2.5 illustrates this procedure. After some steps of pre-training (so that the algorithm is efficient enough), an attack method (usually PGD, as it can be cheap to compute) is used to create perturbations, which are then added to the training set for the next iteration. Although lacking any guarantee, adversarial training has proven to be one of the most effective defense mechanisms yet.

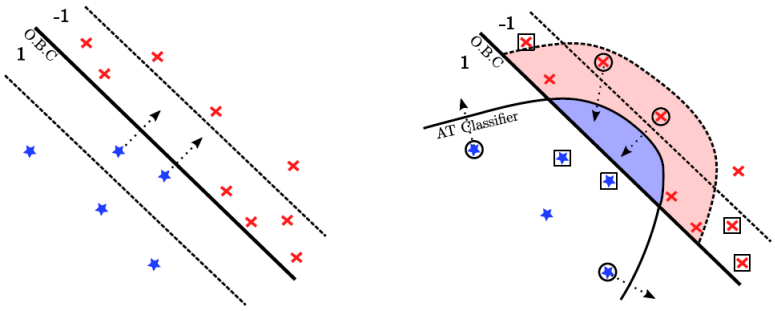


Figure 2.5: Illustration of adversarial training on a linear classifier. The attacks of the two blue stars in the middle are added to the training set, with the consequence of "curving" the decision boundary.

2.3.2 Adaptive attacks and the need for provable defenses

As the performance of empirical defenses seemed to grow, several voices began to challenge their evaluation model, and proceed to defeat existing defenses with *adaptive attacks*, which are designed specifically against the defense mechanism they target. For example, [Carlini and Wagner, 2017a] show that the C&W attack, when tweaked, allows to defeat several detection-based defenses, and [Athalye et al., 2018] show how obfuscated gradients can be defeated by specific methods. This lead the community to use better evaluation frameworks for empirical defenses, creating adaptive attacks against their own classifiers.

However, even with better evaluations, most efficient defenses ended up being defeated by stronger, more specific attacks at some points. [Tramer et al., 2020] provided a methodology for designing arbitrarily strong adaptive attacks, and proceeded to demonstrte it by defeating ten of the strongest defenses at that time, which were previously evaluated using adaptive attacks.

This leads to the following question :

Why is it so hard to obtain a defense that performs well against every attack ?

Two main hypotheses can be formulated to answer that question :

H1: *A perfect classifier exists, and the current situation is simply a war of processing power between the attack and defense, for who can compute the better approximation;*

H2 *There is no universally perfect classifier, and defenses will always be specific to some category of attacks.*

We will provide insights on this question in this thesis, and show that when no randomization is involved, **H2** is the correct hypothesis (see Section 2.4)

2.3.3 Provable defenses

The limitations of empirical defenses and the difficulty of providing definitive evaluations of defense mechanisms have created the need for provable guarantees of robustness, that leave no doubt on the efficiency of a method. Several frameworks have been developed to provide such guarantees:

- [Wong and Kolter, 2018] introduced the notion of Convex Outer Adversarial Polytops : instead of computing the set of all possible adversarial attacks (which is highly non-convex), they focus on a convex regularization of that set, which is easier to analyze through linear programming, and allows to derive some bounds on the adversarial risk (see Figure 2.6). This approach however perform decreasingly well when the dimension of the problem increases, due to the poor scaling of linear programming.
- Inspired by differential privacy, [Lecuyer et al., 2018] proposed an empirical method for increasing the robustness of a classifier, using a local averaging under some noise distribution. This was later made into a provable defense by [Cohen et al., 2019], which showed how to obtain computable certificates of robustness using the Neyman-Pearson Lemma. We will give a more thorough analysis of the state of the art for Randomized smoothing in Chapter 4.

2.4 Introduction to game theory

Game theory is the study of how agents make rational decisions depending on the behavior of others. Its purpose is to identify how order (in the form of stable states) can emerge when each agent (or player) only cares for its own interest. In this section, we will give a brief introduction to the field of adversarial game theory.

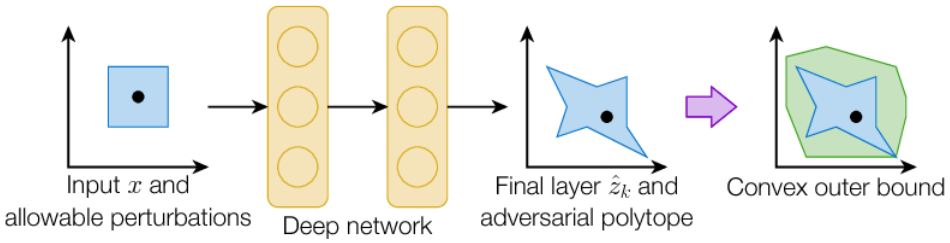


Figure 2.6: Illustration of the Convex Outer Adversarial Polytope

	N	D
N	$(1, 1)$	$(-2, 5)$
D	$(5, -2)$	$(0, 0)$

Table 2.1: The prisoner's dilemma. Each player can either Defect or Not. If no one defects, they both get mild sentences, but if only one of them does, it will go free while the other will have a heavy prison time. If both players defects, they get medium sentences. In this game, defecting is a stricly dominant strategy, since it always reduces the prison time, whatever the other player plays. It follows that (D, D) is the only Nash equilibrium, i.e. "stable" outcome.

2.4.1 Two player strategic game

Definition 12 (Two player strategic game, ([Laraki et al., 2019])). *A game in strategic form is a triple (I,J,g), where I and J are the non-empty set of (pure) strategies for each player, and* $g = (g_1, g_2)$ *where* $g_k : (I \times J) \to \mathbb{R}$ *is the payoff function of player* $k \in \{1, 2\}$. *If we have* $g_1 = -g_{-1}$, *we say that the game is zero-sum.*

The idea is that player 1 chooses some strategy in I, without knowing the behavior of player 2 which chooses a strategy in J. When both players have revealed their strategy, a payoff is computed for each player. We usually summarize games in a table, such as Table 2.1. The columns represent the possible strategies of player 2, and the rows of player 1. The payoffs are written as a pair $(g_1(i, j), g_2(i, j))$

We want to understand what outcomes will emerge when both players are rational. A first tool to study that is *dominant strategies*.

Definition 13 (Dominant strategy). *A strategy $i_0 \in I$ is called* dominant *for player 1 if we have :*

$$\forall i \neq i_0, \forall j \in J, g_1(i,j) < g_1(i_0,j) \tag{2.13}$$

A similar definition can be given for player 2 with g_2 instead of g_1.

Intuitively, if a dominant strategy exists for player i, he will always play it. In the Prisoner's Dilemma game, the strategy D is dominant for both players, so that we feel that the outcome (D, D) is the only "rational one". More generally, we consider an outcome to be rational if every player plays the best possible move, according to its anticipations of the other player's actions.

Definition 14 (Best responses). *A strategy i_1 of player 1 is a best response to strategy j_1 of player 2 if for any other strategy i_2 of player 1, we have $g_1(i_2, j_1) \leq g_1(i_1, j_1)$. A symmetric definition holds for player 2's best responses.*

Definition 15. *A Nash Equilibria is a pair of strategies $(i^*, j^*) \in I \times j$ such that for any other strategies $i \in I$, $j \in J$, we have:*

$$\begin{cases} g_1(i, j^*) & \leq g_1(i^*, j^*) \\ g_2(i^*, j) & \leq g_2(i^*, j^*) \end{cases}$$

In other words, a state where no player has an incentive to modify its behavior, since each one plays the best response to the other's strategy.

A pair of dominant strategies will always be a Nash equilibrium, but the converse is not necessarily true. In fact, many games have no dominant strategies. For example in the game in Table 2.2, there is no dominant strategy for either player. However, (V, C) is a Nash equilibrium, which means that **if each player anticipates that the other is playing the equilibrium, its most rational move is to play it as well**.

2.4.2 Zero-sum games and the minimax problem

For zero-sum games, we can study Nash equilibria in terms of the *guaranteed payoffs* of each player. In this setting, we will call g the payoff of player 1, and $-g$ the payoff of player 2 (as the game is zero-sum). Note that player 1 will try to maximize g, whereas player 2 will want to minimize it. We can then define :

	A	B
U	$(3, 0)$	$(0, 1)$
V	$(0, 0)$	$(3, 1)$
W	$(1, 1)$	$(1, 0)$

Table 2.2: A game with no dominant strategy, but a Nash equilibrium (V, B)

Definition 16 (Guaranteed payoffs)**.** *Player* 1 *guarantees* $w \in \mathbb{R} \cup \{-\infty\}$ *if there exists a strategy* i_0 *that induces a payoff at least* w *whatever the other players plays, i.e.*

$$\forall j \in J, g(i_0, j) \geq w$$

Symetrically, player 2 *guarantees* $w \in \mathbb{R} \cup \{+\infty\}$ *if there is* $j_0 \in J$ *such that :*

$$\forall i \in I, g(i, j_0) \leq w$$

A quantity of interest will be the best guaranteed payoff for every player. For every strategy $j \in J$, player 1 can choose the strategy that gives the best payoff against j, and thus guarantee $\inf_{i \in I} g(i, j)$. In the same way, player 2 can guarantee $\sup_{j \in J} g(i_0, j)$ for every strategy i_0 of player 1. Hence :

Definition 17 (minmax and maxmin)**.** *We define the maxmin* $\underline{v}$ *and the minmax* $\overline{v}$ *as respectively the highest payoff that player* 1 *can guarantee, and the lowest payoff that player* 2 *can guarantee.*

$$\underline{v} = \sup_{i \in I} \inf_{j \in J} g(i, j) \in \mathbb{R} \cup \{\pm\infty\}$$
$$\overline{v} = \inf_{j \in J} \sup_{i \in I} g(i, j) \in \mathbb{R} \cup \{\pm\infty\}$$

We can see this two values as worst-case scenarios for each player : $\overline{v}$ happens when player 1 always plays the best response to player 2's strategy (for example if he plays in second, and has already seen the strategy), and $\underline{v}$ happens when player 2 always plays the best response to player 1's strategy. It follows that :

	R	P	S
R	0	-1	1
P	1	0	-1
S	-1	1	0

Table 2.3: Rock-Paper-Scissors. 1 means a win of player 1, -1 of player 2, and 0 a draw.

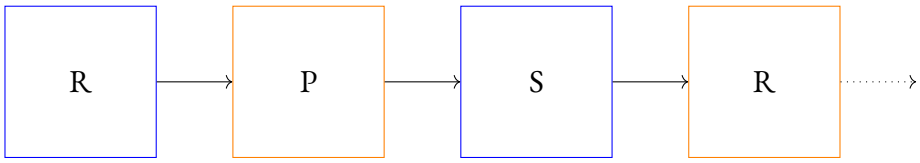


Figure 2.7: Successive strategies for Rock-Paper-Scissors. Blue nodes represent player 1, and orange nodes player 2.

Lemma 1 (Duality gap)**.**

$$\underline{v} \leq \overline{v}$$

When the two are equal, we say that the game has a *value*, which is the rational outcome that will emerge in the game. This corresponds, for zero-sum games, to the existence of a Nash equilibrium (see [Maschler et al., 2020]). Note that these two notions of equivalence do not necessarily coincide for non-zero sum games.

2.4.3 Anticipations, cyclicity and mixed Nash equilibria

However, some games have no Nash equilibrium in pure strategies, and instead exhibit some form of cyclicity. The most classical example is the game Rock-Paper-Scissors:

What will happen in this game ? If player 1 plays rock, then player 2 has every incentive to play Paper. But then, player 1 will want to change its strategy and play Scissors, which in turns convinces player 2 to play Rock, and we are back to the beginning of the cycle (see Figure 2.7).

What will then happen ? One possible solution is that the players can anticipate probabilistic behaviors. For example, player 1 can expect that player 2 will play Rock $\frac{1}{3}$ of the time, Paper $\frac{1}{3}$ and Scissors $\frac{1}{3}$. In their anticipation, the other's player strategy is thus randomized. Furthermore, they can themselves decide to rely on randomness, by for example flipping a coin and deciding on a strategy depending on the outcome. In both cases, we speak of *mixed strategies*.

Definition 18 (Mixed Strategy)**.** *In the game (I, J, g) we say that the set $\Delta(I)$ is the set of all possible mixed strategies for player 1, and $\Delta(J)$ is the set of mixed strategies for player 2, where $\Delta(S)$ represents the convex extension (or simplex) of any set S.*

For example, a strategy for Rock-Paper-Scissors could be to play Rock $\frac{2}{3}$ of the time, Paper $\frac{1}{6}$ of the time and Scissors $\frac{1}{6}$. When the set of pure strategies for a player is finite, then the set of mixed strategies is constituted of *mixtures*, i.e. combinations of Dirac measures.

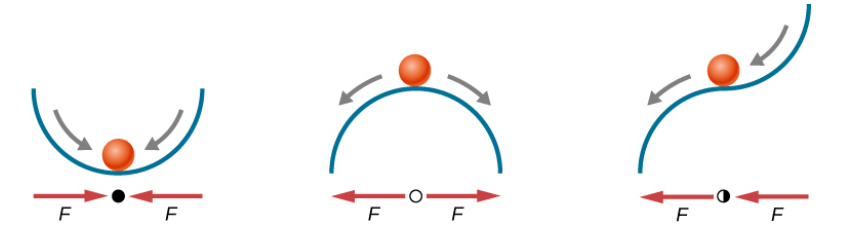
Mixed Nash equilibria Introducing mixed strategies amounts to a convex relaxation of the problem. As that makes both player stronger, the duality gap can only be smaller than in the deterministic case. For finite zero-sum games, this is enough to ensure the existence of a value, and thus of a Nash equilibrium.

Theorem 2 (Minimax theorem [Neumann, 1928])**.** *The mixed extension of a finite two-player zero-sum game always has a value, and thus a mixed Nash equilibrium.*

In the case of Rock-Paper-Scissors, the only Nash equilibrium happens when both player play the purely random strategy $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, giving an expected payoff of 0. Note that for infinite zero-sum game, this theorem stops being valid, which is why we needed to develop different tools to analyze Nash Equilibria in Chapter 3.

2.4.4 Stability of Nash Equilibria

For finite games, the notion of Nash equilibria is enough to satisfyingly capture the mechanism of stability. However, for continuous games, such as the one we will study in Chapter 3, players usually can't compute their strategy exactly, but have access to some confidence interval around it. For example, if both players play a real number, computers will only be able to approximate that number, and infinitesimal variations can easily happen as an artifact of the computation. It follows that any realistic notion of equilibrium in a continuous setting must resist to asymptotical variations ([Van Damme, 1991]).



(a) Stable equilibrium point (b) Unstable equilibrium point (c) Unstable equilibrium point

Figure 2.8: Illustration of the stability of equilibrium via a physics metaphor. A Nash equilibrium ensures that no player has an interest to deviate, but does not tell anything about what happens if some deviation happens by accident. For stable equilibria, the perturbation has either no effect, or induces an incentive to go back to the Equilibrium.

Definition 19 (Stable Nash equilibria)**.** *Let* (i^*, j^*) *be some Nash equilibrium in a continuous game, and* d *be some distance over the space of player* 1*'s strategies. It is said to be* $\delta-$*stable if for any* $i \in I$ *such that* $d(i, i^*) \leq \delta$*, we have the two following properties :*

- $g(i, j^*) \leq g(i^*, j^*)$*;*
- $g(i, j^*) = \inf_{j \in J} g(i, j)$

In other words, player 1 *has won nothing by deviating infinitesimally, and player* 2 *has no incentive to change its strategy after the small perturbation of his opponent's strategy. If the first inequality is strict, the equilibrium will be strictly stable, and deviations from the equilibrium will converge back to it.*

This can be interpreted as follow : Once a standard Nash equilibrium has been reached, no player has an incentive to deviate from it. But if, by some computational instability, one of the player ends up deviating, what happens ? When the equilibrium is stable, the other player won't react, and the new state reached will also be a Nash equilibrium. In a sense, it is more of an "equilbrium region", than an "equilibrium point". On the other hand, if the equilibrium is unstable, then some of the deviations may result in a change in the other player's strategy, and both player go back to the cat-and-mouse of best responses, potentially snowballing away from the equilibrium.

2.5 Optimal Transport and Kantorovich duality

Now that we have a framework to define equilibria, and the tools of statistical learning theory to study the Defender's problem, we need a way to analyze the Attacker's. His goal is to modify the two conditional distributions $\mathbb{P}[X|Y=1]$ and $\mathbb{P}[X|Y=-1]$ to make them as similar as possible, thus creating "indecision zones", where both classes have the same likelihood and the classifier "cannot make a choice". This is similar to the classical problem of Optimal Transport, which has been widely studied. We will present here an introduction to that field, that largely relies on the work of [Villani, 2009].

2.5.1 Motivating example : moving sand

In his "Traité sur les déblais et remblais" (1781), Monge introduced the following problems : given a pile of sand and a whole, what is the most economic way to fill the whole using the sand of the pile ? In this formulation, the cost induced by moving a unit of mass between two points was the euclidean distance between both points. More formally, when we consider the *relative height/depth* of the sand, this can be formulated in terms of measure theory as follows :

Definition 20 (Monge Optimal Transport problem)**.** *Let μ and ν be probability measures on $\mathcal{X}$, and $c : \mathcal{X} \times \mathcal{X} \to [0, +\infty]$. The Monge Optimal Transport problem consists in finding a measurable application $T : \mathcal{X} \to \mathcal{X}$ that is a solution to the following minimization problem :*

$$\inf_T \int c(x, T(x)) d\mu(x)$$
$$\text{s.t. } T\#\mu = \nu$$

Where $T\#\mu$ is the pushforward measure of μ by T, defined by :

$$T\#\mu(C) = \mu(T^{-1}(C)) = \mu(\{x \in \mathcal{X} | T(x) \in C\})$$

We are interested in the existence and unicity of the solution, as well as whether such a solution can be computed in the general way. However, as such, the problem is very difficult to tackle, as well as ill posed for many distributions μ, ν. For example, if μ contains some Dirac distribution δ_{x_0}, and ν is an absolutely continuous probability distribution (such that no point has a non-zero mass), then there is no transport application T that can satisfy $T\#\mu = \nu$. This is because we are not able to split the mass from point x_0 to

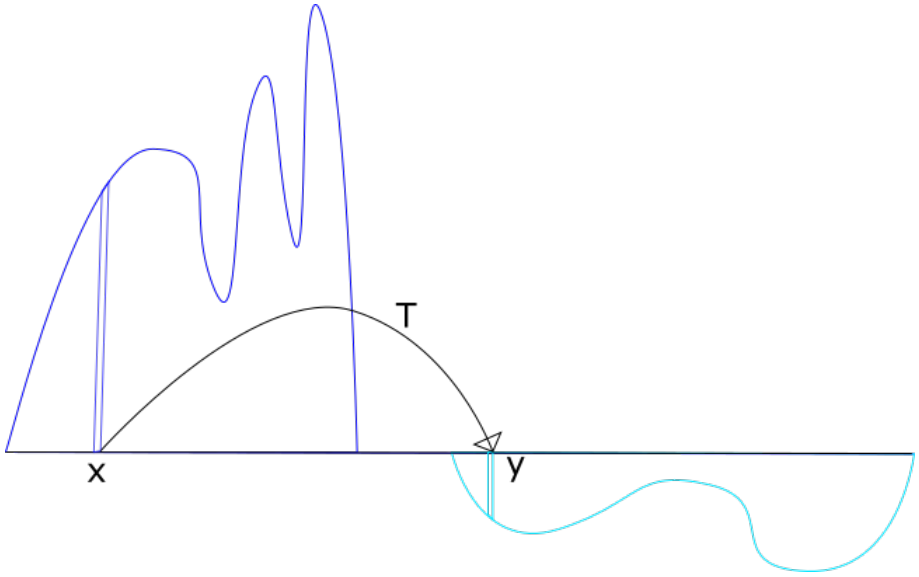


Figure 2.9: Illustration of the Monge Problem. The transport application T transports units of sand from μ to ν, but cannot "split" the mass. For example, here, the mass at point x can be wholly transported on y, but not divided between y and some other point.

distribute it over some region $A \subset \mathcal{X}$, so any distribution $T\#\mu$ would exhibit a Dirac on $T(x_0)$.

2.5.2 Breaking rocks : the Kantorovich relaxation

To solve that issue, we need a more general way of defining transport applications. This was done by Leonid Kantorovich in the 1940s, which gave the modern formulation of the Optimal Transport problem. Intuitively, we can "break" the mass at point x by allowing $T(x)$ to be a probability distribution over $\mathcal{X}$. This is formalized in the notion of Kernel function. Let $\mathcal{P}(\mathcal{X})$ denote the set of probability measures over $\mathcal{X}$.

Definition 21 (Kernel)**.** *A kernel on $\mathcal{X}$ is an application $p : \mathcal{X} \to \mathcal{P}(\mathcal{X})$, which associates to each input $x \in \mathcal{X}$ a probability distribution p_x over $\mathcal{X}$, such that for any measurable bounded function $f : \mathcal{X} \to \mathbb{R}$, the function $x \mapsto \int_{\mathcal{X}} f(u)p_x(du)$ is measurable.*

Although rarely used in practice, this formulation will be very useful to us to interpret couplings as randomized attacks later in Chapter 3, Chapter A. The equivalent of a pushforward measured can be defined as $\nu = \mu p$ where:

$$\mu p(C) = \int_{x \in \mathcal{X}} \int_{y \in C} p_x(dy) d\mu(x)$$

Kernels can be written in a symmetric ways as *couplings*, seing p_x as the conditional expectation on x for some joint probability distribution over $\mathcal{X} \times \mathcal{X}$.

Definition 22 (Coupling). *Let $\mu, \nu \in \mathcal{P}(\mathcal{X})$. A coupling between μ and ν is a probability distribution $\pi \in \mathcal{P}(\mathcal{X} \times \mathcal{X})$ such that :*

$$\begin{cases} proj_1 \# \pi = \mu \\ proj_2 \# \pi = \nu \end{cases}$$

where $proj_i \# \pi$ stands for the $i-th$ marginal of π. We therefore want the first one to corresponds to μ and the second to ν. We call $\Pi(\mu, \nu)$ the set of all couplings between μ and ν.

We can see that there is a one-to-one correspondance between couplings and kernels. A kernel naturally induces a coupling, and for the reverse :

Theorem 3 (Coupling-Kernel equivalence [Gozlan et al., 2018]). *Let $\pi \in \Pi(\mu, \nu)$. There is a kernel p, unique $\mu-$almost surely, such that for any bounded Borel-measurable function $f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, we have:*

$$\int f(x,y)\pi(dxdy) = \int \left(\int f(x,y) p_x(dy) \right) \mu(dx)$$

We can now define the Monge-Kantorovich problem in terms of couplings :

Definition 23 (The Monge-Kantorovich Optimal Transport problem)**.** *Let $\mu, \nu \in \mathcal{P}(\mathcal{X})$, and a measurable cost function $c : \mathcal{X} \times \mathcal{X} \to [0, +\infty]$. We are looking for a coupling $\pi \in \Pi(\mu, \nu)$ that is solution of the following problem :*

$$\inf_{\pi \in \Pi(\mu,\nu)} I_c(\pi) = \int c(x, y)\pi(dxdy)$$

We call the optimal transport cost between μ and ν the value $\mathcal{T}_c(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} I_c(\pi)$. A coupling π is called a transport plan between μ and ν.

2.5.3 The Kantorovich duality theorem

We can study the existence of optimal transport plans in an interesting way by casting that problem as a min-max one, and using a duality theorem such as Fenchel-Rockafeller. In this thesis, we will use reasonings that are very similar to the Kantorovich Duality theorem :

Theorem 4 (Kantorovich duality)**.** *Let $c : \mathcal{X} \times \mathcal{X} \to [0, +\infty[$ be a lower semi-continuous function, and $\mu, \nu \in \mathcal{P}(\mathcal{X})$ be such that $\mathcal{T}_c(\mu, \nu) < +\infty$. We then have :*

$$\mathcal{T}_c(\mu, \nu) = \sup_{(\phi,\psi)\in\Phi_c} \left\{ \int \psi(x)\mu(dx) + \int \phi(y)\nu(dy) \right\}$$

Where Φ_c is the set of all continuous bounded functions such that $\psi(x) + \phi(y) \leq c(x, y)$. Furthermore, there is an optimal transport plan that attains the infimum in $\mathcal{T}_c(\mu, \nu)$.

Theorem 4 relies on a powerful min-max result, which is called the Fenchel-Rockafeller theorem. To state it, we first need to define the Fenchel-Legendre transform, which corresponds to the highest linear function that is lower than a given function.

Definition 24 (Fenchel-Legendre transform)**.** *Let θ be a convex function on a normed vector space E, with values in $\mathbb{R} \cup \{+\infty\}$. The Legendre-Transform is the function θ^*, defined on the topological dual E^* of E, defined by :*

$$\theta^*(z^*) = \sup_{z\in E} [\langle z^*|z\rangle - \theta(z)] \tag{2.14}$$

Theorem 5 (Fenchel-Rockafeller theorem [Villani, 2021]). *Let X, Y be Banach spaces, and $F : X \to \mathbb{R} \cup \{+\infty\}$ and $G : Y \to \mathbb{R} \cup \{+\infty\}$. Let A be some bounded linear operator $X \to Y$. Then if $0 \in relint(domG - AdomF)$, then we have :*

$$\inf_{x \in X} (f(x) + g(Ax)) = - \inf_{\phi \in Y} (F^*(A^*\phi) + G^*(-\phi^*)) \tag{2.15}$$

And the right-hand side infimum is attained if it is finite.

We will use this theorem in Chapter 3 Chapter A to show the existence of Mixed Nash equilibria in particular situations.
In the next section, we will summarize the existing works on the game-theory or min-max perspective on adversarial examples, as well as position our results with regards to them.

2.6 Saddlepoint analysis of the adversarial classification problem, and our relative positioning

2.6.1 Game theory analysis of the problem

Additive perturbations To our knowledge, the first formulation of the adversarial example problem as a two-player zero-sum game was in [Pal and Vidal, 2020]. Their framework is however very different from ours : they consider that the classifier h is fixed, and that both the attack and the defense are additive perturbations $a(x)$ and $d(x)$ for every point x. They evaluate a and d by checking whether $x + a(x) + d(x)$ is classified differently than x by h_L, which is the locally linear approximation of h.

$$g(x, a(x), d(x)) = \begin{cases} 1 \text{ if } \operatorname{sign}(h_L(x)) \neq \operatorname{sign}(h_L(x + a(x) + d(x))) \\ -1 \text{ otherwise.} \end{cases}$$

Using this framework, they show the existence of a Nash equilibrium between the fast gradient-sign attack, and Randomized smoothing as a defense. However, the margin of the Defender is extremely tight compared to most papers in the field, as it only authorizes small-size perturbations from the Bayes classifier, which has no theoretical justification. We can view this as a form of correlated equilibrium, when both players agreee to restrict the defender's strategy accordingly.

Our framework With the paper [Pinot et al., 2020], we introduced the game theory formulation of the problem that is currently used by the literature, and showed the non-existence of Nash equilibria in the deterministic regime. We will describe these results in more details in Chapter 3.

Mixed Nash equilibria This framework was then extended by [Meunier et al., 2021], who use the approach of Distributionally robust optimization. They authorized both the defender and the attacker to use very general forms of randomization : the attacker plays a transport plane instead of a Monge pushforward measure, and the defender plays a Borel probability measure on $(\mathcal{X} \times \mathcal{Y})$. In this setting, they show the existence of mixed Nash equilibria, and how it can be approached using an entropic relaxation of the attacker's optimal transport problem.

Their approach differ from our analysis in Chapter A in two main ways :

- They use a much stronger relaxation of the defender's space of strategies, which is not necessary to obtain an equilibrium with convex, surrogate losses;
- They consider the case of the $0/1$ losses, whereas we focus on convex surrogates. Hence, those two approaches are complementary.

2.6.2 Analyzing the adversarial risk via optimal transport

One of the main challenges in the study of adversarial robustness is to derive bounds on the risk under attack, to quantify how much incompressible loss these attacks will incur, depending on the distribution. [Pydi and Jog, 2020a] (and [Bhagoji et al., 2019], although with a smaller hypothesis class) provide a very elegant analysis of the adversarial risk through optimal transport, leveraging Strassen's theorem to formulate the optimal adversarial risk as the optimal transport distance between the two conditional measures:

Definition 25 (ϵ-transport cost)**.** *For $\epsilon > 0$, d some distance over $\mathcal{X}$. We define the cost function $c_\epsilon : (x, z) \mapsto \mathbb{1}_{\{d(x,y)>2\epsilon\}}$, and the optimal transport cost D_ϵ by :*

$$D_\epsilon(\mu, \nu) := \inf_{\pi \in \Pi(\mu,\nu)} \mathbb{E}_{(x,z)\sim\pi} c_\epsilon(x, z) \tag{2.16}$$

This corresponds to the minimal average number of points that need to be moved by more than ϵ for each distribution toward the decision boundary (or equivalently from 2ϵ for only one of the distributions, the other staying in place), to make the two distributions equal. For $\epsilon = 0$, this corresponds to the total variation distance between μ and ν.

Theorem 6 (Adversarial risk via optimal transport, [Pydi and Jog, 2020b]). *Let μ_1, μ_{-1} be the distributions from $\mathcal{D}$, conditional to $y = 1$ and $y = -1$ respectively. We consider the class of binary classifiers of the form $\mathbb{1}_A$ where $A \subset \mathcal{X}$ is a closed set. Then the optimal adversarial risk is given by :*

$$\inf_{h \in \mathcal{H}_1} \mathcal{R}^{adv}_{L_{0/1}}(h) = \frac{1}{2}[1 - D_\epsilon(\mu_1, \mu_{-1})] \tag{2.17}$$

(Recall that $\mathcal{R}^{adv}_L(h) := \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\sup_{z \in B_p(x,\epsilon)} L(h(z), y)\right]$*)*

This is very closely related to our work, as Strassen's theorem is a particular case of the Kantorowitch duality for $0/1$ cost functions. Reading these papers was a huge influence on our work, and motivated us to take the optimal transport approach.

In our Chapter 3, Chapter A, we also cast the attacker's problem as an optimal transport. However, we make the couplings explicit in our approach, since we consider the existence of an optimal strategy, and not just the value of the optimal risk. Furthermore, we work with more general transport costs (for which Equation (2.16) is a particular case), and convex surrogate losses whereas [Pydi and Jog, 2020a] use the $0/1$.

2.7 Our positioning relative to previous papers

The novelty of our contributions in Chapter 3 can be summarized as follows:

- We cast the adversarial classification problem as a two-player zero-sum game, and show that under most realistic constraints for the attacker, no pure Nash equilibrium can exist in the deterministic setting;
- We introduce the notion of stability of equilibria, to study their realizability in practical settings, and show that no pure Nash equilibrium can be stable in the deterministic setting;
- We study the importance of randomization for the Defender, showing that noise injection increases the stability of equilibria, and that a boosting-like process allows to outperform any deterministic classifier;
- We show that allowing for general randomized attacks leads to Nash equilibria when using convex surrogate losses, and give some conditions for the existence of optimal deterministic attacks.

3 Studying Nash equilibria via Optimal Transport

Contents

In this chapter, we will tackle the first major question of this thesis, namely the existence of a "optimal" classifier, that would provide the best performance under any given attack. The problem of adversarial risk minimization has a natural formulation as a two-players zero-sum game. On one side, **the Defender** tries to find the best classifier under a given attack, which amounts to a standard classification problem. On the other side, **the Attacker** tries to perturbate the input distribution to maximize the probability of error. This is a form of optimal transport problem, as we will see in Section 3.1.2.

In this framework, **Q1** becomes the larger question of the existence of a Nash equilibrium, a stable state of the game where no player has an incentive to modify its behavior.

Q1: *Is it possible to design a classifier that performs optimally under any attack ?*

Q1bis: *Will the study of adversarial attacks and defenses reach a stable state ?*

3.1 A zero-sum game of attacks and defenses

3.1.1 The Defender : a robust classification problem

Binary classification task. Let $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} = \{-1, 1\}$. Let L be some loss function. We consider a distribution $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$ that we assume to be of support $\mathcal{X} \times \mathcal{Y}$. The Defender is looking for a hypothesis (classifier) h in some class of functions $\mathcal{H}$, minimizing the L-risk of h w.r.t. $\mathcal{D}$:

$$\begin{aligned} \mathcal{R}(h) :&= \mathop{\mathbb{E}}_{(X,Y)\sim\mathcal{D}}[L(h(X), Y)] \\ &= \mathop{\mathbb{E}}_{Y\sim\nu}\left[\mathop{\mathbb{E}}_{X\sim\mu_Y}[L(h(X), Y)]\right] \\ &= \sum_{i=\pm 1} q_i \int_{\mathcal{X}\times\mathcal{X}} L_i(h(x))d\mu_i(x) \end{aligned} \tag{3.1}$$

where $q \in \mathcal{P}(\mathcal{Y})$ is the probability measure that defines the law of the random variable Y, and for any $y \in \mathcal{Y}$, $L_y = L(\cdot, y)$ and $\mu_y \in \mathcal{P}(\mathcal{X})$ is the conditional law of $X|(Y = y)$.

Prior beliefs on robust classification As we stated in Chapter 2, the choice of the hypothesis set $\mathcal{H}$ is of prime importance for the generalization capabilities of the classifier, and usually reflects our prior beliefs on how "good" solutions should behave. For robust classification, a necessary condition is that the classifier does not vary too much in the neighborhood of a point, which is often referred to in the literature as a *smoothness prior* ([Goodfellow et al., 2016]).

In this thesis, we will thus consider $\mathcal{H}$ to be the set of all continuous functions. Later, we will investigate an even stronger prior, namely the *Lipschitz prior* (Appendix A).

3.1.2 The Attacker : an optimal transport problem

On the other side, given a classifier $h : \mathcal{X} \to \mathcal{Y}$, the Adversary seeks, for every data sample $(x, y) \sim \mathcal{D}$, a perturbation $\tau \in \mathcal{X}$ that modifies x enough to change its class, *i.e.* $h(x + \tau) \neq y$. This amounts to constructing, for each label $y \in \mathcal{Y}$, a measurable function ϕ_y (called a *transport map*) such that $\phi_y(x)$ is the perturbation associated with the labeled example (x, y). This function naturally induces a probability distribution over adversarial examples, which is simply the push-forward measure $\phi_y \# \mu_y$.
After attack, the probability of misclassification is now :

$$\mathcal{R}_{\text{adv}}(h, \boldsymbol{\phi}) := \mathop{\mathbb{E}}_{Y \sim \nu} \left[\mathop{\mathbb{E}}_{X \sim \phi_Y \# \mu_Y} [L(h(X), Y)] \right]. \tag{3.2}$$

On top of seeking misclassification, the Attacker follows some constraints. The most natural one being *imperceptibility*: the size of the perturbation must be small enough to be invisible for humans. We formulate these in a very general way as *constraint functions* $\Omega(\boldsymbol{\phi})$, that can be any function of the attack.

Let $\mathcal{F}_{\mathcal{X}}$ be the set of all measurable functions from $\mathcal{X}$ to $\mathcal{X}$. The goal of the Adversary is thus to find $\boldsymbol{\phi} = (\phi_{-1}, \phi_1) \in (\mathcal{F}_{\mathcal{X}})^2$ that maximizes the adversarial risk $\mathcal{R}_{\Omega}(h, \boldsymbol{\phi})$ defined as follows:

$$\mathcal{R}_{\Omega}(h, \boldsymbol{\phi}) := \mathcal{R}_{\text{adv}}(h, \boldsymbol{\phi}) - \Omega(\boldsymbol{\phi}) \tag{3.3}$$

As we will see in the following of this thesis, realistic constraints are often generated by some pointwise function, in which case we call them transport-based :

Definition 26 (Transport-based constraint)**.** *We say that Ω_c is a transport-based constraint if there exists a measurable function $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+ \cup \{\infty\}$, that is lower semicontinuous, such that for every $x \in \mathcal{X}, c(x, x) = 0$, and so that :*

$$\forall \boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2, \Omega_c(\phi) = \mathop{\mathbb{E}}_{Y \sim \nu} \left[\mathop{\mathbb{E}}_{X \sim \mu_Y} [c(X, \phi_Y(X)] \right]$$

We call c the transport cost associated to Ω_c.

For any input x and some point z, $c(x, z)$ represents the cost of moving x to z for the Attacker. In the case of transport-based constraint, the Attacker's problem now becomes very similar to an optimal transport problem :

$$\mathcal{R}_{\Omega_c}(h, \boldsymbol{\phi}) = \sum_{i=\pm 1} q_i \int_{\mathcal{X} \times \mathcal{X}} [L_i(h(\phi_i(x))) - c(x, \phi_i(x))] d\mu_i(x) \tag{3.4}$$

3.1.3 Modeling cost functions for realistic adversaries

Definition 27 (Indicator cost)**.** *Let ϵ be some threshold for the human vision. The most natural way to represent the constraint of imperciptibility is with a cost of the form:*

$$c_\epsilon(x, z) = \begin{cases} 0 & \text{if } \|x - z\| \leq \epsilon \\ +\infty & \text{otherwise} \end{cases} \tag{3.5}$$

Note that this transport cost is lower semicontinuous (as an indicator function). Furthermore, such a cost strictly forbids any attack of norm greater than ϵ. As we will see later in this chapter, there are other, more continuous ways to model the imperceptibility conditions, as well as additional constraints that realistic Attackers must fullfill.

Definition 28 (Imperceptibility-enforcing transport cost)**.** *Let $\epsilon > 0$, and k be some transport cost. The (ϵ, k) imperceptibility-enforcing transport cost is :*

$$\begin{aligned} c_{\epsilon,k}(x,z) &= c_\epsilon(x,z) + k(x,z) \\ &= \begin{cases} k(x,z) & \text{if } \|x-z\| \leq \epsilon \\ +\infty & \text{otherwise} \end{cases} \end{aligned}$$

This is the general family of cost functions we will study in most of this chapter. It encompasses two things :

- The imperceptibility constraint c_ϵ, which state that attacks must be of size at most ϵ or fail;
- Additional constraints k, penalizing the transport of x to z.

In the next sections, we will focus on *positive* costs, which means that moving a point (i.e. attacking) is never completely free for the Attacker, and always incurs some penalty, even infinitesimal. Non-positive costs are highly unrealistic, considering computational costs, discretion, and so on.

Definition 29. *A transport cost k is said to be positive if $\forall x \neq z \in \mathcal{X}, k(x,z) > 0$. A (ϵ, k)-imperceptibility enforcing constraint is said to be positive if the associated transport cost is positive.*

Here are two example of realistic costs for the Attacker, parametrized by a strength parameter $\lambda > 0$:

Example 1 (The Mass cost)**.** *From a computer-security point of view, the first limitation that comes to mind is to limit the number of queries the Adversary can send to the classifier. This amounts to penalizing the mass of points that the function moves:*

$$k_{mass} = \lambda \mathbb{1}\{x \neq z\}$$

Example 2 (The Carlini-Wagner cost)**.** *This cost function is frequently used to compute attacks, since it is an easily-optimizable relaxation of the hard imperceptibility constraint. In particular, it is used by Carlini & Wagner [Carlini and Wagner, 2017b] to compute the eponymous attack.*

$$k_{CW}(x,z) = \lambda \|x-z\|_2^2$$

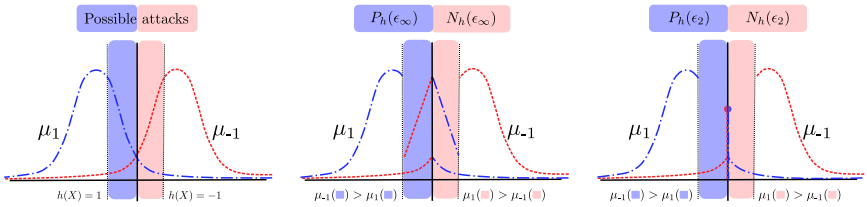


Figure 3.1: Illustration of the impact of cost-based constraints on the optimal attack. On the left, both conditional distributions along with the attackable zones in blue and red. In the middle, a possible optimal attack under the mass cost, where the attackable zone is pushed on the other side of the decision boundary while the rest of the distribution is keeped fixed. On the right, the only optimal attack under the Carlini& Wagner cost : a projection on the decision boundary to minimize the distance travelled.

Note that both of these cost functions are positive, and so induce positive imperceptibility-enforcing constraints. We will now use the definitions from this section to formulate the problem as a two-player zero-sum game.

3.1.4 Formulating the problem as a zero-sum game

Note that **for any fixed** ϕ, $\Omega(\phi)$ is independent of h. Hence, it is equivalent for the Defender to minimize $\mathcal{R}(h, \phi)$ or $\mathcal{R}(h, \phi) - \Omega(\phi)$ in h.

This means that the Defender's optimization problem is not affected by the constraint over the Attacker's strategy, so that we can add this term to the Defender's score without loss of generality. It follows that the game can be viewed as zero-sum, i.e. both player want to optimize the same score, but in opposite directions. The two players zero-sum game of adversarial example attacks with constraint Ω on the Attacker (that we will call the Ω-game) is :

Definition 30 (2-player zero-sum game of attacks and defenses)**.**

Defender problem:

$$\overline{v}(\Omega) = \inf_{h \in \mathcal{H}} \sup_{\phi \in (\mathcal{F}_\mathcal{X})^2} \mathcal{R}_\Omega(h, \phi). \tag{3.6}$$

Attacker problem:

$$\underline{v}(\Omega) = \sup_{\phi \in (\mathcal{F}_\mathcal{X})^2} \inf_{h \in \mathcal{H}} \mathcal{R}_\Omega(h, \phi). \tag{3.7}$$

The first problem corresponds to when the Defender plays first, so that the Attacker can adjust his strategy depending on the chosen classifier h. This means that the attack will always be a best response to h, i.e. $\arg\max_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_\Omega(h, \boldsymbol{\phi})$. In the case where the sup is not attained, the Attacker will be able to choose a strategy as close to it as it wants.

The second problem is when the Attacker plays first, which means that the Defender can play a best response against its strategy, namely $\arg\min_{h\in\mathcal{H}} \mathcal{R}_\Omega(h, \boldsymbol{\phi})$, or a classifier giving as close a risk as needed when the optimum is not attained.

When Ω is a transport-based constraint with cost c, we will call the game a c-game. Note that the values of both problems have no reason to coincide. In general, there exists what is called a *duality gap* between both.

Proposition 2 (duality gap)**.** *For any constraint function Ω such that both problems are well-defined, we have :*

$$\underline{v}(\Omega) \leq \overline{v}(\Omega)$$

i.e.

$$\sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \inf_{h\in\mathcal{H}} \mathcal{R}_\Omega(h, \boldsymbol{\phi}) \leq \inf_{h_1\in\mathcal{H}} \sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_\Omega(h_1, \boldsymbol{\phi})$$

The difference between both is called the duality gap, and is always non-negative.

Proof.

$$\begin{aligned}
&\forall h_1, \boldsymbol{\phi}_1, \inf_{h\in\mathcal{H}} \mathcal{R}_\Omega(h, \boldsymbol{\phi_1}) \leq \mathcal{R}_\Omega(h_1, \boldsymbol{\phi_1}) \\
\implies &\forall h_1, \sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \inf_{h\in\mathcal{H}} \mathcal{R}_\Omega(h, \boldsymbol{\phi}) \leq \sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_\Omega(h_1, \boldsymbol{\phi}) \\
\implies &\sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \inf_{h\in\mathcal{H}} \mathcal{R}_\Omega(h, \boldsymbol{\phi}) \leq \inf_{h_1\in\mathcal{H}} \sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_\Omega(h_1, \boldsymbol{\phi})
\end{aligned}$$

□

3.1.5 Strategies, best responses and Nash equilibrium

The choice of a classifier h for the Defender, as well as the choice of a pair of attack functions $\boldsymbol{\phi} = (\phi_1, \phi_{-1})$ for the Attacker, are what we call *pure (or deterministic) strategies* for the game. Both player play their strategy simultaneously, and without prior knowledge of what the other will do. If the game is played multiple time, or if players are allowed to modify their strategy anytime, then each of them will naturally adapt its behavior to what

the other player just played, and thus look for the best possible strategy to defeat that move.

Definition 31 (Best Response)**.** *Let $h \in \mathcal{H}$, and $\phi \in (\mathcal{F}_\mathcal{X})^2$ be a pair of strategies.*

- *A* best response *from the Defender to ϕ is a hypothesis $h^* \in \mathcal{H}$ such that*

$$\mathcal{R}_\Omega(h^*, \phi) = \min_{h \in \mathcal{H}} \mathcal{R}_\Omega(h, \phi).$$

- *Similarly, a* best response *from the adversary to h is an attack $\phi^* \in (\mathcal{F}_\mathcal{X})^2$ such that*

$$\mathcal{R}_\Omega(h, \phi^*) = \max_{\phi \in (\mathcal{F}_\mathcal{X})^2} \mathcal{R}_\Omega(h, \phi).$$

We write $h^* \in \mathcal{BR}(\phi)$ and $\phi^* \in \mathcal{BR}(h)$

The only way the game could reach a stable state is if no player has any incentive to modify its behavior after seeing the other's strategy, i.e. if both player play best responses to each other. We say that such a state is a pure (or deterministic) Nash Equilibrium.

Definition 32 (Pure Nash Equilibrium)**.** *In the zero-sum game (Eq. 3.6), a* pure Nash equilibrium *is a couple of strategies $(h, \phi) \in \mathcal{H} \times (\mathcal{F}_\mathcal{X})^2$ such that*

$$\begin{cases} h & \in \mathcal{BR}(\phi) \text{ and,} \\ \phi & \in \mathcal{BR}(h). \end{cases}$$

Remark 1. *All the definitions in this section assume that player play deterministic strategies – i.e. that neither the Defender nor the adversary use randomization – hence the notion of "Pure" Nash Equilibrium in the game theory terminology. Later in this thesis, we will consider the case of randomized strategies.*

3.1.6 0/1 loss and convex surrogates

Before diving into the mathematical results, it is important to discuss the choice of which loss functions to consider in this problem. The most natural choice is the $0/1$ loss : $L_{0/1}(h, y) = \mathbb{1}_{\text{sign}(h) \neq y}$. This amounts to counting the number of mistakes made by the classifier, and is the "true" score that both players wish to optimize.

This loss is however non-convex, and worse, optimizing it is an NP-hard problem. Hence, all practical implementations rely on convex *surrogate* loss functions, that exhibit

the property of consistency (see Definition 3), ensuring that optimizing them automatically leads to optimizing the $0/1$-loss in the non-adversarial case.

However, in the presence of attacks, both optimization problems are no longer equivalent. As we show in appendix Chapter B (see also [Meunier et al., 2022b]), convex surrogate losses cannot be adversarially consistent, so that optimizing them provide no guarantee for the $0/1$-loss. We are thus faced with two distinct problems, the theoretical one, representing what "should" happen if a consistent surrogate is found someday, and the practical one, that represents what currently happens in the machine learning field, using convex surrogate loss functions.

In the following thesis, we will study both problems, and show when Nash equilibrium can and cannot happen. We will then study the stability of these equilibria, to see if they can realistically occur in practical situations.

3.2 Study of the deterministic regime for the $0/1$ loss

In this section, we will study the deterministic regime of the game, and investigate the existence of pure Nash equilibria depending on the constraints on the Attacker. We will first characterize the Defender's best response to a given attack, which is the same for all Bayes Consistent losses, and does not depend on the cost function used. Then we will show that pure Nash equilibria can exist when the Attacker has no constraint outside of the imperceptibility, but show that these disappear as soon as the smallest, infinitesimal positive transport cost is added. We'll conclude the section with a discussion on what these non-existence results mean for the field.

Notations In this whole section, we consider the $0/1$ loss $L_{0/1}(h(x), y) := \mathbb{1}_{\text{sign}(h(x))\neq y}$. To simplify the notations, instead of having a classifier h and its sign, we will consider the class of functions $\mathcal{H} = \{h = \text{sign}(g), g \text{ continuous}\}$, so that the $0/1$ loss becomes $L_{0/1}(h(x), y) = \mathbb{1}\{h(x) \neq y\}$.

3.2.1 Defender's best response

At a first glance, one would suspect that the best response for the Defender ought to be the Optimal Bayes Classifier for the transported distribution. However, that is only well defined if the conditional distributions admit a probability density function. This might not always hold here for the transported distribution. Nevertheless, we show that there is a property, shared by the Optimal Bayes Classifier when defined, that always holds for the Defender's best response.

Lemma 2. *Let us consider $\boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2$. If we take $h \in \mathcal{BR}(\boldsymbol{\phi})$, then for any $y \in \{\pm 1\}$ and $B \subset C_y(h)$ of non-zero measure, one has*

$$\mathbb{P}(Y = y | X \in B) \geq \mathbb{P}(Y = -y | X \in B)$$

with $Y \sim \nu$ and for all $y \in \{\pm 1\}$, $X|(Y = y) \sim \phi_y \# \mu_y$.

Proof. First let us see that, from Baye's rules, we have :

$$\begin{aligned}\mathbb{P}(Y = y | X \in B) &= \frac{\mathbb{P}(X \in B | Y = y)\mathbb{P}(Y = y)}{\mathbb{P}(X \in B)} \\ &= \frac{\phi_y \# \mu_y(B) \times \nu_y}{\mathbb{P}(X \in B)}\end{aligned}$$

Since $\mathbb{P}(X \in B)$ is constant in y, we just need to show that for all $B \subset C_y(h)$ of non-zero measure, we have $\nu_{\text{-}y}\phi_{\text{-}y}\#\mu_{\text{-}y}(B) \leq \nu_y \phi_y \# \mu_y(C)$

For that, we reason ad absurdum. The main idea is that if there is a zone where class $-y$ is dominant, changing the value of the classifier on that zone would give a strictly better score, which is a contradiction since the classifier is assumed to be optimal.

Let us suppose that there exists $B \subset C_1(h)$ of non-zero measure such that $\nu_{\text{-}1}\phi_{\text{-}1}\#\mu_{\text{-}1}(B) > \nu_1 \phi_1 \# \mu_1(B)$ (a symmetric result holds with $B \subset C_{-1}(h)$ and inverting 1 and -1 in all computations). We can then construct h_1 as follows:

$$h_1(x) = \begin{cases} h(x) & \text{if } x \notin B \\ -1 & \text{otherwise.} \end{cases}$$

Since h and h_1 are identical outside B, the difference between the adversarial risks of h and h_1 writes as follows:

$$\begin{aligned}\mathcal{R}_\Omega(h, \boldsymbol{\phi}) - \mathcal{R}_\Omega(h_1, \boldsymbol{\phi}) &= \sum_{y=\pm 1} \nu_y \int_B \left(\mathbb{1}\{h(x) \neq y\} - \mathbb{1}\{h_1(x) \neq y\} \right) \, d(\phi_y \# \mu_y)(x) \\ &= \sum_{y=\pm 1} \nu_y \int_B \left(\mathbb{1}\{1 \neq y\} - \mathbb{1}\{-1 \neq y\} \right) \, d(\phi_y \# \mu_y)(x)\end{aligned}$$

Since $B \subset C_1(h)$ so that $h = 1$ over B. By definition, $h_1 = -1$ over B. It follows that :

$$\mathcal{R}_\Omega(h, \boldsymbol{\phi}) - \mathcal{R}_\Omega(h_1, \boldsymbol{\phi}) = q_{-1}\phi_{-1}\#\mu_{\text{-}1}(B) - \nu_1 \phi_1 \# \mu_1(B) \tag{3.8}$$

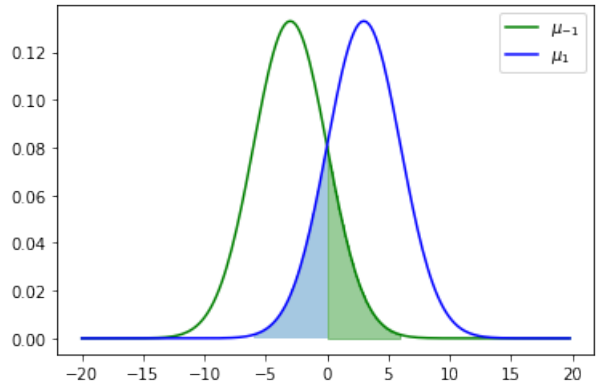


Figure 3.2: 2 normal conditional distributions. The blue zone represents the conditional risk of class 1, and the green zone the conditional risk of class -1.

Since by hypothesis $q_{-1}\phi_{-1}\#\mu_{-1}(C) > \nu_1\phi_1\#\mu_1(C)$, the difference between the adversarial risks of h and h_1 is strictly positive. This means that h_1 gives strictly lower adversarial risk than the best response h. Since, by definition h is supposed to be optimal, this leads to a contradiction. Hence Lemma 2 holds. □

In particular, when $\phi_1\#\mu_1$ and $\phi_{-1}\#\mu_{-1}$ admit probability density functions, Lemma 2 simply means that h is the Bayes optimal classifier for the distribution $(\nu, \phi_1\#\mu_1, \phi_{-1}\#\mu_{-1})$[1].

3.2.2 Unbridled Attacker : trivial Nash equilibria can exist

In this section, we will consider the case where the Attacker only follows an imperceptibility constraint for some $\epsilon > 0$, i.e. with a transport cost $c = c_\epsilon$. Recall that :

$$c_\epsilon(x, z) = \begin{cases} 0 & \text{if } \|x - z\| \leq \epsilon \\ +\infty & \text{otherwise} \end{cases}$$

We will show that in that setting, trivial deterministic Nash equilibria may exist, i.e. equilibria where the Defender plays the same strategy with or without attack. For this, we will use an example from [Pydi and Jog, 2020a].

Let us take the example of two symmetrical normal distributions, with the same variance. Let $m, \sigma > 0$. We define $\mu_{-1} = \mathcal{N}(-m, \sigma^2)$ and $\mu_1 = \mathcal{N}(m, \sigma^2)$ (see Figure 3.7, with $m = 3$ and $\sigma = 3$). Let $\epsilon < m$.

The Bayes optimal classifier corresponding to these two distributions is simply $\mathbb{1}\{x \geq 0\}$. Furthermore, as demonstrated by [Pydi and Jog, 2020a] in Theorem 5, the optimal adversarial risk is :

[1] We prove this result in the supplementary material.

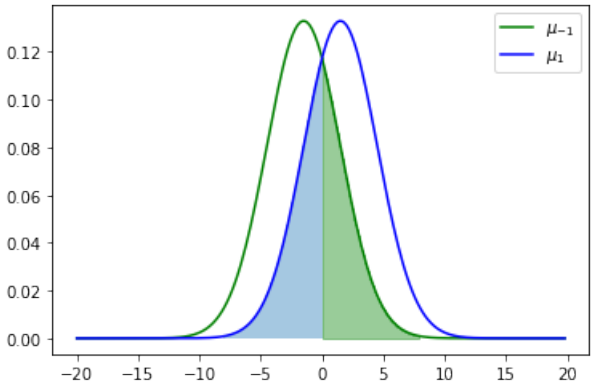


Figure 3.3: Distributions after attack of size $\epsilon = 1.5$

$$\overline{v} = \inf_{h \in \mathcal{H}} \sup_{\phi \in (\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_{\Omega}(h, \phi) = \Phi\left(\frac{m - \epsilon}{\sigma}\right)$$

Where Φ is the cumulative distribution function of the normal distribution $\mathcal{N}(0, 1)$.

We know from Proposition 2 that for any distribution, $\underline{v} \leq \overline{v}$. We will now show that for this particular distribution, $\underline{v} \geq \overline{v}$, thus prooving that the two are equal.

Let $\psi_1 : x \mapsto x - \epsilon$, and $\psi_{-1} : x \mapsto x + \epsilon$. This represents the most "basic" attack, i.e. pushing the whole conditional distributions toward the decision boundaries. This transforms μ_1 and μ_{-1} into $\psi_{-1}\#\mu_{-1} = \mathcal{N}(-m + \epsilon, \sigma^2)$ and $\psi_1\#\mu_1 = \mathcal{N}(m - \epsilon, \sigma^2)$ (see Figure 3.3)

A nice property of these distributions is that $\mu_1(x) > \mu_{-1}(x) \iff x > 0$. Hence, it is immediate from Lemma 2 that the optimal classifier, with or without attack, is $h^* : x \mapsto \mathbb{1}_{x>0}$. We can thus compute the adversarial risk easily :

$$\mathcal{R}(h^*, \psi) = \mathbb{P}\left[\mathcal{N}(m - \epsilon, \sigma^2) < 0\right] = \Phi\left(\frac{m - \epsilon}{\sigma}\right)$$

We thus have :

$$\underline{v} = \sup_{\phi} \inf_{h \in \mathcal{H}} \mathcal{R}(h, \phi) \geq \inf_{h \in \mathcal{H}} \mathcal{R}(h, \psi) = \Phi\left(\frac{m - \epsilon}{\sigma}\right) = \overline{v}$$

Hence the result. It follows that (ψ, h^*) is a pure Nash equilibrium in our game. We call this kind of equilibria *trivial*, because the Defender plays the same strategy with or without attack.

For convex surrogates, pure nash equilibria are also possible, as a special case of Appendix A when the cost allows the Attacker's optimal transport problem to have a Monge solution.

We will now study the existence of Nash equilibria for general, positive transport-based, imperceptibility-enforcing constraints. For that, we will study both player's best response for a given strategy of the opponent. In this section, we will consider the most fundamental loss function, namely the $0/1$-loss.

3.2.3 Attacker's best response

We will now study the Attacker's best response against a given classifier h. For this, we will need a few notations that will allow us a better grasp of the situation.

Definition 33. *A pure Nash equilibrium (h, ϕ) of the game is said to be trivial if $\phi = (\mathrm{Id}_{\mathcal{X}}, \mathrm{Id}_{\mathcal{X}})$*

Recall the definition of the classfication zones :

Definition 34 (Classification zones)**.** *The classification zones $C_y(h)$ are the partitions of $\mathcal{X}$ defined by h. More precisely :*

$$\forall y \in \{\pm 1\}, C_y(h) = \{x \in \mathcal{X} | h(x) = y\} \tag{3.9}$$

We will now analyze further these zones, by separating the points where an attack may succeed from the others.
Even when a point can be attack, the penalty incurred by moving the point is not always worth the increase in the loss function. We will now define the vulnerable zone, as the region where there exists an attack that brings a positive overall in score.

Definition 35 (Vulnerable zone)**.** *Let c be some transport cost. The vulnerable zones are the portion of each classification zone where the benefit of attacking strictly outweights the cost. It is the zone where a rational Attacker should always choose to attack.*

$$\forall y \in \{\pm 1\}, V_y(h, c) = \{x \in C_y(h) | \exists z \in C_{-y}(h), c(x, z) < 1\} \tag{3.10}$$

Finally, we define the indifference zone:

Definition 36 (Indifference zone)**.** *Let $\epsilon > 0$, c be some transport cost. The indifference zone is the portion of each classification zone where the benefit of attacking exactly matches the cost, which means that the Attacker is indifferent between attacking and not moving the point.*

$$\forall y \in \{\pm 1\}, I_y(h, c) = \{x \in C_y(h) | \exists z \in C_{-y}(h), c(x, z) = 1\} \tag{3.11}$$

We will now formalize these intuitions into the following lemma :

Lemma 3. *Let c be a positive transport cost. We consider a game with cost-based constraint of cost c. Let h be some classifier, and $\boldsymbol{\phi} \in \mathcal{BR}(h)$.. Then :*

- *For $x \in \mathcal{X} \setminus (V_y(h, c) \cup I_y(h, c))$, $\phi_y(x) = x$ almost surely;*
- *For $x \in V_y(h, c)$, $\phi_y(x) \in C_{-y}(h)$ almost surely;*
- *For $x \in I_y(h, c)$, $\phi_y(x) \in \{x\} \cup C_{-y}(h)$ almost surely.*

Proof. Let $\boldsymbol{\phi} \in \mathcal{BR}(h)$.
First step : not moving is better outside of the attackable zone.
Let us reason ad absurdum and assume that there is a zone $A \subset \mathcal{X} \setminus (V_y(h, c) \cup I_y(h, c))$ of nonzero measure such that $\forall x \in A, \phi_1(x) \neq x$. The proof works exactly the same when replacing 1 with -1. Let $\boldsymbol{\psi}$ be such that $\psi_{-1} = \phi_{-1}$ and:

$$\psi_1(x) = \begin{cases} \phi_1(x) & \text{for } x \in \mathcal{X} \setminus A \\ x & \text{for } x \in A \end{cases}$$

Let $\Delta R = \mathcal{R}_\Omega(h, \boldsymbol{\phi}) - \mathcal{R}_\Omega(h, \boldsymbol{\psi})$ Then :

$$\begin{aligned} \Delta R &= \sum_{i=\pm 1} q_i \int_{\mathbb{R}^d} [\mathbb{1}\{h(\phi_i(x)) \neq i\} - c(x, \phi_i(x))] - [\mathbb{1}\{h(\psi_i(x)) \neq i\} - c(x, \psi_i(x))] d\mu_i(x) \\ &= \int_A [q_1(\mathbb{1}\{h(\phi_1(x)) \neq 1\} - c(x, \phi_1(x))) - q_{-1}(\mathbb{1}\{h(\psi_1(x)) \neq 1\} - c(x, \psi_1(x)))] d\mu_1(x) \end{aligned}$$

Since $\psi_{-1} = \phi_{-1}$ and $\psi_1(x) = \phi_1(x)$ outside of A. Furthermore, $\psi_1(x) = x$ and $\phi_1(x) \neq x$ on A. $A \cap (V_1(h, \epsilon) \cup I_y(h, \epsilon, c)) = \emptyset$ so we know that for all $x \in A, z \in C_{-1}(h)$, $c(x, z) > 1$. It follows that :

$$\begin{aligned}
\Delta R &= \int_A q_1[\mathbb{1}\{h(\phi_1(x)) \neq 1\} - c(x, \phi_1(x))] - q_{-1}[\mathbb{1}\{h(x) \neq 1\} - c(x, \psi_1(x))] d\mu_1(x) \\
&\leq \int_A q_1[1 - c(x, \phi_1(x))] - q_{-1}[0 - k(x, x)] d\mu_1(x) \\
&\leq \int_A q_1(\underbrace{1 - c(x, \phi_1(x))}_{<0}) d\mu_1(x) \\
&< 0
\end{aligned}$$

Since A is of nonzero measure. We have constructed $\boldsymbol{\psi}$ that gives a strictly higher score than $\boldsymbol{\phi}$, contradiction.

Second step : When the cost is lower than the gain, then attacking is always the strictly best solution.

We once again reason ad absurdum and assume that there exists a zone $A \subset C_{-1}(h) \cup V_1(h, \epsilon, c)$ of nonzero measure such that $\forall x \in A, \phi_1(x) \notin C_{-1}(h)$. Let $\boldsymbol{\psi}$ be such that $\psi_{-1} = \phi_{-1}$ and :

$$\psi_1(x) = \left\{ \begin{array}{l} \phi_1(x) \\ z \in N_h \text{ such that } c(x, z) < 1 \end{array} \middle| \begin{array}{l} \text{for } x \in \mathcal{X} \setminus A \\ \text{for } x \in A \end{array} \right.$$

We know that such a z exists for $x \in V_1(h, \epsilon, c)$ by definition, and for $x \in C_{-1}(h)$, $z = x$ works. Let $\Delta R = \mathcal{R}_\Omega(h, \boldsymbol{\phi}) - \mathcal{R}_\Omega(h, \boldsymbol{\psi})$ Then :

$$\begin{aligned}
\Delta R &= \sum_{i=\pm 1} q_i \int_{\mathbb{R}^d} [\mathbb{1}\{h(\phi_i(x)) \neq i\} - c(x, \phi_i(x))] - [\mathbb{1}\{h(\psi_i(x)) \neq i\} - c(x, \psi_i(x))] d\mu_i(x) \\
&= \int_A q_1[\mathbb{1}\{h(\phi_1(x)) \neq 1\} - c(x, \phi_1(x))] - q_{-1}[\mathbb{1}\{h(\psi_1(x)) \neq 1\} - c(x, \psi_1(x))] d\mu_1(x)
\end{aligned}$$

Since $\psi_{-1} = \phi_{-1}$ and $\psi_1(x) = \phi_1(x)$ outside of A.
By hypothesis, $h(\phi_1(x)) = 1$ on A, and by construction $h(\psi_1(x)) = -1$ on A. Hence :

$$\begin{aligned}
\Delta R &= \int_A q_1[0 - c(x, \phi_1(x))] - q_{-1} \underbrace{[1 - c(x, \psi_1(x))]}_{>0 \text{ by construction}} d\mu_1(x) \\
&< q_1 \int_A [-c(x, \phi_1(x))] < 0
\end{aligned}$$

Hence a contradiction.

Third step : when the cost is equal to the gain, the Attacker is indifferent between not moving and doing an attack that passes the decision boundary.
Ad absurdum, let us assume that there is a zone $A \subset I_1(h, \epsilon, c)$ of nonzero measure, such that $\forall x \in A, \phi_1(x) \neq x$ and $\phi_1(x) \notin C_{-1}(h)$. Let $\boldsymbol{\psi}$ be such that $\psi_{-1} = \phi_{-1}$ and:

$$\psi_1(x) = \left\{ \begin{array}{l|l} \phi_1(x) & \text{for } x \in \mathcal{X} \setminus A \\ x & \text{for } x \in A \end{array} \right.$$

Let $\Delta R = \mathcal{R}_\Omega(h, \boldsymbol{\phi}) - \mathcal{R}_\Omega(h, \boldsymbol{\psi})$ Then :

$$\begin{aligned} \Delta R &= \sum_{i=\pm 1} q_i \int_{\mathbb{R}^d} [\mathbb{1}\{h(\phi_i(x)) \neq i\} - c(x, \phi_i(x))] - [\mathbb{1}\{h(\psi_i(x)) \neq i\} - c(x, \psi_i(x))] d\mu_i(x) \\ &= \int_A q_1[\mathbb{1}\{h(\phi_1(x)) \neq 1\} - c(x, \phi_1(x))] - q_{-1}[\mathbb{1}\{h(\psi_1(x)) \neq 1\} - c(x, \psi_1(x))] d\mu_1(x) \end{aligned}$$

Since $\psi_{-1} = \phi_{-1}$ and $\psi_1(x) = \phi_1(x)$ outside of A. Furthermore, $\psi_1(x) = x$ and $\phi_1(x) \neq x$ on A, and $h(\phi_1(x)) = 1$. Hence :

$$\begin{aligned} \Delta R &= \int_A q_1[\mathbb{1}\{h(\phi_1(x)) \neq 1\} - c(x, \phi_1(x))] - q_{-1}[\mathbb{1}\{h(x) \neq 1\} - c(x, \psi_1(x))] d\mu_1(x) \\ &= \int_A q_1[0 - c(x, \phi_1(x))] - q_{-1}[0 - c(x, x)] d\mu_1(x) \\ &= q_1 \int_A (-c(x, \phi_1(x))) d\mu_1(x) \\ &< 0 \end{aligned}$$

Since A is of nonzero measure and for every $x \in A$, $\phi_y(x) \neq x$ so $k(x, \phi_y(x)) > 0$. We have constructed $\boldsymbol{\psi}$ that gives a strictly higher score than $\boldsymbol{\phi}$, contradiction.

□

3.2.4 Non-existence of pure Nash equilibria in this setting

Theorem 7. *In the zero-sum game using $L_{0/1}$, with any positive cost-based constraint, there can be no pure Nash equilibrium.*
Note that this is true in particular for positive imperceptibility-preserving constraints.

Proof. Let us assume, ad absurdum, that there is a pure Nash Equilibrium $(h, \boldsymbol{\phi})$. A major consequence from Lemma 3 is that for all y, $\phi_y \# \mu_y(V_y(h, c)) = 0$. This is easy to see, since :

- for $x \in C_{-y}(h) \cup V_y(h, c)$, $\phi_y(x) \in C_{-1}(h)$ so $\phi_y(x) \notin V_y(h, c)$;
- for $x \in C_y(h) \setminus (V_y(h, c) \cup I_y(h, c))$, $\phi_y(x) = x \notin V_y(h, c)$;
- for $x \in I_y(h, c)$, either $\phi_y(x) = x$ or $\phi_y(x) \in C_{-y}(x)$.
  In both cases, $\phi_y(x) \notin V_y(h, \epsilon, k)$.

Another direct consequence of the lemma is that

$$\phi_{-y} \# \mu_{-y}(V_y(h, c)) = \mu_{-y}(V_y(h, c) > 0$$

since $\phi_{-y} = Id$ on $C_y(h)$. It follows that :

$$\nu_{-y}\phi_{-y} \# \mu_{-y}(V_y(h, c)) > 0 = \nu_y \phi_y \# \mu_y(V_y(h, c)$$

and by Lemma 2 $h \notin \mathcal{BR}(\boldsymbol{\phi})$. Contradiction. □

3.2.5 Consequences of this non-existence result

We will now analyze the importance of the previously obtained result. There are two major consequences to the non-existence of Nash equilibria in the deterministic setting:

Consequence 1. *There is no free lunch for transferable attacks.*

To understand this statement, remark that, thanks to weak duality, the following inequality always holds:

$$\sup_{\boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2} \inf_{h \in \mathcal{H}} \mathcal{R}_\Omega(h, \boldsymbol{\phi}) \leq \inf_{h \in \mathcal{H}} \sup_{\boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_\Omega(h, \boldsymbol{\phi}).$$

On the left side problem (sup-inf), the Adversary looks for the best attacking strategy $\boldsymbol{\phi}$ against any *unknown* classifier. This is tightly related to the notion of *transferable attacks* (*e.g.* [Tramèr et al., 2017]), which refers to attacks successful against a wide range of

classifiers. On the right side our problem (inf-sup), where the Defender tries to find the best classifier under any possible attack, whereas the Adversary plays in second and specifically attacks this classifier. As a consequence of Theorem 7, the inequality is always strict:

$$\sup_{\phi \in (\mathcal{F}_{\mathcal{X}})^2} \inf_{h \in \mathcal{H}} \mathcal{R}_\Omega(h, \phi) < \inf_{h \in \mathcal{H}} \sup_{\phi \in (\mathcal{F}_{\mathcal{X}})^2} \mathcal{R}_\Omega(h, \phi).$$

This means that both problems are not equivalent. In particular, an attack designed to succeed against *any* classifier (*i.e.* a transferable attack) will not be as good as an attack tailored for a given classifier.

Consequence 2. *No deterministic defense may be proof against every attack.*

Let us consider the state-of-the-art defense which is Adversarial Training. The idea is to compute an efficient attack ϕ, and train the classifier on created adversarial examples, in order to move the decision boundary and make the classifier more robust to new perturbations by ϕ.
To be fully efficient, this method requires that ϕ remains an optimal attack on h even after training. Our theorem shows that it is never the case: after training our classifier h to become (h') robust against ϕ, there will always be a different optimal attack ϕ' that is efficient against h'. Hence Adversarial Training will never achieve a perfect defense.

3.3 Randomized Defender : outperforming deterministic defenses

In this section, we will study the behavior of randomized defenses, and show that under some specific cost functions, they can always outperform deterministic algorithms under attack.

We keep using the $0/1$ loss, as well as the notations from Section 3.2. We will focus here on the ℓ_2 norm (but any hilbert norm would work as well), and two kinds of imperceptibility-preserving costs : the Carlini-Wagner cost and the Mass cost, as defined in Section 3.1.3. Recall that both costs are parametrized by a strength factor $\lambda > 0$, and are defined by :

$$\begin{aligned}\Omega_{\text{norm}}(\phi) := \lambda \mathop{\mathbb{E}}_{Y \sim \nu} \Big[\mathop{\mathbb{E}}_{X \sim \mu_Y} \big[\|X - \phi_Y(X)\|_2 \\ + \infty \mathbb{1}\{\|X - \phi_Y(X)\|_2 > \epsilon_2\} \big] \Big],\end{aligned} \tag{3.12}$$

$$\Omega_{\text{mass}}(\boldsymbol{\phi}) := \lambda \mathop{\mathbb{E}}_{Y\sim\nu}\Big[\mathop{\mathbb{E}}_{X\sim\mu_Y}\big[\mathbb{1}\{X \neq \phi_Y(X)\} + \infty\mathbb{1}\{\|X-\phi_Y(X)\|_\infty > \epsilon_\infty\}\big]\Big]. \tag{3.13}$$

On more general costs functions We believe that the results from this section could be easily obtained for any positive cost function, as long as we can characterize the associated optimal attack against any classifier. This is the reason why we selected examples of costs instead of providing a general result. The method, however, is easily transposable to other costs.

We call $\mathcal{BR}_{\text{norm}}$ and $\mathcal{BR}_{\text{mass}}$ the corresponding best response sets for the Attacker.

3.3.1 Best response analysis for both cost functions

Before we can introduce randomization, we must first study the behavior of the optimal attack under both costs functions. Let us introduce a more specific version of the attackable zone :

Definition 37. $P_h(\epsilon) = \{x \in C_1(h) | \exists z \in C_{-1}(h), \|z-x\|_2 \leq \epsilon\}$

Lemma 2. *Let $h \in \mathcal{H}$ and $\boldsymbol{\phi} \in \mathcal{BR}_{norm}(h)$. Then the following assertion holds:*

$$\phi_1(x) = \begin{cases} \pi(x) & \text{if } x \in P_h(\epsilon_2) \\ x & \text{otherwise.} \end{cases}$$

Where π is the orthogonal projection on $(P_h)^{\complement}$. ϕ_{-1} is characterized symmetrically.

Proof. We will make this proof a little more general by only assuming that $\mathcal{X}$ is an Hilbert space with dot product $<|>$ and associated norm $||.|| = \sqrt{<.\,|\,.>}$. Let us first simplify the worst case adversarial risk for h. Recall that $h = \text{sign}(g)$ with g continuous. From the definition of adversarial risk we have:

$$\sup_{\boldsymbol{\phi}\in\left(\mathcal{F}_{\mathcal{X}|\epsilon_2}\right)^2} \mathcal{R}_{\Omega_{norm}}(h, \boldsymbol{\phi}) \tag{3.14}$$

$$= \sup_{\boldsymbol{\phi}\in(\mathcal{F}_{\mathcal{X}})^2} \sum_{y=\pm1} \nu_y \mathop{\mathbb{E}}_{X\sim\mu_y}\big[\mathbb{1}\{h(\phi_y(X)) \neq y\} - \lambda\|X-\phi_y(X)\| - \infty\mathbb{1}\{\|X-\phi_y(X)\| > \epsilon_2\}\big] \tag{3.15}$$

$$= \sup_{\boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2} \sum_{y=\pm 1} \nu_y \mathop{\mathbb{E}}_{X \sim \mu_y} \left[\mathbb{1}\{g(\phi_y(X))y \leq 0\} - \lambda \|X - \phi_y(X)\| - \infty \mathbb{1}\{\|X - \phi_y(X)\| > \epsilon_2\} \right] \tag{3.16}$$

$$= \sum_{y=\pm 1} \nu_y \sup_{\phi_y \in \mathcal{F}_{\mathcal{X}}} \mathop{\mathbb{E}}_{X \sim \mu_y} \left[\mathbb{1}\{g(\phi_y(X))y \leq 0\} - \lambda \|X - \phi_y(X)\| - \infty \mathbb{1}\{\|X - \phi_y(X)\| > \epsilon_2\} \right] \tag{3.17}$$

Finding ϕ_1 and ϕ_1 are two independent optimization problems, hence, we focus on characterizing ϕ_1 (*i.e.* $y = 1$).

$$\sup_{\phi_1 \in \mathcal{F}_{\mathcal{X}}} \mathop{\mathbb{E}}_{X \sim \mu_1} \left[\mathbb{1}\{g(\phi_1(X)) \leq 0\} - \lambda \|X - \phi_1(X)\| - \infty \mathbb{1}\{\|X - \phi_1(X)\| > \epsilon_2\} \right] \tag{3.18}$$

$$= \mathop{\mathbb{E}}_{X \sim \mu_1} \left[\operatorname*{essup}_{z \in B_{\|.\|}(X, \epsilon_2)} \mathbb{1}(g(z) \leq 0) - \lambda \|X - z\| \right] \tag{3.19}$$

$$= \int_{\mathcal{X}} \operatorname*{essup}_{z \in B_{\|.\|}(x, \epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda \|x - z\| \; d\mu_1(x). \tag{3.20}$$

Let us now consider $(H_j)_{j \in J}$ a partition of $\mathcal{X}$, we can write.

$$\sup_{\phi_1 \in \mathcal{F}_{\mathcal{X}}} \mathop{\mathbb{E}}_{X \sim \mu_1} \left[\mathbb{1}\{g(\phi_1(X)) \leq 0\} - \lambda \|X - \phi_1(X)\| - \infty \mathbb{1}\{\|X - \phi_1(X)\| > \epsilon_2\} \right] \tag{3.21}$$

$$= \sum_{j \in J} \int_{H_j} \operatorname*{essup}_{z \in B_{\|.\|}(x, \epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda \|x - z\| \; d\mu_1(x) \tag{3.22}$$

In particular, we consider here $H_0 = P_h^{\complement}$, $H_1 = P_h \setminus P_h(\epsilon_2)$, and $H_2 = P_h(\epsilon_2)$.

For $x \in H_0 = P_h^{\complement}$. Taking $z = x$ we get $\mathbb{1}\{g(z) \leq 0\} - \lambda \|x - z\| = 1$. Since for any $z \in \mathcal{X}$ we have $\mathbb{1}\{g(z) \leq 0\} - \lambda \|x - z\| \leq 1$, this strategy is optimal. Furthermore, for any other optimal strategy z', we would have $\|x - z'\| = 0$, hence $z' = x$, and an optimal attack will never move the points of $H_0 = P_h^{\complement}$.

For $x \in H_1 = P_h \setminus P_h(\epsilon_2)$. We have $B_{\|.\|}(x, \epsilon_2) \subset P_h$ by definition of $P_h(\epsilon_2)$. Hence, for any $z \in B_{\|.\|}(x, \epsilon_2)$, one gets $g(z) > 0$. Then $\mathbb{1}\{g(z) \leq 0\} - \lambda \|x - z\| \leq 0$. The only optimal z will thus be $z = x$, giving value 0.

Let us now consider $x \in H_2 = P_h(\epsilon_2)$ which is the interesting case where an attack is possible. We know that $B_{\|.\|}(x, \epsilon_2) \cap P_h^{\complement} \neq \emptyset$, and for any z in this intersection, $\mathbb{1}(g(z) \leq 0) = 1$. Hence :

$$\operatorname*{essup}_{z \in B_{\|.\|}(x,\epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda\|x - z\| = \max(1 - \lambda \operatorname*{ess\,inf}_{z \in B_{\|.\|}(x,\epsilon_2) \cap P_h^{\complement}} \|x - z\|, 0) \tag{3.23}$$

$$= \max(1 - \lambda \pi_{B_{\|.\|}(x,\epsilon_2) \cap P_h^{\complement}}(x), 0) \tag{3.24}$$

Where $\pi_{B_{\|.\|}(x,\epsilon_2) \cap P_h^{\complement}}$ is the projection on the closure of $B_{\|.\|}(x, \epsilon_2) \cap P_h^{\complement}$. Note that $\pi_{B_{\|.\|}(x,\epsilon_2) \cap P_h^{\complement}}$ exists: g is continuous, so $B_{\|.\|}(x, \epsilon_2) \cap P_h^{\complement}$ is a closed set, bounded, and thus compact, since we are in finite dimension. The projection is however not guaranteed to be unique since we have no evidence on the convexity of the set. Finally, let us remark that, since $\lambda \in (0, 1)$, and $\epsilon_2 \leq 1$, one has $1 - \lambda \pi_{B_{\|.\|}(x,\epsilon_2) \cap P_h^{\complement}}(x) \geq 0$ for any $x \in H_2$. Hence, on $P_h(\epsilon_2)$, the optimal attack projects all the points on the decision boundary. For simplicity, and since there is no ambiguity, we write the projection π.

Finally. Since $H_0 \cup H_1 \cup H_2 = \mathcal{X}$, Lemma 2 holds. Furthermore, the score for this optimal attack is:

$$\sup_{\phi \in \left(\mathcal{F}_{\mathcal{X}|\epsilon_2}\right)^2} \mathcal{R}_{\text{adv}}^{\Omega_{\text{norm}}}(h, \phi) \tag{3.25}$$

$$= \sum_{y=\pm 1} \nu_y \sum_{j \in J} \int_{H_j} \operatorname*{essup}_{z \in B_{\|.\|}(x,\epsilon_2)} \mathbb{1}\{g(z)y \leq 0\} - \lambda\|x - z\| \; d\mu_y(x) \tag{3.26}$$

Since the value is 0 on $P_h \setminus P_h(\epsilon_2)$ (resp. on $N_h \setminus N_h(\epsilon_2)$) for ϕ_1 (resp. $\phi_{\text{-}1}$), one gets:

$$= q_1 \left[\int_{P_h(\epsilon_2)} \big(1 - \lambda\|x - \pi(x)\|\big) d\mu_1(x) + \int_{P_h^{\complement}} 1 d\mu_1(x) \right] + \nu_{\text{-}1} \left[\int_{N_h(\epsilon_2)} \big(1 - \lambda\|x - \pi(x)\|\big) d\mu_{\text{-}1}(x) + \int_{N_h^{\complement}} 1 d\mu_{\text{-}1}(x \right. \tag{3.27}$$

$$= q_1 \left[\int_{P_h(\epsilon_2)} \big(1 - \lambda\|x - \pi(x)\|\big) d\mu_1(x) + \mu_1(P_h^{\complement}) \right] + \nu_{\text{-}1} \left[\int_{N_h(\epsilon_2)} \big(1 - \lambda\|x - \pi(x)\|\big) d\mu_{\text{-}1}(x) + \mu_{\text{-}1}(N_h^{\complement}) \right] \tag{3.28}$$

$$= \mathcal{R}(h) + q_1 \int_{P_h(\epsilon_2)} \big(1 - \lambda \|x - \pi(x)\|\big) d\mu_1(x) + \nu_{\text{-}1} \int_{N_h(\epsilon_2)} \big(1 - \lambda \|x - \pi(x)\|\big) d\mu_{\text{-}1}(x) \tag{3.29}$$

The last equation holds since $\mathcal{R}(h) = \mathbb{P}(h(X) \neq Y)\mathbb{P}(g(X)Y \leq 0) = q_1\mu_1(P_h^{\complement}) + \nu_{\text{-}1}\mu_{\text{-}1}(N_h^{\complement})$. This provides an interesting decomposition of the adversarial risk into the risk without attack and the loss on the attack zone. □

Lemma 3. *Let $h \in \mathcal{H}$ and $\boldsymbol{\phi} \in \mathcal{BR}_{mass}(h)$. Then the following assertion holds:*

$$\begin{cases} \phi_1(x) \in (P_h)^{\complement} & \text{if } x \in P_h(\epsilon_2) \\ \phi_1(x) = x & \text{otherwise.} \end{cases}$$

Where $(P_h)^{\complement}$, the complement of P_h in $\mathcal{X}$. $\phi_{\text{-}1}$ is characterized symmetrically.

Proof. Following the same proof schema as before the adversarial risk writes as follows:

$$\sup_{\boldsymbol{\phi} \in \left(\mathcal{F}_{\mathcal{X}|\epsilon_2}\right)^2} \mathcal{R}_{\Omega_{\text{mass}}}(h, \phi) \tag{3.30}$$

$$= \sup_{\boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2} \sum_{y=\pm 1} \nu_y \mathop{\mathbb{E}}_{X \sim \mu_y} [\mathbb{1}\{h(\phi_y(X)) \neq y\} - \lambda \mathbb{1}\{X \neq \phi_y(X)\} - \infty \mathbb{1}\{\|X - \phi_y(X)\| > \epsilon_2\}] \tag{3.31}$$

$$= \sup_{\boldsymbol{\phi} \in (\mathcal{F}_{\mathcal{X}})^2} \sum_{y=\pm 1} \nu_y \mathop{\mathbb{E}}_{X \sim \mu_y} [\mathbb{1}\{g(\phi_y(X))y \leq 0\} - \lambda \mathbb{1}\{X \neq \phi_y(X)\} - \infty \mathbb{1}\{\|X - \phi_y(X)\| > \epsilon_2\}] \tag{3.32}$$

$$= \sum_{y=\pm 1} \nu_y \sup_{\phi_y \in \mathcal{F}_{\mathcal{X}}} \mathop{\mathbb{E}}_{X \sim \mu_y} [\mathbb{1}\{g(\phi_y(X))y \leq 0\} - \lambda \mathbb{1}\{X \neq \phi_y(X)\} - \infty \mathbb{1}\{\|X - \phi_y(X)\| > \epsilon_2\}] \tag{3.33}$$

Finding ϕ_1 and ϕ_1 are two independent optimization problem, hence we focus on characterizing ϕ_1 (*i.e.* $y = 1$).

$$\sup_{\phi_1 \in \mathcal{F}_{\mathcal{X}}} \mathop{\mathbb{E}}_{X \sim \mu_1} \big[\mathbb{1}\{g(\phi_1(X)) \leq 0\} - \lambda \mathbb{1}\{X \neq \phi_1(X)\} - \infty \mathbb{1}\{\|X - \phi_1(X)\| > \epsilon_2\}\big] \tag{3.34}$$

$$= \mathop{\mathbb{E}}_{X \sim \mu_1} \left[\operatorname*{essup}_{z \in B_{\|.\|}(X, \epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda \mathbb{1}\{X \neq z\} \right] \tag{3.35}$$

$$= \int_{\mathcal{X}} \operatorname*{essup}_{z \in B_{\|.\|}(x,\epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda \mathbb{1}\{x \neq z\} \; d\mu_1(x). \tag{3.36}$$

Let us now consider $(H_j)_{j \in J}$ a partition of $\mathcal{X}$, we can write.

$$\sup_{\phi_1 \in \mathcal{F}_{\mathcal{X}}} \mathop{\mathbb{E}}_{X \sim \mu_1} \left[\mathbb{1}\{g(\phi_1(X)) \leq 0\} - \lambda \mathbb{1}\{X \neq \phi_1(X)\} - \infty \mathbb{1}\{\|X - \phi_1(X)\| > \epsilon_2\} \right] \tag{3.37}$$

$$= \sum_{j \in J} \int_{H_j} \operatorname*{essup}_{z \in B_{\|.\|}(x,\epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda \mathbb{1}\{x \neq z\} \; d\mu_1(x) \tag{3.38}$$

In particular, we can take $H_0 = P_h^{\complement}$, $H_1 = P_h \setminus P_h(\epsilon_2)$, and $H_2 = P_h(\epsilon_2)$.

For $x \in H_0 = P_h^{\complement}$ or $x \in H_1 = P_h \setminus P_h(\epsilon_2)$. With the same reasoning as before, any optimal attack will choose $\phi_1(x) = x$.

Let $x \in H_2 = P_h(\epsilon_2)$. We know that $B_{\|.\|}(x, \epsilon_2) \cap P_h^{\complement} \neq \emptyset$, and for any z in this intersection, one has $g(z) \leq 0$ and $z \neq x$. Hence $\operatorname*{essup}_{z \in B_{\|.\|}(x,\epsilon_2)} \mathbb{1}\{g(z) \leq 0\} - \lambda \mathbb{1}\{z \neq x\} = \max(1 - \lambda, 0)$. Since $\lambda \in (0, 1)$ one has $\mathbb{1}\{g(z) \leq 0\} - \lambda \mathbb{1}\{z \neq x\} = 1 - \lambda$ for any $z \in B_{\|.\|}(x, \epsilon_2) \cap P_h^{\complement}$. Then any function that given a $x \in \mathcal{X}$ outputs $\phi_1(x) \in B_{\|.\|}(x, \epsilon_2) \cap P_h^{\complement}$ is optimal on H_2.

Finally. Since $H_0 \cup H_1 \cup H_2 = \mathcal{X}$, Lemma 3 holds. □

Armed with these lemma, we can now consider mixtures of classifiers.

3.3.2 Modelling randomized defenses

We have shown, in Theorem 7 that under any realistic constraints on the Attacker, there can be no pure Nash equilibrium. This means that no deterministic classifier may be proof against every attack. We would therefore need to allow for a wider class of strategies. A natural extension of the game would thus be to allow randomization for both players, who would now choose a distribution over pure strategies, leading to this game:

$$\inf_{\eta \in \mathcal{P}(\mathcal{H})} \sup_{\varphi \in \mathcal{P}((\mathcal{F}_{\mathcal{X}})^2)} \mathop{\mathbb{E}}_{\substack{h \sim \eta \\ \boldsymbol{f} \sim \varphi}} [\mathcal{R}_{\Omega}(h, \boldsymbol{\phi})]. \tag{3.39}$$

Without making further assumptions on this game (e.g. compactness), we cannot apply known results from game theory (e.g. Sion theorem) to prove the existence of an equilib-

rium in this setting. These assumptions would however make the problem loose much generality, and does not hold here.

Randomization matters. Even without knowing if an equilibrium exists in the randomized setting, we can prove that *randomization matters*. More precisely we show that, under mild condition on the data distribution, any deterministic classifier can be outperformed by a randomized one in terms of the worst case adversarial risk. To do so we simplify Equation A.1 in two ways.

1. We do not consider the Adversary to be randomized, i.e we restrict the search space of the Adversary to $(\mathcal{F}_{\mathcal{X}})^2$ instead of $\mathcal{P}((\mathcal{F}_{\mathcal{X}})^2)$. This condition corresponds to the current state-of-the-art in the domain: to the best of our knowledge, no efficient randomized adversarial example attack has been designed (and so is used) yet. We will explore that problem in Appendix chapter A.

2. We only consider a subclass of randomized classifiers, called mixtures, which are discrete probability measures on a finite set of classifier. We show that this kind of randomization is enough to strictly outperform any deterministic classifier. We will discuss later the use of more general randomization (such as noise injection) for the Defender. Let us now define a mixture of classifiers:

Definition 38 (Mixture of classifier)**.** *Let $n \in \mathbb{N}$, $\mathbf{h} = (h_1, ..., h_n) \in \mathcal{H}^n$, and $\mathbf{q} \in \mathcal{P}([n])$. A* mixed classifier of $\mathbf{h}$ by $\mathbf{q}$ *is a mapping $m_{\mathbf{h}}^{\mathbf{q}}$ from $\mathcal{X}$ to $\mathcal{P}(\mathcal{Y})$ such that for all $x \in \mathcal{X}$, $m_{\mathbf{h}}^{\mathbf{q}}(x)$ is the discrete probability distribution that is defined for all $y \in \mathcal{Y}$ by:*

$$m_{\mathbf{h}}^{\mathbf{q}}(x)(y) := \mathop{\mathbb{E}}_{i\sim\mathbf{q}}[\mathbb{1}\{h_i(x) = y\}].$$

We call such a mixture a *mixed strategy* of the Defender. Given some $x \in \mathcal{X}$, this amounts to picking a classifier h_i from $\mathbf{h}$ at random following the distribution $\mathbf{q}$, and use it to output the predicted class for x, i.e $h_i(x)$. Note that a mixed strategy for the Defender is a non deterministic algorithm, since it depends on the sampling one makes on $\mathbf{q}$. Hence, even if the attacks are defined in the same way as before, the Adversary now needs to maximize a new objective function which is the expectation of the adversarial risk under the distribution $m_{\mathbf{h}}^{\mathbf{q}}$. It writes as follows:

$$\mathcal{R}_{\Omega}(m_{\mathbf{h}}^{\mathbf{q}}) = \mathop{\mathbb{E}}_{Y\sim\nu}\left[\mathop{\mathbb{E}}_{X\sim\phi_Y\#\mu_Y}\left[\mathop{\mathbb{E}}_{\hat{Y}\sim m_{\mathbf{h}}^{\mathbf{q}}(X)}\left[\mathbb{1}\left\{\hat{Y} \neq Y\right\}\right]\right]\right] - \Omega(\phi). \tag{3.40}$$

We also write $\mathcal{R}_{\Omega}$ to mean Equation (3.40), when it is clear from context that the Defender uses a mixed classifier. Using this new set of strategies for the Defender, we can study

whether mixed classifiers outperform deterministic ones, and how to efficiently design them.

3.3.3 Mixtures can always outperform deterministic defenses

In this section, we will demonstrate that the efficiency of any deterministic defense can be improved using a simple mixture algorithm. This method presents similarities with the notions of fictitious play [Brown, 1951] in game theory, and boosting in machine learning [Freund and Schapire, 1995]. Given a deterministic classifier h_1, we combine it (via randomization) with the best response h_2 to its optimal attack. The rational behind this idea is that, by construction, efficient attacks on one of these two classifiers will not work on the other. If we can then calibrate the weights so that attacks on important zones have a low probability of succeeding, then the average risk under attack on the mixture will be low. We will thus need the following condition on the data distribution :

Definition 39 ((ϵ, p)-dilation and vanishing measure)**.** *Let U be a subset of $\mathcal{X}$, ϵ a positive value, $p \in \{2, \infty\}$, and μ a probability measure.*

1. *The* (ϵ, p)-dilation of U *is defined as follows:*

$$U \overset{p}{\oplus} \epsilon := \Big\{ u + v \mid (u, v) \in U \times \mathcal{X} \text{ and } \|v\|_p \leq \epsilon \Big\}.$$

2. *We say that μ is (ϵ, p)-vanishing*[a] *on U if we have:*

$$\mu\Big(U \overset{p}{\oplus} \epsilon \backslash U\Big) \leq \mu(U).$$

[a] As for P_h^p we omit p when it is clear from the context.

This is because mixing h_1 with h_2 has two opposite consequences on the adversarial risk. On one hand, where we only had to defend against attack on h_1, we are now also vulnerable to attacks on h_2, so the total set of possible attacks is now bigger. On the other hand, each attack will only work part of the time, depending on the probability distribution $\mathbf{q}$. When Definition 39 applies on the attackable zones, it ensures that we gain more than we loose.

Definition 40 (Attackable zone). *Let $\epsilon > 0$ be the imperceptibility threshold. Then the attackable zone is defined by :*

$$A_y(h, \epsilon) = \{x \in \mathcal{C}_y(h), \exists z \in C_{-y}(h), \|z - x\|_2 \leq \epsilon\}$$

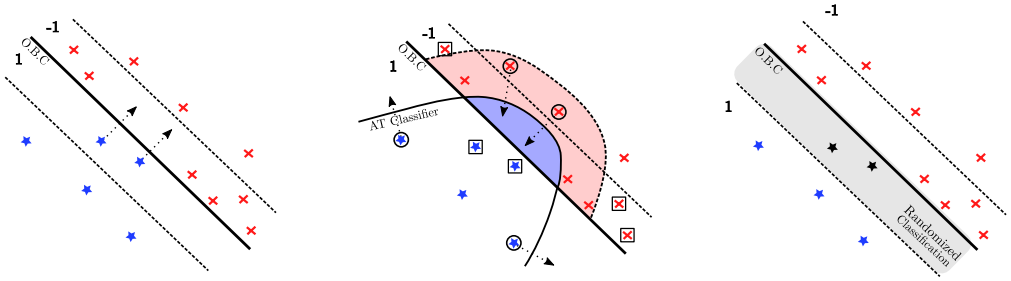


Figure 3.4: Illustration of adversarial examples (only on class 1 for more readability) crossing the decision boundary (left), adversarially trained classifier for the class 1 (middle), and a randomized classifier that defends class 1. Stars are natural examples for class 1, and crosses are natural examples for class -1. The straight line is the optimal Bayes classifier, and dashed lines delimit the points close enough to the boundary to be attacked resp. for class 1 and -1.

On the vanishing measure condition. Let us briefly explain this property. To defend against an attack, the general tactic is to change the classifier output, when points are close to the border (either all the time, as in Adversarial Training where we move the decision boundary to incorporate adversarial examples, or part of the time as in our randomized algorithm so that the attack only works with a given probability).

For example on figure 3.4, we mix the Bayes classifier (left) with its optimal attack that swaps the blue and red zone between the dotted line, on the gray area that is the former attack zone for the blue class. This gives the figure on the right. If the first classifier has a weight $\alpha = 0.5$, the 10 old attacks (points between the dotted lines) now succeed only with probability 0.5 (the new optimal attack for star points being to leave them in place), whereas 3 new attacks are created (blue points outside of the gray area) that succeed with probability 0.5, for a total attack score of 6.5, which is lower than the old attack score of 10.

When adversarially training a classifier (Figure 3.4, middle), we change its output on the blue zone, so that four of the star points cannot be successfully attacked anymore. But in exchange, the dilation of this zone (in red) can now be attacked. For Adversarial Training to work, we need the number of new potential attacks (*i.e.* the points that are circled, 2 red ones in the dilatation and 2 blue ones that are close to the new boundary) to

be smaller than the number of attacks we prevent (the points that are in a square, 4 blue ones that an attack would send in the blue zone, and 3 red points that are far from the new decision boundary).

This discussion shows that when no measure have any vanishing zone, Adversarial Training cannot bring any gain. By contraposition, whenever a deterministic classifier can be improved by Adversarial Training, it will also be outperformed under optimal attack by a randomized algorithm (see Theorem 8).

With these new definitions, we now can state our second main result: mixtures outperform deterministic classifiers.

Theorem 8. *(Randomization matters) Let $h_1 \in \mathcal{H}$, $\lambda \in (0,1)$, $\phi \in \mathcal{BR}(h_1)$, and $h_2 \in \mathcal{BR}(\phi)$. If for some $y \in \{\pm 1\}$, μ_y is ϵ_2-vanishing on $A_y(h, \epsilon_2)$, then for any $\alpha \in (\frac{1+\lambda\epsilon_2}{2}, 1)$ one has:*

$$\forall \phi' \in BR_{norm}(m_{\mathbf{h}}^{\mathbf{q}}), \mathcal{R}_{\Omega_{norm}}(m_{\mathbf{h}}^{\mathbf{q}}, \phi') < \mathcal{R}_{\Omega_{norm}}(h_1, \phi).$$

Where $\mathbf{h} = (h_1, h_2)$, $\mathbf{q} = (\alpha, 1-\alpha)$, and $m_{\mathbf{h}}^{\mathbf{q}}$ is the mixture of $\mathbf{h}$ by $\mathbf{q}$. A similar result holds for the mass penalty, with $\alpha \in (\frac{1+\lambda}{2}, 1)$.

Proof of Theorem 8. We first start by proving the result for the mass penalty.

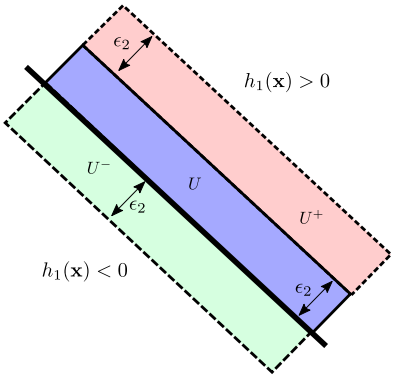


Figure 3.5: Illustration of the notations U, U^+, and U^- for proof of Theorem 8.

To demonstrate this theorem, let us assume for example $y = 1$ and denote $U = C_1(h, \epsilon_2)$. We can construct h_2 as follows

$$h_2(x) = \begin{cases} -h_1(x) & \text{if } x \in U \\ h_1(x) & \text{otherwise.} \end{cases}$$

This means that h_2 changes the class of all points in U, and do not change the rest, compared to h_1. Then taking $\alpha \in (0,1)$, we can define $m_{\boldsymbol{h}}^{\boldsymbol{q}}$, and $\boldsymbol{\phi}' \in \mathcal{BR}(m_{\boldsymbol{h}}^{\boldsymbol{q}})$. We aim to find a condition on α so that the score of $m_{\boldsymbol{h}}^{\boldsymbol{q}}$ is lower than the score of h_1. Finally, let us recall that

$$
\begin{aligned}
&\mathcal{R}_{\Omega_{\text{mass}}}(m_{\boldsymbol{h}}^{\boldsymbol{q}}, \boldsymbol{\phi}') \\
&= q_1 \int_{\mathcal{X}} \sup_{z \in B_{\|.\|}(x,\epsilon_2)} \alpha \mathbb{1}\{h_1(z) = \text{-}1\} + (1-\alpha)\mathbb{1}\{h_2(z) = \text{-}1\} - \lambda \mathbb{1}\{x \neq z\}\, d\mu_1(x) \\
&+ \nu_{\text{-}1} \int_{\mathcal{X}} \sup_{z \in B_{\|.\|}(x,\epsilon_2)} \alpha \mathbb{1}\{h_1(z) = 10\} + (1-\alpha)\mathbb{1}\{h_2(z) = 1\} - \lambda \mathbb{1}\{x \neq z\}\, d\mu_{\text{-}1}(x).
\end{aligned}
$$

The only terms that may vary between the score of h_1 and the score of $m_{\boldsymbol{h}}^{\boldsymbol{q}}$ are the integrals on U, $U \oplus \epsilon_2 \cap C_1(h_1)$ and $\phi_{\text{-}1}^{-1}(U)$ – inverse image of U by $\phi_{\text{-}1}$. These sets represent respectively the points we mix on, the points that may become attacked – when changing from h_1 to $m_{\boldsymbol{h}}^{\boldsymbol{q}}$ – by moving them on U, and the ones that were – for h_1 – attacked before by moving them on U. Hence, for simplicity, we only write those terms. Furthermore, we denote

$$
U^+ := U \oplus \epsilon_2 \cap C_1(h_1) \setminus U,\ U^- := \phi_{\text{-}1}^{-1}(U) \text{ and recall } U := A_1(h, \epsilon_2).
$$

One can refer to Figure 3.5 for visual interpretation of this sets. We can now evaluate the worst case adversarial score for h_1 restricted to the above sets. Thanks to Lemma 3 that characterizes $\boldsymbol{\phi}$, we can write

$$
\begin{aligned}
&\mathcal{R}_{\text{adv}}^{\Omega_{\text{mass}}}(h_1, \boldsymbol{\phi})_{|U,\, U^+,\, U^-} \\
&= (1-\lambda) \times q_1 \mu_1(U) + \nu_{\text{-}1}\mu_{\text{-}1}(U) \\
&+ 0 \times q_1 \mu_1\big(U^+\big) + \nu_{\text{-}1}\mu_{\text{-}1}\big(U^+\big) \\
&+ q_1 \mu_1\big(U^-\big) + (1-\lambda) \times \nu_{\text{-}1}\mu_{\text{-}1}\big(U^-\big).
\end{aligned}
$$

Similarly, we can write the worst case adversarial score of the mixture on the sets we consider. Note that the max operator comes from the fact that the adversary has to make a choice between attacking the zone or just take advantage of the error due to randomization.

$$
\begin{aligned}
&\mathcal{R}_{\text{adv}}^{\Omega_{\text{mass}}}(m_{\boldsymbol{h}}^{\boldsymbol{q}}, \boldsymbol{\phi}')_{|U,\, U^+,\, U^-} \\
&= \max(1-\alpha, 1-\lambda) \times q_1 \mu_1(U) + \max(\alpha, 1-\lambda) \times \nu_{\text{-}1}\mu_{\text{-}1}(U) \\
&+ \max(0, 1-\alpha-\lambda) \times q_1 \mu_1\big(U^+\big) + \nu_{\text{-}1}\mu_{\text{-}1}\big(U^+\big)
\end{aligned}
$$

$$+ q_1\mu_1\big(U^-\big) + \max(0, \alpha - \lambda) \times \nu_{\text{-}1}\mu_{\text{-}1}\big(U^-\big).$$

Computing the difference between these two terms, we get the following

$$\mathcal{R}_{\text{adv}}^{\Omega_{\text{mass}}}(h_1, \boldsymbol{\phi}) - \mathcal{R}_{\text{adv}}^{\Omega_{\text{mass}}}(m_{\boldsymbol{h}}^{\boldsymbol{q}}, \boldsymbol{\phi}') \tag{3.41}$$
$$= (1 - \lambda - \max(1 - \alpha, 1 - \lambda)) \times q_1\mu_1(U) \tag{3.42}$$
$$+ (1 - \max(\alpha, 1 - \lambda)) \times \nu_{\text{-}1}\mu_{\text{-}1}(U) \tag{3.43}$$
$$- \max(0, 1 - \alpha - \lambda) \times q_1\mu_1\big(U^+\big) \tag{3.44}$$
$$+(1 - \lambda - \max(0, \alpha - \lambda)) \times \nu_{\text{-}1}\mu_{\text{-}1}\big(U^-\big) \tag{3.45}$$

Let us now simplify Equation (3.41) using additional assumptions.

- First, we have that Equation (3.43) is equal to

$$\min(1 - \alpha, \lambda)\mu_{\text{-}1}(U)\nu_{\text{-}1} > 0.$$

  Thus, a sufficient condition for the difference between the adversarial scores to be positive is to have the other terms greater or equal to 0.

- To have Equation (3.42) ≥ 0 we can always set $\max(1 - \alpha, 1 - \lambda) = 1 - \lambda$. This gives us $\alpha \geq \lambda$.

- Also note that to get (3.44) ≥ 0, we can force $\max(1 - \alpha - \lambda, 0) = 0$. This gives us $\alpha \geq 1 - \lambda$.

- Finally, since $\alpha \geq \lambda$, we have that $1 - \lambda - \max(0, \alpha - \lambda) = 1 - \alpha$ thus Equations (3.45) > 0.

With the above simplifications, we have (3.41) > 0 for any $\alpha > \max(\lambda, 1 - \lambda)$ which concludes the proof. □

Let us now prove the version with the norm penalty :

Proof. Let us take $U \subset A_1(h, \epsilon_2)$ such that

$$\min_{x \in U} \|x - \pi_{P_h \setminus P_h(\epsilon_2)}(x)\| = \delta \in (0, \epsilon_2)$$

. We construct h_2 as follows.

$$h_2(x) = \begin{cases} -h_1(x) & \text{if } x \in U \\ h_1(x) & \text{otherwise.} \end{cases}$$

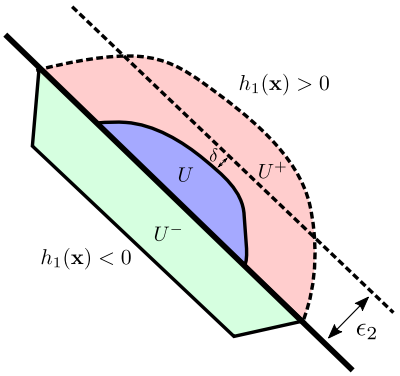


Figure 3.6: Illustration of the notations U, U^+, U^- and δ for proof of Theorem **??**.

This means that h_2 changes the class of all points in U, and do not change the rest. Let $\alpha \in (0,1)$, the corresponding mixture $m_{\boldsymbol{h}}^{\boldsymbol{q}}$, and $\boldsymbol{\phi}' \in \mathcal{BR}(m_{\boldsymbol{h}}^{\boldsymbol{q}})$. We will find a condition on α so that the score of $m_{\boldsymbol{h}}^{\boldsymbol{q}}$ is lower than the score of h_1. Recall that

$$\begin{aligned}
&\mathcal{R}_{\Omega_{\text{norm}}}(m_{\boldsymbol{h}}^{\boldsymbol{q}}, \boldsymbol{\phi}') \\
&= q_1 \int_{\mathcal{X}} \sup_{z \in B_{\|.\|}(x,\epsilon_2)} \alpha \mathbb{1}\{h_1(z) = \text{-}1\} + (1-\alpha)\mathbb{1}\{h_2(z) = \text{-}1\} - \lambda \|x - z\| \, d\mu_1(x) \\
&+ \nu_{\text{-}1} \int_{\mathcal{X}} \sup_{z \in B_{\|.\|}(x,\epsilon_2)} \alpha \mathbb{1}\{h_1(z) = 1\} + (1-\alpha)\mathbb{1}\{h_2(z) = 1\} - \lambda \|x - z\| \, d\mu_{\text{-}1}(x).
\end{aligned}$$

As we discussed in proof of Theorem 8, the only terms that may vary between the score of h_1 and the score of $m_{\boldsymbol{h}}^{\boldsymbol{q}}$ are the integrals on U, $U \oplus \epsilon_2 \cap C_1(h_1)$ and $\phi_{\text{-}1}^{-1}(U)$. Hence, for simplicity, we only write those terms. Furthermore, we denote

$$U^+ := U \oplus \epsilon_2 \cap C_1(h_1) \setminus U, \; U^- := \phi_{\text{-}1}^{-1}(U) \text{ and } P_{\epsilon_2} := A_1(h, \epsilon_2).$$

One can refer to Figure 3.6 for a visual interpretation of this ensembles. We can now evaluate the worst case adversarial score for h_1 restricted to the above sets. Thanks to Lemma 2 that characterizes $\boldsymbol{\phi}$, we can write

$$\begin{aligned}
&\mathcal{R}_{\text{adv}}^{\Omega_{\text{norm}}}(h_1, \boldsymbol{\phi}) \\
&= q_1 \int_{U} \Big(1 - \lambda \|x - \pi_{C_1(h_1)^{\complement}}(x)\|\Big) d\mu_1(x) + \nu_{\text{-}1}\mu_{\text{-}1}(U) \\
&+ q_1 \int_{U^+ \setminus P_{\epsilon_2}} 0 \, d\mu_1(x) + \nu_{\text{-}1}\mu_{\text{-}1}\big(U^+ \setminus P_{\epsilon_2}\big)
\end{aligned}$$

$$+ q_1 \int\limits_{U^+ \cap P_{\epsilon_2}} \Big(1 - \lambda \|x - \pi_{C_1(h_1)^{\complement}}(x)\|\Big) d\mu_1(x) + \nu_{\text{-}1}\mu_{\text{-}1}\big(U^+ \cap P_{\epsilon_2}\big)$$
$$+ q_1\mu_1\big(U^-\big) + \nu_{\text{-}1} \int\limits_{U^-} \Big(1 - \lambda \|x - \pi_U(x)\|\Big) d\mu_{\text{-}1}(x).$$

Similarly we can evaluate the worst case adversarial score for the mixture,

$$\begin{aligned}
&\mathcal{R}^{\Omega_{\text{norm}}}_{\text{adv}}(m^{\boldsymbol{q}}_{\boldsymbol{h}}, \boldsymbol{\phi}') \\
&= q_1 \int\limits_{U} \max\Big(1 - \alpha, 1 - \lambda \|x - \pi_{C_1(h_1)^{\complement}}(x)\|\Big)\ d\mu_1(x) \\
&+ \nu_{\text{-}1} \int\limits_{U} \max(\alpha, 1 - \lambda \|x - \pi_{U^+}(x)\|)\ d\mu_{\text{-}1}(x) \\
&+ q_1 \int\limits_{U^+ \setminus P_{\epsilon_2}} \max(0, 1 - \alpha - \lambda \|x - \pi_U(x)\|)\ d\mu_1(x) + \nu_{\text{-}1}\mu_{\text{-}1}\big(U^+ \setminus P_{\epsilon_2}\big) \\
&+ q_1 \int\limits_{U^+ \cap P_{\epsilon_2}} \max\Big(1 - \alpha - \lambda \|x - \pi_U(x)\|, 1 - \lambda \|x - \pi_{C_1(h_1)^{\complement}}(x)\|\Big)\ d\mu_1(x) \\
&+ \nu_{\text{-}1}\mu_{\text{-}1}\big(U^+ \cap P_{\epsilon_2}\big) + q_1\mu_1\big(U^-\big) \\
&+ \nu_{\text{-}1} \int\limits_{U^-} \max\Big(0, 1 - \lambda \|x - \pi_{C_{\text{-}1}(h_1)^{\complement} \setminus U}(x)\|, \alpha - \lambda \|x - \pi_U(x)\|\Big) d\mu_{\text{-}1}(x).
\end{aligned}$$

Note that we need to take into account the special case of the points in the dilation that were already in the attacked zone before, and that can now be attacked in two ways, either by projecting on U – but that works with probability α, since the classification on U is now randomized – or by projecting on $C_1(h_1)^{\complement}$, which works with probability 1 but may use more distance and so pay more penalty. We can now compute the difference between both scores.

$$\mathcal{R}^{\Omega_{\text{norm}}}_{\text{adv}}(h_1, \boldsymbol{\phi}) - \mathcal{R}^{\Omega_{\text{norm}}}_{\text{adv}}(m^{\boldsymbol{q}}_{\boldsymbol{h}}, \boldsymbol{\phi}') \tag{3.46}$$
$$= q_1 \int\limits_{U} 1 - \lambda \|x - \pi_{C_1(h_1)^{\complement}}(x)\| - \max\Big(1 - \alpha, 1 - \lambda \|x - \pi_{C_1(h_1)^{\complement}}(x)\|\Big) d\mu_1(x) \tag{3.47}$$
$$+ \nu_{\text{-}1} \int\limits_{U} 1 - \max(\alpha, 1 - \lambda \|x - \pi_{U^+}(x)\|) d\mu_{\text{-}1}(x) \tag{3.48}$$

$$- q_1 \int_{U^+ \setminus P_{\epsilon_2}} \max(1 - \alpha - \lambda \|x - \pi_U(x)\|, 0) d\mu_1(x) \tag{3.49}$$

$$+ q_1 \int_{U^+ \cap P_{\epsilon_2}} 1 - \lambda \|x - \pi_{C_1(h_1)^\complement}(x)\|$$

$$- \max\Big(1 - \alpha - \lambda \|x - \pi_U(x)\|, 1 - \lambda \|x - \pi_{C_1(h_1)^\complement}(x)\|\Big) d\mu_1(x) \tag{3.50}$$

$$+ \nu_{-1} \int_{U^-} 1 - \lambda \|x - \pi_U(x)\|$$

$$- \max\Big(0, 1 - \lambda \|x - \pi_{C_{-1}(h_1)^\complement \setminus U}(x)\|, \alpha - \lambda \|x - \pi_U(x)\|\Big) d\mu_{-1}(x). \tag{3.51}$$

Let us simplify Equation (3.46) using using additional hypothesis:

- First, note that Equation (3.48)> 0. Then a sufficient condition for the difference to be strictly positive is to ensure that other lines are ≥ 0.

- In particular to have (3.47) ≥ 0 it is sufficient to have for all $x \in U$

$$\max\Big(1 - \alpha, 1 - \lambda \|x - \pi_{C_1(h_1)^\complement}(x)\|\Big) = 1 - \lambda \|x - \pi_{C_1(h_1)^\complement}(x)\|.$$

  This gives us $\alpha \geq \lambda(\epsilon_2 - \delta) \geq \lambda \max_{x \in U} \|x - \pi_{C_1(h_1)^\complement}(x)\|$.

- Similarly, to have (3.49) ≥ 0, we should set for all $x \in U^+ \setminus P_{\epsilon_2}$

$$\alpha \geq 1 - \lambda \|x - \pi_U(x)\|.$$

  Since $\min_{x \in U^+ \setminus P_{\epsilon_2}} \|x - \pi_U(x)\| = \delta$, we get the condition $\alpha \geq 1 - \lambda\delta$.

- Finally (3.51) ≥ 0, since by definition of U^-, for any $x \in U^-$ we have

$$\|x - \pi_{C_{-1}(h_1)^\complement \setminus U}(x)\| \geq \|x - \pi_U(x)\|.$$

Finally, by summing all these simplifications, we have (3.46) > 0. Hence the result hold for any $\alpha > \max(1 - \lambda\delta, \lambda(\epsilon_2 - \delta))$ □

3.3.4 Improving a base classifier via randomization

Based on Theorem 8 we devise a new procedure (Algorithm 2), called Boosted Adversarial Training (BAT) to construct robust classifiers. It is based on three core principles: Adversarial Training, Boosting and Randomization.

Algorithm 1: Boosted Adversarial Training

Input : n the number of classifiers, D the training data set and α the weight update parameter.

Create and adversarially train h_1 on D

$\mathbf{h} = (h_1)$; $\mathbf{q} = (1)$

for $i = 2, \ldots, n$ **do**

  Generate the adversarial data set $\tilde{D}$ against $m_{\mathbf{h}}^{\mathbf{q}}$.

  Create and naturally train h_i on $\tilde{D}$

  $q_k \leftarrow (1-\alpha)q_k \quad \forall k \in [i-1]$

  $q_i \leftarrow \alpha$

  $\mathbf{q} \leftarrow (q_1, \ldots, q_i)$

  $\mathbf{h} \leftarrow (h_1, \ldots, h_i)$

end

return $m_{\mathbf{h}}^{\mathbf{q}}$

Contrary to classical algorithms such as *Fictitious play* that also generates mixtures of classifiers, and whose theoretical guarantees rely on the existence of a Mixed Nash Equilibrium, the performance of our algorithm is ensured by Theorem 8 to be at least as good as the classifier it uses as a basis. Moreover, the implementation of Fictitious Play would be impractical on high dimensional dataset we consider, due to computational costs.

Given a dataset D and a weight update parameter $\alpha \in [0, 1]$, BAT starts by constructing an adversarially trained classifier on D, and gives it a weight of 1. Then, at each step of the algorithm, we train a new classifier on a data set $\tilde{D}$ built from D that contains adversarial examples created to fool the current mixture. This new classifier is added to the mixture with a weight of α. Previous weights are then multiplied by $1 - \alpha$.

At each step, we use ℓ_∞**-PGD** with 20 iterations and $\epsilon_\infty = 0.031$ to attack the current mixture and build the adversarial dataset $\tilde{D}$. We choose this attack to fairly compare against Adversarial Training, which uses it during the training procedure.

On evaluating against ℓ_∞-PGD We use Expectation over Transformation (EOT) following [Athalye et al., 2018] and [Carlini et al., 2019], when implementing an ℓ_∞**-PGD** attack against a mixture of classifier. Indeed, it is important to compute the expected loss over the mixture, so that the attack optimizes Equation (3.40). Previous works such

Training method	Natural	ℓ_∞-**PGD**	ℓ_2-**C&W 0.4**	ℓ_2-**C&W 0.6**	ℓ_2-**C&W 0.8**
Natural	0.88	0.00	0.00	0.00	0.00
[Madry et al., 2018]	0.83	0.42	0.67	0.60	0.51
BAT ($n = 10, \alpha = 0.06$)	0.80	**0.58**	**0.70**	**0.65**	**0.59**

Table 3.1: Evaluation on CIFAR10 without *data augmentation*. Accuracy under attack of a single classifier adversarially trained and the mixture formed with our Algorithm 2. The evaluation is made with ℓ_∞-**PGD** and ℓ_2-**C&W** attacks both computed with 100 steps. For ℓ_∞-**PGD** we use an epsilon equal to $8/255$ (≈ 0.031), a step size equal to $2/255$ (≈ 0.008) and we allow random initialization. For ℓ_2-**C&W** we use a learning rate equal to 0.1, 9 binary search steps, the initial constant to 0.001, we allow the abortion when it has already converged and we give the results for the different values of rejection threshold $\epsilon_2 \in \{0.4, 0.6, 0.8\}$. Since the mixture draws a classifier in $\mathbf{h}$ according to $\mathbf{q}$ to predict a class for each sample, we run 100 times the evaluation to compute the expected accuracy under attack of the mixture. The width of the 95% confidence interval is negligible (< 0.01). For this reason, we chose to omit it.

as [Dhillon et al., 2018] and [**?**] estimate the expected loss through a Monte Carlo sampling. Since we assume perfect information for the Adversary, it knows the exact distribution of the mixture. Hence it can directly compute the expected loss without using a sampling method.

We conduced a grid-search to evaluate the influence of α (see the supplementary material for more details). For the results we present here, the optimal α we found is equal to 0.06 for 10 classifiers. In Table 3.1 we compare the accuracy (on the CIFAR10 dataset [Krizhevsky and Hinton, 2009]) of Boosted and classical Adversarial Training under attack with ℓ_∞-**PGD** run for 100 iterations.

Results against ℓ_∞-PGD. We compute 100 steps of ℓ_∞-**PGD** for the attack at test time, while only 20 steps during the training. The idea behind this difference is that the Adversary may target only a few specific points, and so may have access to more computational power for attacks than the Defender that trains on the whole dataset. For a classifier to be fully robust, its loss of accuracy should be controlled when the attacks are strongest than what it was trained on.

As shown in Table 3.1, the mixture generated by BAT with 10 classifiers and $\alpha = 0.06$ outperforms adversarial training on all four attacks. This is already the case for 2 classifiers, which corroborates the result from Theorem 8. We refer the reader to the supplementary material for additional results on how the size of the mixture influences the performance.

On Evaluating against ℓ_2-C&W. Adversarial Training can also be used to defend against ℓ_2-**C&W**. We conducted experiments to evaluate whether the mixture constructed with BAT also outperforms it against this attack. Since the basic ℓ_2-**C&W** attack creates an unbounded perturbation on examples, we implemented the constraint from Equation 3.12 by checking at test time whether the ℓ_2-norm of the perturbation exceeds a certain threshold $\epsilon_2 \in \{0.4, 0.6, 0.8\}$. If this holds, we keep the natural example, instead of its adversary version.

For the attacks to be comparable, the radiuses of the balls must be chosen carefully. For CIFAR10, which is a $3 \times 32 \times 32$ dimensional space, this gives $\epsilon_2 = 0.8$ and $\epsilon_\infty = 0.03$. The results of this evaluation are presented in Table 3.1. Note that we ran 100 steps for the ℓ_2-**C&W** as well.

Results against L_2-C&W. The accuracy under attack of our mixture is higher than that of adversarial training for all the thresholds. Our mixture is especially more robust than Adversarial Training when the threshold (*i.e. the budget for a perturbation*), is high. Here again, we see that with two classifiers the mixture already gives an accuracy under attack of 0.53 against ℓ_2-**C&W** with $\epsilon_2 = 0.8$ and outperforms Adversarial Training. This result also corroborates Theorem 8.

3.3.5 Implementation details

In this section, we consider $\mathcal{X} = [0,1]^{3\times32\times32}$ to be the set of images, and $\mathcal{Y} = \{1, ..., 10\}$ or $\mathcal{Y} = \{1, ..., 100\}$ according to the dataset at hand.

Adversarial attacks Let $(x, y) \sim D$ and $h \in \mathcal{H}$. We consider the following attacks:

(i) ℓ_∞-PGD attack. In this scenario, the Adversary maximizes the loss objective function, under the constraint that the ℓ_∞ norm of the perturbation remains bounded by some value ϵ_∞. To do so, it recursively computes:

$$x^{t+1} = \Pi_{B_{\|.\|}(x,\epsilon_\infty)}\left[x^t + \beta \text{sign}\left(\nabla_x \mathcal{L}\left(h\left(x^t\right), y\right)\right)\right] \tag{3.52}$$

where $\mathcal{L}$ is some differentiable loss (such as the cross-entropy), β is a gradient step size, and Π_S is the projection operator on S. One can refer to [Madry et al., 2018] for implementation details.

(ii) ℓ_2-C&W attack. In this attack, the Adversary optimizes the following objective:

$$\arg\min_{\tau \in \mathcal{X}} \|\tau\|_2 + \lambda \times \text{cost}(x + \tau) \tag{3.53}$$

where $\text{cost}(x+\tau) < 0$ if and only if $h(x+\tau) \neq y$. The authors use a change of variable $\tau = \frac{1}{2}(\tanh(w) - x + 1)$ to ensure that $x + \tau \in \mathcal{X}$, a binary search to optimize the constant λ, and Adam or SGD to compute an approximated solution. One should refer to [Carlini and Wagner, 2017b] for implementation details.

Datasets. To illustrate our theoretical results we did experiments on the **CIFAR10** and **CIFAR100** datasets. See [Krizhevsky et al., 2009] for more details.

Classifiers. All the classifiers we use are WideResNets (see [Zagoruyko and Komodakis, 2016]) with 28 layers, a widen factor of 10, a dropout factor of 0.3 and LeakyRelu activations with a 0.1 slope.

Natural Training. To train an undefended classifier we use the following hyperparameters.

- **Number of Epochs:** 200
- **Batch size:** 128
- **Loss function:** Cross Entropy Loss
- **Optimizer :** SGD algorithm with momentum 0.9, weight decay of 2×10^{-4} and a learning rate that decreases during the training as follows:

$$lr = \begin{cases} 0.1 & \text{if} \quad 0 \leq \text{epoch} < 60 \\ 0.02 & \text{if} \quad 60 \leq \text{epoch} < 120 \\ 0.004 & \text{if} \quad 120 \leq \text{epoch} < 160 \\ 0.0008 & \text{if} \quad 160 \leq \text{epoch} < 200 \end{cases}$$

Adversarial Training. To adversarially train a classifier we use the same hyperparameters as above, and generate adversarial examples using the ℓ_∞-**PGD** attack with 20 iterations. When considering that the input space is $[0, 255]^{3\times32\times32}$, on **CIFAR10** and **CIFAR100**, a perturbation is considered to be imperceptible for $\epsilon_\infty = 8$. Here, we consider $\mathcal{X} = [0, 1]^{3\times32\times32}$ which is the normalization of the pixel space $[0.255]^{3\times32\times32}$. Hence, we choose $\epsilon_2 = 0.031$ ($\approx 8/255$) for each attack. Moreover, the step size we use for ℓ_∞-**PGD** is 0.008 ($\approx 2/255$), we use a random initialization for the gradient descent and we repeat the procedure three times to take the best perturbation over all the iterations *i.e* the one that maximises the loss. For the ℓ_∞-**PGD** attack against the mixture $m^{\mathbf{q}}_{\mathbf{h}}$, we use the same parameters as above, but compute the gradient over the loss of the expected logits.

Evaluation Under Attack. At evaluation time, we use 100 iterations instead of 20 for **Adaptive-ℓ_∞-PGD**, and the same remaining hyperparameters as before. For the **Adaptive-ℓ_2-C&W** attack, we use 100 iterations, a learning rate equal to 0.01, 9 binary search steps, and an initial constant of 0.001. We give results for several different values of the rejection threshold: $\epsilon_2 \in \{0.4, 0.6, 0.8\}$.

Computing Adaptive-ℓ_2-C&W on a mixture To attack a randomized model, it is advised in the literature [Tramer et al., 2020] to compute the expected logits returned by this model. However this advice holds for randomized models that return logits in the same range for a same example (*e.g.* classifier with noise injection). Our randomized model is a mixture and returns logits that depend on selected classifier. Hence, for a same example, the logits can be very different. This phenomenon made us notice that for some example in the dataset, computing the expected loss over the classifier (instead of the expected logits) performs better to find a good perturbation (it can be seen as computing the expectation of the logits normalized thanks to the loss). To ensure a fair evaluation of our model, in addition of using EOT with the expected logits, we compute in parallel EOT with the expected loss and take the perturbation that maximizes the expected error of the mixture. See the submitted code for more details.

Library used. We used the Pytorch and Advertorch libraries for all implementations.

Machine used. 6 Tesla V100-SXM2-32GB GPUs

Sanity checks for Adaptive attacks In [Tramer et al., 2020], the authors give a lot of sanity checks and good practices to design an Adaptive attacks. We follow them and here are the information for **Adaptive-ℓ_∞-PGD** :

- We compute the gradient of the loss by doing the expected logits over the mixture.
- The attack is repeated 3 times with random start and we take the best perturbation over all the iterations.
- When adding a constant to the logits, it doesn't change anything to the attack
- When doing 200 iterations instead of 100 iterations, it doesn't change the performance of the attack
- When increasing the budget ϵ_∞, the accuracy goes to 0, which ensures that there is no gradient masking. Here are some values to back this statement:

Epsilon	0.015	0.031	0.125	0.250
Accuracy	0.638	0.546	0.027	0.000

Table 3.2: Evolution of the accuracy under **Adaptive-ℓ_∞-PGD** attack depending on the budget ϵ_∞

- The loss doesn't fluctuate at the end of the optimization process.

Selecting the first element of the mixture. Our algorithm creates classifiers in a boosting fashion, starting with an adversarially trained classifier. There are several ways of selecting this first element of the mixture: use the classifier with the best accuracy under attack (option 1, called bestAUA), or rather the one with the best natural accuracy (option 2). Table 3.3 compares both options.

Beside the fact that any of the two mixtures outperforms the first classifier, we see that the fisrt option always outperforms the second. In fact, when taking option 1 (bestAUA = True) the accuracy under ℓ_∞**-PGD** attack of the mixture is 3% better than with option 2 (bestAUA = False). One can also note that both mixtures have the same natural accuracy (0.80), which makes the choice of option 1 natural.

Training method	base NA	base AUA	mixture NA	mixture AUA
BAT (bestAUA=True)	0.77	**0.46**	**0.80**	**0.55**
BAT (bestAUA=False)	**0.83**	0.42	**0.80**	0.52

Table 3.3: Comparison of the mixture that has as first classifier the best one in term of natural accuracy and the mixture that has as base classifier the best one in term of Accuracy under attack. The accuracy under attack is computed with the ℓ_∞**-PGD** attack. NA means matural accuracy, and AUA means accuracy under attack.

3.3.6 Extension to more than two classifiers

As we mentioned earlier, a mixture of more than two classifiers can be constructed by adding at each step t a new classifier trained naturally on the dataset $\tilde{D}$ that contains adversarial examples against the mixture at step $t-1$. Since $\tilde{D}$ has to be constructed from a mixture, one would have to use an adaptive attack as **Adaptive-ℓ_∞-PGD**. Here is the algorithm for the extented version :

Algorithm 2: Boosted Adversarial Training

Input : n the number of classifiers, D the training data set and α the weight update parameter.

Create and adversarially train h_1 on D

$\mathbf{h} = (h_1)$; $\mathbf{q} = (1)$

for $i = 2, \ldots, n$ **do**

  Generate the adversarial data set $\tilde{D}$ against $m_{\mathbf{h}}^{\mathbf{q}}$.

  Create and naturally train h_i on $\tilde{D}$

  $q_k \leftarrow (1-\alpha)q_k \quad \forall k \in [i-1]$

  $q_i \leftarrow \alpha$

  $\mathbf{q} \leftarrow (\alpha, \ldots, q_i)$

  $\mathbf{h} \leftarrow (h_1, \ldots, h_i)$

end

return $m_{\mathbf{h}}^{\mathbf{q}}$

Here to find the parameter α, the grid search is more costly. In fact in the two-classifier version we only need to train the first and second classifier without taking care of α, and then test all the values of α using the same two classifier we trained. For the extended version, the third classifier (and all the other ones added after) depends on the first classifier, the second one and their weights $1-\alpha$ and α. Hence the third classifier for a certain value of α can't be use for another one and, to conduct the grid search, one have to retrain all the classifiers from the third one. Naturally the parameters α depends on the number of classifiers n in the mixtures.

3.4 Stability of Nash equilibria

In the case where Nash equilibria exist, the natural question to ask is whether either player can actually compute its optimal strategy. This is a trivial question for discrete zero-sum game, but in our case both players play continuous strategies from an infinite set of possibilities. This means that players will only control their choice of strategy up to some (ideally small) confidence interval. Typically, both the classifier and the attack are computed using some variant of gradient descent on a loss function, to get as close as possible to some local optimum. Thus, we need to know if sequences of attack and defenses have any chance to converge to the equilibrium.

3.4.1 Stability to a perturbation of the attack

In this context, an equilibrium will be stable when a small variation from one of the players around the equilibrium strategy does not change the best response of the other. In other

words, both strategies are optimal not only against their best response, but also in a small region around it. A gradient descent algorithm would only need to get into that optimality region to find the equilibrium. To formalize this notion of stability, we first need a notion of "distance" between strategies. We will focus in the rest of the section on the Attacker side.

Definition 41 (Metrics on the attacks)**.** *Let $\phi, \psi : \mathcal{X} \to \mathcal{X}$, and μ a probability measure over $\mathcal{X}$. We define two families of distances between these two functions with regards to μ :*

$$d_\mu(\phi, \psi) = \mathbb{E}_{X\sim\mu}\left[\mathbb{1}_{\phi(X)\neq\psi(X)}\right]$$

$$d_{d_1}(\phi, \psi) = \mathbb{E}_{X\sim\mu}[d_1(\phi(x), \psi(x))]$$

where d_1 is a metric on $\mathcal{X}$.

The first metric counts the average number of points that are moved differently between both attacks, whereas the second ponders it by how far away from each other both attacks move the same point, according to some distance d_1 over the input space.

Proposition 3. *d_μ and d_{d_1} are both distances over $\mathcal{X}$.*

Proof. The result is immediate for d_{d_1} as d_1 is a distance. For d_μ, we have :

- $d_\mu(\phi, \phi) = 0$.
- $d_\mu(\phi, \psi) = d_\mu(\psi, \phi)$
- $\mathbb{1}_{\phi(x)\neq\psi(x)} \leq \mathbb{1}_{\phi(x)\neq\tilde{\phi}(x)} + \mathbb{1}_{\tilde{\phi(x)}\neq\psi(x)}$ for any $x \in \mathcal{X}$ and attack $\tilde{\phi}$. Hence : $d_\mu(\phi, \psi) \leq d_\mu(\phi, \tilde{\phi}) + d_\mu(\tilde{\phi}, \psi)$.
- if $d_{mu}(\phi, \psi) = 0$, then $\phi = \psi$ $\mu-$almost everywhere.

□

This allows us to give a formal definition for Attacker-stability :

Definition 42 ($\delta-$Attacker stability)**.** *We say that a Nash Equilibrium (ϕ^*, h^*) is $\delta-$Attacker stable if for any attack ψ such that $d_\mu(\phi^*, \psi) \leq \delta$, we have $h^* \in \mathcal{BR}(\psi)$*

This means that if the Adversary does a small variation of its attack, moving less than a mass δ of points differently, the Defender has no incentive to change its strategy. An even stronger definition of stability would also require $\psi \notin \mathcal{BR}(h^*)$, i.e. that the Attacker gets a worse score after perturbation (see Definition 19). This would ensure that a local optimization procedure would bring him back to the equilibrium. However, as we will show, even the lighter version of **??** is not met in the deterministic regime.

3.4.2 Nash equilibria cannot be stable in the deterministic regime

Using the definition from the previous section, we can now show our first main result about the stability of pure Nash equilibria :

Theorem 9 (No stable Pure Nash Equilibrium)**.** *We consider either the* $0/1$ *loss or any convex Bayes consistent surrogate. Let* μ *be a probability measure of infinite support. Let* (ϕ^*, h^*) *be a Pure Nash Equilibrium for* μ*. Then it cannot be* $\delta-$*Attacker stable for any* $\delta > 0$*.*

The idea behind this theorem is that small changes in the attack are enough to create a hole in one of the conditional distributions, i.e. a zone of measure zero. This forces the Bayes classifier to change its value on the hole, so that we leave the equilibrium.

We will now show an important property of pure Nash equilibria, namely that if the optimal attack creates a hole in a zone H that was classified 1 for the distribution μ_1, then it must also create it for μ_{-1}. The intuition behind this result is that otherwise μ_{-1} would dominate μ_1 on H after attack, forcing the Bayes classifier to change its classification on that zone. This is not possible, since by definition of the Nash Equilibrium the initial classifier should be a best response to its optimal attack.

Lemma 4. *Let* $H \subset C_1(h^*)$*. Then if* $\mu_1 \# \phi_1^*(H) = 0$*, then we also have* $\mu_{-1} \# \phi_{-1}^*(H) = 0$*.*
The same result is true when swapping 1 *and* -1*.*

Proof. By contradiction, if there is H such that $\begin{cases} \mu_1 \# \phi_1^*(H) & = 0 \\ \mu_{-1} \# \phi_{-1}^*(H) & \neq 0 \end{cases}$ then h^* will classify H as -1 (else changing it to -1 would give it a better score under attack, which is contradictory with its optimality). This is not possible, because $H \subset C_1(h^*)$. Contradiction. □

We will now show that a small perturbation of the optimal attack can create a hole in one of the distributions and not the other. For that, the first step is to ensure that there is a zone of nonzero measure after attack, where a hole can be created.

Lemma 5. *There is a* $M > 0$ *and* $i \in \{-1, 1\}$ *such that* $H = B(0, M) \cap \mathcal{C}_i$ *has positive measure under* $\mu_i \# \phi_i^*$ *and* $\mu_{-i} \# \phi_{-i}^*$.

Proof. First of all, if $\mu_1 \# \phi_1^*(\mathcal{C}_1) = 0$, then this means that $\mu_{-1} \# \phi_{-1}^*(\mathcal{C}_{-1}) = 1$. So there is a i such that $\mu_i \# \phi_i^*(\mathcal{C}_i) \neq 0$. Let us assume without loss of generality that this i is 1.

We then reason ad absurdum and suppose that for any $M > 0$,
$\mu_1 \# \phi_1^*(B(0, M) \cap \mathcal{C}_1) = 0$. Then, by $\sigma-$additivity of $\mu_1 \# \phi_1^*$, we have :

$$\mu_1 \# \phi_1^* \left(\bigcup_{M=0}^{\infty} (B(0, M) \cap \mathcal{C}_1) \right) \leq \sum_{M=0}^{\infty} (\mu_1 \# \phi_1^*(B(0, M) \cap \mathcal{C}_1)) = \sum_{M=0}^{\infty} 0 = 0.$$

But the left hand term is equal to $\mu_1 \# \phi_1^*(\mathcal{C}_1)$, which we shown to be non-zero. Contradiction. □

We then show that we can find a zone of positive measure, and of diameter small enough to be emptied by an attack.

Lemma 6. *For* $\epsilon < \min(M, \epsilon_{adv})$*, there exists a subset* $H \subset \mathcal{C}_1$ *such that :*

- $\mathrm{diam}(H) < \epsilon$
- $\mu_1 \# \phi_1^*(H) > 0$.

Proof. $B(0, M)$ is compact in $\mathcal{X}$ which is of finite dimension. Hence we can extract a finite covering of the balls $B(x, \frac{\epsilon}{2})$ for $x \in B(0, M)$, namely $\left\{(B(x_i, \frac{\epsilon}{2}), i = 1..m\right\}$. We have $\mathcal{C}_1 \cap B(0, M) \subset \bigcup_{i=1}^{m} \left(B(x_i, \frac{\epsilon}{2}) \cap \mathcal{C}_1\right)$, so by $\sigma-$additivity of $\mu_1 \# \phi_1^*$, one of these sets must have a nonzero measure. Hence there exists i_0 such that :

$$\mu_1 \# \phi_1^*(H) > 0$$

where $H = B(x_{i_0}, \frac{\epsilon}{2}) \cap \mathcal{C}_1$. And H has a diameter of at most ϵ by definition. □

We can now proove the main theorem :

proof of theorem 9. Let $H \subset \mathcal{C}_1$ such that $\mu_i \# \phi_i^*(H) > 0$ for both i=1 and -1, and so that $\mathrm{diam}(H) < \epsilon$. Since μ is of infinite support, we can (by replacing H by one of its subsets if needed) assume that $\mu(H) < \delta$ as well.

Then for every $x \in H$, there exists $z_x \notin H$ such that $d(x, z_x) \leq \epsilon$. Let us construct a new attack function ψ :

$$\psi_1(x) = \left| \begin{array}{cc} \phi_1^*(x) & \text{for } x \notin H \\ z_x & \text{for } x \in H. \end{array} \right.$$

And $\psi_{-1} = \phi_{-1}^*$

Then we have $d_\mu(\phi^*, \psi) < \delta$ since the only points moved differently are on H. Let us now show that $h^* \notin \mathcal{BR}(\psi)$, which will give the desired result.

We have $\mu_1 \# \psi_1^*(H) = 0$ by definition of ψ, and $\mu_{-1} \# \psi_{-1}^*(H) = 0$. If h^* was a best response to ψ, it would thus classify almost all points of H as -1, else changing the classification to -1 would strictly improve the score. But since $H \subset \mathcal{C}_1$, this is not the case. Hence the result. □

3.4.3 A more granular criterion : the instability factor

Our current definition of stability is useful as a criteria to evaluate whether an equilibrium is realistic or not, but does not provide a "degree of stability" for non-stable equilibria, in order to compare them and evaluate how different modifications of the game lead to more or less stability. We therefore introduce the instability factor of an equilibrium, and show that introducing randomization on the Defender's side (and constraining to it) leads to an increase in stability.

Definition 43 (instability factor)**.** *A δ-instability factor of a Nash equilibrium $(h^*, \boldsymbol{\phi}^*)$ is the maximum variation of measure that can be incurred from a variation of the attack of size δ. More formally, we say that the Nash equilbrium has a $\delta-$instability factor $K > 0$ if for every attack $\boldsymbol{\psi}$ such that, for $i = \pm 1$, $d_\mu(\phi_i^*, \psi_i) \leq \delta$, we have :*

$$\forall H \subset \mathcal{X}, |\psi_i \# \mu_i(H) - \phi_i^* \# \mu_i(H)| \leq K \delta \mu(H)$$

In particular, if $\psi_i \# \mu_i$ and $\phi_i^* \# \mu_i$ have densities p_i and $\tilde{p_i}$, we have :

$$\forall x \in \mathcal{X}, |p_i(x) - \tilde{p_i}(x)| \leq K\delta \tag{3.54}$$

Proposition 4. *If an equilibrium has a $\delta-$instability factor of 0, then it is $\delta-$Attacker stable.*

Proof. This result is immediate. The instability factor of 0 means that a variation of the attack of size less then δ cannot change the transported measure. It follows that the Bayes classifier will stay the same. □

This definition provides us with a less strict criterion for stability. Typically, a very small instability factor means that perturbating the attack will only incur a very small modification of the transported measure, and can only shift the Bayes classifier on the regions where the uncertainty was already very strong.

For deterministic classifiers, the instability factor will always be 1. That is because it is always possible to empty a zone of measure δ for one of the two conditional distributions by modifying the optimal attack.

Proposition 5 (instability factor for a pure Nash equilibrium)**.** *A pure Nash equilibrium always has a instability factor of 1.*

Proof. During the proof of Theorem 9, we showed that the instability factor cannot be lower than 1, since we can always use an attack of size δ to empty a zone of size δ in one of the distributions, effectively reducing $\phi_i^* \# \mu_i$ by δ on that zone. We will now show that the instability factor is exactly one. Let ψ be such that $d_\mu(\psi, \phi^*) \leq \delta$. Let $H \subset \mathcal{X}$. Let us show that $|\psi_i \# \mu_i(H) - \phi_i^* \# \mu_i(H)| \leq \delta \mu_i(H)$.
Let us call $\mathcal{Z} = \{x \in \mathcal{X} | \psi_i(x) \neq \phi_i^*(x)\}$. Then $\mu_i(Z) \leq \delta$ by hypothesis.
Furthermore, we have, by definition of $\mathcal{Z}$, $\psi_i^{-1}(H) \setminus \phi_i^{*^{-1}}(H) \subset \mathcal{Z}$. It follows :

$$\begin{aligned}
\psi_i \# \mu_i(H) &= \mu_i(\psi_i^{-1}(H)) \\
&= \mu_i(\psi_i^{-1}(H) \cap \phi_i^{*^{-1}}(H)) + \mu_i(\psi_i^{-1}(H) \setminus \phi_i^{*^{-1}}(H)) \\
&\leq \mu_i(\phi_i^{*^{-1}}(H)) + \mu_i(\mathcal{Z}) \\
&\leq \mu_i(\phi_i^{*^{-1}}(H)) + \delta
\end{aligned}$$

An exactly symmetric computation gives us $\mu_i(\phi_i^{*^{-1}}(H)) \leq \mu_i(\psi_i^{-1}(H)) + \delta$, and by combining both :

$$-\delta \leq \mu_i(\phi_i^{*^{-1}}(H)) - \mu_i(\psi_i^{-1}(H)) \leq \delta$$

Which is the desired result. □

We will now show that when altering the class of possible strategies for the Defender by allowing a particular type of randomization, we can reach much more stable Nash equilibria.

3.4.4 Noise injection for the Defender stabilizes Nash equilibria, at the price of accuracy

In this subsection, we will restrict the Defender to the class of Noise-Injected classifiers.

Definition 44 (Noise-injected classifier)**.** *Let h be a deterministic classifier, q_0 be some isotropic probability density function. The corresponding Noise-injected classifier of distribution q_0 takes the average of h over the distribution q_0 :*

$$h_{q_0} : x \mapsto \int_{\mathcal{X}} h(z) q_0(z - x) d\mu(x)$$

We call the class of Noise-injected classifiers $\mathcal{NI}$.

This class of function preserves some of the most trivial Nash Equilibria, such as the one described in Section 3.2.2, but contributes to stabilize them. In particular, their instability factor is linked to ϵ and the Lipschitz constant of the noise distribution.

Proposition 6 (Instability factor for Noise-Injected classifiers)**.** *Let q_0 be some isotropic probability density function, L its Lipschitz constant, and $\epsilon > 0$. Then in the zero-sum game with attacks of size at most ϵ, all Nash equilibria have a instability factor of $\|L\|_\infty \epsilon$.*

Remark 2. *For a normal distribution $\mathcal{N}(0, \sigma^2 Id)$, the Lipschitz constant is*

$$L = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\right) \simeq \frac{0.24}{\sigma}$$

We can see two important things from this :

- *For $\sigma = 1$ and $\epsilon = 0.5$, this gives us a instability factor of 0.12, which is much smaller than 1. Adding noise has considerably increased the stability of the equilibrium.*
- *There is a **Stability-Accuracy tradeoff** : as σ increases, the instability factor decreases, but so does the natural accuracy. By taking noises of large variance, we can*

reach arbitrarily high levels of stability, but at the cost of global score. This phenomenon is general to noise injection, as increasing the variance of the noise will always decrease the Lipschitz constant of the pdf.

Proof. Let (h_{q_0}, ϕ) be a Nash equilibrium, and ψ such that $d_\mu(\phi, \psi) \leq \delta$. Let $\mathcal{Z}_i = \{x \in \mathcal{X} | \psi_i(x) \neq \phi_i(x)\}$. Note that $\mu_i(\mathcal{Z}_i) \leq \delta$ Then we have :

$$
\begin{aligned}
|\phi_i \# \mu_i(H) - \psi_i \# \mu_i(H)| &= \left| \sum_{i=\pm 1} q_i \int_{\mathcal{X}} \left(\int_{\mathcal{X}} q_0(z - \phi_i(x)) d\mu_i(x) - \int_{\mathcal{X}} q_0(z - \psi_i(x)) d\mu_i(x) \right) d\mu_i(z) \right| \\
&\leq \sum_{i=\pm 1} q_i \int_{\mathcal{X}} \left(\left| \int_{\mathcal{X}} q_0(z - \phi_i(x)) d\mu_i(x) - \int_{\mathcal{X}} q_0(z - \psi_i(x)) d\mu_i(x) \right| \right) d\mu_i(z) \\
&\leq \sum_{i=\pm 1} q_i \int_{\mathcal{X}} \left(\int_{\mathcal{X}} L |\phi_i(x) - \psi_i(x)| d\mu_i(x) \right) d\mu_i(z) \\
&\leq \sum_{i=\pm 1} q_i \int_{\mathcal{X}} \left(\int_{\mathcal{Z}_i} L |\phi_i(x) - \psi_i(x)| d\mu_i(x) \right) d\mu_i(z) \\
&\leq \sum_{i=\pm 1} q_i \int_{\mathcal{X}} \left(\int_{\mathcal{Z}_i} L \epsilon d\mu_i(x) \right) d\mu_i(z) \\
&\leq \sum_{i=\pm 1} q_i \int_{\mathcal{X}} L \epsilon \mu_i(\mathcal{Z}_i) d\mu_i(z) \\
&\leq L \epsilon \delta \mu(H)
\end{aligned}
$$

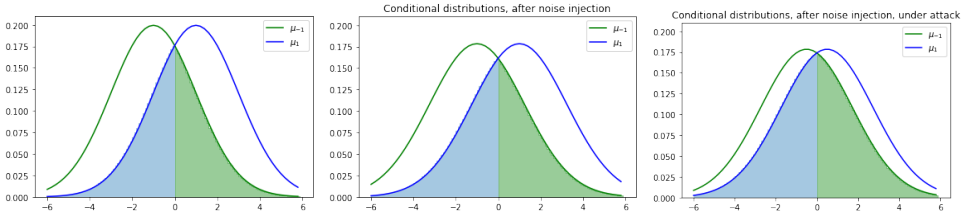
Hence the desired result. □

3.4.5 Empirical visualization of the accuracy/stability tradeoff

Let us take the example of two normal distributions : $\mu_{-1} = \mathcal{N}(-1, 4)$ and $\mu_1 = \mathcal{N}(1, 4)$ (see Figure 3.7), with similar prior probability $\frac{1}{2}$.

The Bayes classifier corresponding to these two distributions is simply $\mathbb{1}\{x \geq 0\}$. The corresponding adversarial risk is simply the average of the green and blue colored zone, namely $\mathbb{P}[\mathcal{N}(-1, 4) \geq 0]$ (here again by symmetry).

After injecting a noise $\mathcal{N}(0, \sigma^2)$, this risk becomes $\mathbb{P}[\mathcal{N}(-1, 4 + \sigma^2) \geq 0]$. The adversarial risk is simply the risk when both distributions are translated toward the decision



(a) initial distributions (b) after noise injection (c) under attack

Figure 3.7: 2 normal conditional distributions. The blue zone represents the conditional risk of class 1, and the green zone the conditional risk of class -1.

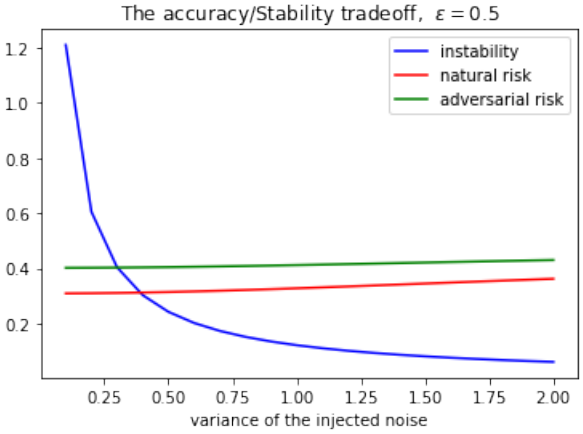


Figure 3.8: Natural and adversarial risk versus instability factor, for several values of σ

boundary, i.e. $\mathbb{P}[\mathcal{N}(-1+\epsilon, 4+\sigma^2) \geq 0]$ (see Section 3.2.2 for a more thorough analysis of the case of two normal conditional distributions).

We can thus quantify the accuracy/stability tradeoff for several values of σ. Here we took $\epsilon = 0.5$ as it is a classical value in adversarial learning problems.

3.5 Summary of our results

In Table 3.4, we summarize the state of current knowledge regarding the game theory perspective on adversarial attacks. We have shown several important non-existence and non-stability results in the deterministic case, highlighting the importance of randomization, and exhibited an accuracy/stability tradeoff in the case of a randomized Defender, showing that noise injection increases stability at the cost of accuracy.

In Appendix A, we also provide a general study of the case of randomized Attackers when using a surrogate loss function, and show in particular some conditions for the existence of a pure Nash equilibrium in this setting.

Regime | cost	0/1-loss	convex surrogate	stability ?
Pure | none	✓	✓	✗
Pure | positive	✗	✓	✗
Mixed Attacker | none	✓	✓	?
Mixed Attacker | positive	?	✓	?
Noise injection | none	?	?	A/S tradeoff
Random (both) | none |	✓ [Meunier et al., 2021]	?	?

Table 3.4: Summary of our current state of knowledge. The blue checkmark means that an equilibrium can exist / be stable, whereas the red cross means that it is not possible. The orange question mark means that the question is hard and remains open

Furthermore, we have shown that mixtures can strictly outperform any deterministic classifier, and provided a framework to implement such mixture (Boosted Adversarial Training).

3.5.1 Future works and open problems

As we have seen, several regimes remained to be studied for the existence and stability of equilibria. Furthermore, we identify three key steps that would be necessary to have a complete picture of the problem :

- **Studying different forms of randomization** We have shown the impact of noise injection (with uniform distributions) on stabilizing the equilibria. We believe that this is a consequence of randomization as a whole, and several other forms would benefit from being studied, such as different distributions for noise injection, finite mixtures, general convolutions, and so on.

- **Quantifying the duality gaps** We know for instance that the randomized regime provides a better performance than the deterministic one for the Defender's problem. But by how much ? To know that, we must quantify the difference between the optimal risk in the randomized and deterministic cases.

- **Computing the Nash equilibria** Knowing when equilibria exist and are stable is important to orient the research, and know which regimes are worth studying further. The next step however, is to compute such equilibria. This is a complex problem, as it encompasses for example the computation of transport plans for the Adversary. One possible lead would be to use a primal-dual algorithm such as Chambolle-Pock.

4 Background on randomized smoothing and certification

After studying the existence and computability of optimal classifiers in various setting, we will now focus on how to certify the performance of such classifiers. The current state of the art framework for obtaining guarantees of robustness under attack is called *randomized smoothing*. As it uses limited information on the classifier, it works with arbitrarily large networks. However, it has been recently shown that the method suffers from impossibility results, so that it cannot scale well to high-dimension problems. In Chapter 5, we will provide a deeper analysis of randomized smoothing, and show that these limitations can be bypassed by collecting more information.

In this chapter, we will first give a general presentation of Randomized smoothing (the method, how to derive certificates, and how to pre-train the classifier), then we will focus on computing the certificates, by analyzing the geometry of the Neyman-Pearson set. Finally, we will describe the impossibility results and their limitations.

4.1 Randomized smoothing

4.1.1 From differential privacy to certified robustness

Randomized smoothing was first introduced by [Lecuyer et al., 2018] as a provable defense. The inspiration came from the field of privacy preserving machine learning, which is the study of how much information a learning algorithm reveals on the data it has been trained on. Noise injection is often used in this field to ensure that this data cannot be reconstructed with only access to the output of the training algorithm.

Privacy in machine learning Several definitions of privacy exist, among which *differential privacy* appears as the current standard. It consists in bounding the maximal variation of the output of an algorithm when an element is added or removed from the training set.

Definition 45 ((ϵ, δ)-Differential privacy for machine learning)**.** *Let $\mathfrak{S}_n$ be the space of all data samples of size n, and $\mathcal{H}$ a class of hypotheses. A Learning algorithm T is a (possibly randomized) function which links a training sample $\mathcal{S} \in \mathfrak{S}_n$ to a hypothesis $h \in \mathcal{H}$. Then T is said to be (ϵ, δ)-differentially private if for any $\mathcal{S}, \mathcal{S}' \in \mathfrak{S}_n$ that only differ from one input-output pair, and any $h \in \mathcal{H}$, we have*

$$\mathbb{P}[T(\mathcal{S}) = h] \leq \exp(\epsilon)\mathbb{P}[T(\mathcal{S}') = h] + \delta$$

A major contribution from [Lecuyer et al., 2018] was to show that $(\epsilon, \delta)-$differential privacy also translates into a robustness guarantee. Hence, algorithms that guarantee differential privacy will naturally exhibit a provable certificate of robustness. The most classical example is injecting Gaussian noise on the input at test time. This can be done either by injecting the noise once, to obtain a randomized algorithm, or by sampling it several times and computing the average reaction of the classifier to the noise distribution.

This is the idea behind Randomized smoothing : starting from a base classifier h, we sample at every input x some noise Z from an isotropic distribution q_0, obtaining samples $x + Z_1, \ldots, x + Z_n$. We then classify each of these points, and perform a majority voting to return the most probable class.

Randomized smoothing Let $\mathcal{X} = \mathbb{R}^d$ be our input space and $\mathcal{Y} = \{0, 1\}$ our label space. Let $\mathcal{H}$ be the class of measurable functions from $\mathcal{X}$ to $\mathcal{Y}$, and $h \in \mathcal{H}$ be a base classifier. Randomized smoothing creates a new classifier h_{q_0} by averaging h under some probability density function q_0 over $\mathcal{X}$. When receiving an input $x \in \mathcal{X}$, we compute the probability that h takes value 1 for a point drawn from $q_0(\cdot - x)$:

$$p(x, h, q_0) = \int h(z) q_0(z - x)\, \mathrm{d}z$$

The smoothed classifier then returns the most probable class.

Definition 46 (Randomized smoothing)**.** *The q_0-randomized smoothing of h is the classifier:*

$$h_{q_0} : x \mapsto \mathbb{1}\left\{p(x, h, q_0) > \frac{1}{2}\right\}$$

[Lecuyer et al., 2018], and later [Li et al., 2018], have shown that it is possible to compute a lower bound L_{max} on the size of any successful attack against a randomized smoothed classifier. This provides a *robust region* around each input, in which no attack

can succeed. Furthermore, this certificate only requires information on the probabilities $p(x, h, q_0)$, and no prior knowledge on the classifier. This makes it the first provable defense that scales well to even large network architectures.

4.1.2 Deriving certificates : the Neyman-Pearson lemma

A stronger framework to derive certificates was then introduced by [Cohen et al., 2019]. Using the Neyman-pearson lemma, they compute the worst-case scenario which is consistent with the information $p(x, h, q_0)$, and use it to obtain a *tight* certificate.

Theorem 10 (Neyman-Pearson lemma)**.** *Let q_0 be a probability density functions. Let δ be an attack vector. Then for any $k > 0$, we define the Neyman-Pearson set:*

$$\mathcal{S}_k = \left\{u \in \mathbb{R}^d | q_0(x + u + \delta) \leq k q_0(x + u)\right\}$$

and the associated Neyman-Pearson function:

$$\Phi_k = \mathbb{1}\{\mathcal{S}_k\} \tag{4.1}$$

Then for any function $\Phi : \mathcal{X} \to [0, 1]$ such that $\int \Phi(u) q_0(x + u) d\mu \geq \int \Phi_k(u) q_0(x + u) d\mu$, we have:

$$\int \Phi_k(u) q_0(x + u + \delta) d\mu(u) \leq \int \Phi(u) q_0(x + u) d\mu(u)$$

This means that, among all classifiers that are consistent with the information $p(x, h, q_0)$ that we have gathered about the true classifier h, the Neyman-Pearson function Equation (4.1) is the one that provides the worst certificates. A certificate for the Neyman-Pearson function is thus a certificate for the true classifier. In general both the information and the certificate are computed using Monte-Carlo sampling, which can be costly. However, in the case of the Gaussian noise, this is not necessary.

Gaussian noise In general both the information and the certificate are computed using Monte-Carlo sampling, which can be costly. However, When using a normal distribution for the smoothing, the Neyman-Pearson set $\mathcal{S}_k$ is always a half-space, delimited by a hyperplane. This allows Cohen et Al. to compute a certificate which has a closed form, and is independent on the dimension of the problem :

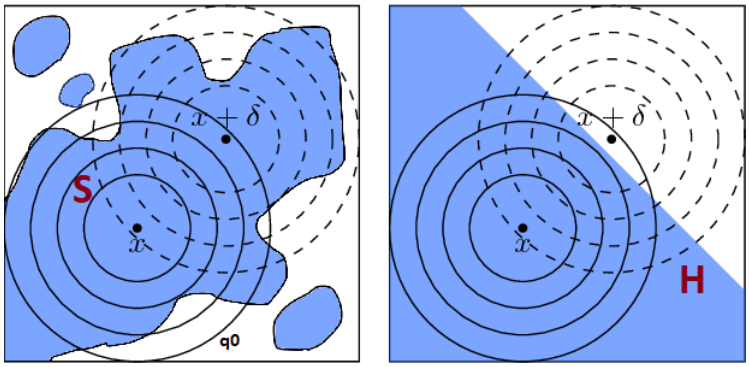


Figure 4.1: Certificate using a Gaussian noise distribution. We gather information around point x about the blue class S, using a distribution q_0. We then compute the Neyman-Pearson boundary, which is the hyperplane H orthogonal to $(x, x + \delta)$ such that the blue half-space it delimits has the same measure as the initial set S for the distribution q_0.

Theorem 11 (Certificate from [Cohen et al., 2019]). *Let $q_0 = \mathcal{N}(0, \sigma^2 I)$ and p_0 be such that $p(x, h, q_0) \geq p_0$. Then $h_{q_0}(x + \delta) = h_{q_0}(x)$ for all δ such that $\|\delta\|_2 < \sigma \Phi^{-1}(p)$.*
Where Φ is the CDF of the standard normal distribution.

4.1.3 Monte-Carlo sampling and confidence intervals

Using Theorem 10 provides us with a general framework to compute certificate:

1. Using Monte-Carlo sampling, compute a lower bound $p_0 \leq p(x, h, q_0)$;
2. Tune the parameter k (using a binary search) so that $p(x, \Phi_k, q_0) \leq p_0$ while being as close as possible, computing the expectation with Monte-Carlo sampling as well. This gives us the Neyman-Pearson function, i.e. the worst-case classifier. We then just need to compute the certificate for this classifier (which is known, so a simple Monte-Carlo sampling works).
3. Compute $\tilde{p} = p(x, h, q_0(\cdot + \delta)$ via Monte-Carlo sampling for several values of δ, to find the larger value of $\|\delta\|$ such that the probability remains $\geq \frac{1}{2}$. This gives the certification radius.

In the case of Gaussian noise, steps 2 and 3 are non-necessary, as the certificate can be computed directly from p_0 using Theorem 11. For all steps, computations are done through Monte-Carlo sampling, which induces randomness. We therefore must control

the variation of the values, to minimize the chances of obtaining a p that is higher than the real one for example, which would lead to a false certificate.

Confidence intervals For that, [Cohen et al., 2019] use a procedure called Predict, which consists in sampling a large number of points from distribution q_0, counting the number of occurences of each class, and deducing the relative probabilities of occurence p_i. This constitutes a sampling from a binomial distribution, whose parameter is the real probability p_0. They then compute a confidence interval for the value p_0 with a given precision $1 - \alpha$, and use that to compute the certificate. Thus, they know that with probability $1 - \alpha$, that certificate will be an underestimation of the real one, and so a valid robustness guarantee.

In Chapter 5, we will use similar confidence intervals for most of our implementations with neural networks.

Distribution shift and noise injection As described by [Lecuyer et al., 2018], injecting noise on a standard network usually leads to a near-zero natural accuracy, due to the fact that the classifier is not prepared to classify noised inputs. More formally, there is a distribution shift when adding noise: the conditional distributions change after convolution with q_0, and there is no reason that the Bayes classifier remains the same. In practice, this is solved by injecting noise as soon as the training step, to ensure that the network is calibrated on the convoluted distributions, and will perform correctly after smoothing.

This constitutes our first hint that the choice of a base classifier is of paramount importance for Randomized smoothing (contrary to the accepted idea in the community that the strength of the method comes from being classifier-agnostic). Noise injection during the training amounts to "tailoring" the base classifier specifically for the method. We will discuss that aspect further in Chapter 5.

4.2 Modern improvements to certificates

The framework described in the previous section made randomized smoothing a promising approach for certified defenses against adversarial attacks. However, several downsides remained :

- Outside of Gaussian noise, there is no easy way to compute certificates in high dimensions;
- Good certification radius seemed to incur high drops in natural accuracy;
- The performance remained way unsufficient for industrial applications, with no guarantees that improvements are possible.

Several papesr have then improved on that method, to address these issues.

4.2.1 Choice of the base classifier

SmoothAdversarialTraining Seeing the importance of noise injection at training time to avoid distribution shift, [Salman et al., 2019] have introduced the idea of *adversarially training the base classifier* to further improve its robustness. For that, they introduced SmoothAdv, an attack framework designed to work against randomized smoothed classifiers, so that the attacks could be incorporated during the training, and target the vulnerabilities of the smoothed network and not just the base one. This idea is very similar to that of the distribution shift, but here on the attacks. This allowed significant increases in the certified accuracy, but at the cost of higher training times, adversarial training being very costly.

Denoised Smoothing In a very different direction, [Salman et al., 2020] introduced denoised smoothing as a way to use randomized smoothing with any base classifier, without re-training it to include noise injection and adversarial training. The main idea was to add a denoiser as the final layer of the base classifier, to cancel the distribution shift. They use a standard denoiser (i.e. an algorithm D that is trained to minimize $V(D) := \mathbb{E}_{\delta\sim q_0}[D(x+\delta)-x]$), but modify the training objective to take into account the need to not perturbate the classification, which becomes :

$$\tilde{V}(D) = \mathbb{E}_{\delta\sim q_0}[(D(x+\delta)-x) + \lambda L(h(D(x+\delta)), h(x))]$$

This denoised smoothing manages to reach similar certified radius to [Cohen et al., 2019] without any pre-training, which is an encouraging lead for plug-and-use randomized smoothing in the future. A lot of work is however required to achieve decent certified accuracy.

4.2.2 The geometry of the Neyman-Pearson set

One of the first difficulties of the framework developed by [Cohen et al., 2019] is that it is very specific to the Gaussian distribution. For other distributions, the Neyman-Pearson set has a more complex geometry, and the certificates often do not have a close form. This considerably limits the possibilities of experimentations, and may even lead to disaster if the Gaussian noise ends up not being viable (which seems to be the case, see Chapter 5 Section 5.2.3).

[Yang et al., 2020] have been the first to study the existence of an *optimal smoothing distribution* : for a given adversary constraint (for example the set $\mathcal{B}$ of possible attacks, like the ℓ_1, ℓ_2 or ℓ_∞ balls), considering that we use uniform distributions, which "shape"

should the support S of the distribution have for optimal certificates ? It turns out that an answer exists : the best distribution uses a support that is in the shape of the *Wulff Crystal* of the constraint set $\mathcal{B}$.

Definition 47 (Wulff Crystal)**.** *The Wulff Crystal relative to a constraint set $\mathcal{B}$ that is a ball for some norm $\|.\|$ is defined as the unit ball of the norm dual to $\|.\|$.*

For example, for the ℓ_2 adversary constraint, the optimal distribution is the uniform ℓ_2, whereas for the ℓ_1 ball it will be a cube (or ℓ_∞ ball). This result in fact generalizes for a way larger set of possible constraints, see [Yang et al., 2020].

This then raises the question : how do we compute the Neyman-Pearson set efficiently for potentially complex distributions ?

4.2.3 Computing the Neyman-Pearson set

For that, [Yang et al., 2020] have introduced two very powerful methods : first using level sets to compute the Neyman-Pearson function exactly, then a gradient-based approach to bound the robust radius. To simplify the notations, we will take the point x as the center of our coordinate system. Furhtermore, let $q_1 = q_0(\cdot + \delta)$. The Neyman-Pearson set thus becomes :

$$\mathcal{S}_k = \left\{u \in \mathbb{R}^d | q_1(u) \leq k q_1(u - \delta)\right\}$$

The level set method The main idea of this method is to divide the Neyman-Pearson set into "level sets" $\partial U_t = \left\{u \in \mathbb{R}^d | q_1(u) = t\right\}$, i.e. sets that have a constant value of q_1. They are the boundaries of the superlevel sets $U_t = \left\{u \in \mathbb{R}^d | q_1(u) \geq t\right\}$. This allows us to express $\mathcal{S}_k$ as :

$$\begin{aligned}\mathcal{S}_k &= \bigcup_{t \geq 0} \left\{ u \in \mathbb{R}^d | q_1(u) = t \text{ and } \frac{t}{k} \leq q_1(u - \delta) \right\} \\ &= \bigcup_{t \geq 0} \left\{ \partial U_t \cap (U_{t/k} - \delta) \right\}\end{aligned}$$

Finally, the measure of the Neyman-Pearson set under q_1 (which is the certificate !) can be computed as:

Theorem 12 (Level-Set method). *We have :*

$$q_1(\mathcal{S}_k) = \int_{t=0}^{+\infty} \int_{\partial U_t \cap (U_{t/k} - \delta)} \frac{t}{\|\nabla q_1(u)\|_2} du dt$$

The main advantage of this method is that we now only need to handle the geometry of the level sets and superlevel-sets of the distribution q_1, instead of the Neyman-Pearson set itself, which is usually much more simple, especially when the distribution q_1 exhibits nice properties of symmetry and invariances. The computation is usually done by Monte-Carlo sampling, with a binary seach on k as usual.

The differential method This method consists in deriving a bound on the robust radius by computing, in a sense, the derivative of the attack score : how much can an infinitesimal perturbation modify the measure under q_1 of any set of given measure p.

Theorem 13 (Differential method, [Yang et al., 2020]). *The robust radius is at least:*

$$R := \int_{1-p}^{1/2} \frac{1}{\zeta(p)} dp$$

where $\zeta(p) := \sup_{\|v\|=1} \sup_{U, q_1(U)=p} \lim_{r \to 0} \frac{q_1(U - rv) - p}{r}$.

They further introduce several techniques that allow for easy computations of the function ζ for isotropic distributions.

4.2.4 Relaxing the attack constraint

Deriving a robust radius in the general case involves evaluating, for point x, all possible cases of the smoothing measure under attack, i.e. $A_{x,\epsilon} := \{q(x + \delta), \|\delta\| \leq \epsilon\}$. The geometry of that set is difficult to grasp in high dimensions, which why the Neyman-Pearson approach is so popular, even though it gives a potentially loose bound on the certificate due to the little information it uses.

[Dvijotham et al., 2020] have introduced an alternative approach to Neyman-Pearson : they relax the set $A_{x,\epsilon}$ by replacing the pointwise constraint ($\|z - x\| \leq \epsilon$) by a global constraint on the distribution, by considering the set $B_{x,\epsilon}$ of all ν such that $d(q(x), \nu) \leq$

K for some distance d over $\mathcal{P}(\mathcal{X})$ and some constant K that depends on ϵ so that $A_{x,\epsilon} \subset B_{x,\epsilon}$.

In particular :

$$A_{x,\epsilon} \subset \left\{ \nu | KL(\nu, q(x)) \leq \frac{\epsilon^2}{2\sigma^2} \right\}$$

Where KL designs the Kullback-Leibler divergence between two probability distributions. They also show similar results for other divergences such as the Renyi and Hockey-Stick. The idea of this framework is that the divergence constraints can be easier to bound than the general attack one.

4.2.5 Current performance

To our knowledge, the current state-of-the art of robust radii was achieved by [Salman et al., 2019] with the SmoothAdv training that we described earlier. Table 4.1 summarizes its performance, and compares it to the base model of [Cohen et al., 2019].

ℓ_2 radius (ImageNet)	0.5	1	1.5	2.0	2.5	3.0	3.5
[Cohen et al., 2019]	49	37	29	19	15	12	9
[Salman et al., 2019]	56	45	38	28	26	20	17

Table 4.1: State of the art for Randomized smoothing certified radius. This shows the percentage of points exhibiting each ℓ_2 radius, on ImageNET, for the best classifiers of respectively Cohen et Al and Salman et Al.

As we can see, current certification methods are still far from providing satisfying robustness guarantees, which would be comparable to the natural accuracy of modern classifiers. The slow rate of improvement and the apparent bottleneck in the increase of certified radii lead many researchers to investigate the limitations of Randomized smoothing.

4.3 Current limitations of Randomized smoothing

Despite the many strengths of Randomized smoothing, in particular its ability to scale with network architectures easily, several limitations have been identified.

4.3.1 Confidence intervals make sampling large radius increasingly costly

The first limitation is due to the intrinsical nature of Monte-Carlo sampling and confidence interval. Namely, even if we managed to design a "perfect" randomized smoothed classifier

that is immune to all adversarial attacks, this wouldn't directly reflect on large robust radii on all points. This is because we use lower confidence bounds on the probability p : even if we sampled N values for $h(x+v)$ with $v \sim q_0$, and all values were of class 1, with confidence value $1-\alpha$ we would only get a bound $\underline{p} = \alpha^{\frac{1}{N}}$, which is very slow to converge to 1 as the number of samples increases.

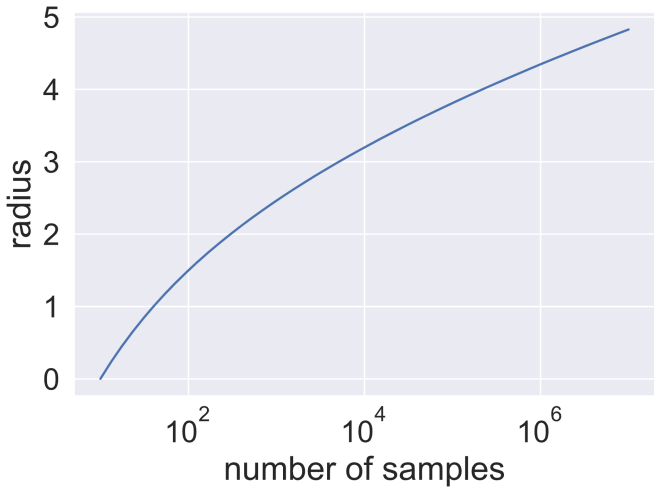


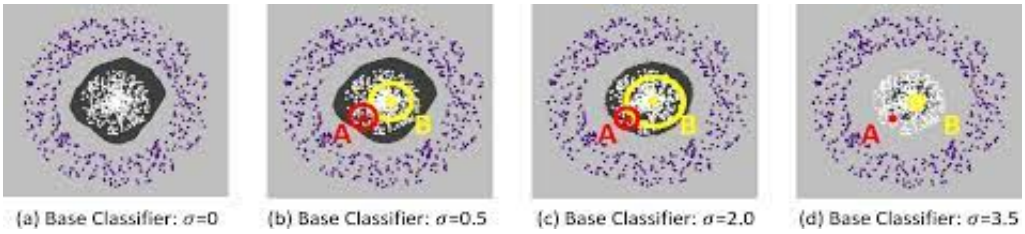
Figure 4.2: For true value p=1, the radius is a concave function of the number of samples, and so the marginal gains are decreasing. It takes more added samples to make the radius grow from 2 to 3 than from 1 to 2. Figure from [Cohen et al., 2019]

4.3.2 Randomized smoothing shrinks convex decision regions

Randomized smoothing works by locally averaging a classifier using an isotropic noise distribution. However, as it uses no information on the base classifier outside of the local probablity $p = \mathbb{E}_{u\sim q_0}[h(x+u)]$, it is blind to the way that probability is allocated in space.

In particular, as we will study in more details in Chapter 5 Section 5.2, randomized smoothing is blind to the local convexity of the decision boundary. Consider $\mathbb{R}^2$ and a classifiaction region that is a disk. When averaging, using a uniform distribution, around each point, the points that are close to the border of the disk will have a majority of points outside, due to the curvature of the disk.

Figure 4.3 shows that for convex, bounded decision regions, the shrinking can be enough to make the whole zone misclassified. This explains why robust radii do not increase for all points as the radius of the noise increases.



(a) Base Classifier: σ=0 (b) Base Classifier: σ=0.5 (c) Base Classifier: σ=2.0 (d) Base Classifier: σ=3.5

Figure 4.3: Randomized smoothing shrinks convex decision boundaries, even more when noise variances are high. The red and yellow circles represent the robust radius around two points.

[Mohapatra et al., 2021] have formalized this notion, and shown some conditions under which the shrinking occurs. They have shown in particular that the shrinking always occurs for bounded decision regions when the noise radius is large enough.

4.3.3 The robustness-accuracy tradeoff

Let h be some base classifier, and $(q_\sigma)_{\sigma>0}$ be a family of smoothing distributions, whose only difference is their variance σ (for example $q_\sigma = \mathcal{N}(0, \sigma^2 I)$. How do we choose the parameter σ ? We have two opposite goals :

- Increasing σ means averaging over a larger zone of the input space. Hence, the smoothed classifier will be robust to a wider size of perturbations;
- Increasing σ also means convoluting the conditional distributions μ_i with larger noises, and so making them harder to separate. It follows that the optimal natural accuracy will decrease, and the smoothed classifier will perform less well without attacks.

This is what we call the **robustness-accuracy** tradeoff, and it is a major limitation of randomized smoothing at the current time. We can formalize these notions as follows:

Definition 48 ($(\epsilon, s) - robustness$ [Yang et al., 2020]). *We say that q is $(\epsilon, s) - robust$ if for any $x, y \in \mathbb{R}^d$ such that $\|x - y\| \leq \epsilon$, and any set (i.e. base classifier) $U \subset \mathbb{R}^d$, we have :*

$$q(x + U) \geq \frac{1}{2} + s \Rightarrow q(y + U) \geq \frac{1}{2}$$

In other words, if any classifier returns a probability $\frac{1}{2} + s$ under q at point x, it has a robust radius of at least ϵ around point x.

Definition 49 (l-accuracy, [Yang et al., 2020]). *For all $x, y \in \mathbb{R}^d$ such that $\|x - y\| \geq 1$, there exists a set (i.e. classifier) U such that :*

$$|q(x + U) - q(y + U)| \geq l$$

In other words, the smoothing scheme does not collapse points : if x and y are away from more than 1, then there should exist a classifier that differentiates them. This is a very weak assumption, since the classifier can be different for every pair of points.

[Yang et al., 2020] have shown that maintaining an even merely decent accuracy leads to vanishing robust radii as the dimension of the problem increases :

Theorem 14 (Impossibility result, [Yang et al., 2020]). *Let $\|.\| = \|.\|_p$ for any $p \in [1, +\infty]$. There exists $c > 0$ such that for any smoothing scheme that is both $(\epsilon, s)-$robust and $l-$accurate with $\frac{s}{l} \leq c$, we have :*

$$\epsilon \leq \mathcal{O}(min(1, d^{-1/2+1/p}))$$

This means that, for any smoothing distribution that uses the Neyman-Pearson guarantee from [Cohen et al., 2019], the robust radius will vanish as the dimension increases. Equivalently, we would need absurdely large variances of noise to maintain even merely decent accuracies in very high dimensions.

4.3.4 Going beyond the Neyman-Pearson certificates

The impossibility result described in the previous section has lead most of the research community away from randomized smoothing, which appeared as a doomed method.

However, these results only apply for certificates that use no information on the classifier outside of the probability p. It follows that more classifier-specific certificates may have a chance to evade these limitations. Several papers have begun investigating that question.

Using the full probability distribution [Dvijotham et al., 2020] derive certificates in what they call the "full-information" setting, which is misleading as the information is only on the output of the smoothed classifier, and not the base one. This means having access to the probability of each class, and not only the dominant one. Furthermore, they speed up computations using the $f - divergence$ method described earlier. This only leads to marginal improvement compared to [Cohen et al., 2019].

First and second-order information [Mohapatra et al., 2020] and [Levine et al., 2021] showed that using first-order or second-order information leads to slightly better certified radius. Mohapatra et al. also shows that it is theoretically possible to reconstruct a Gaussian smoothed classifier using only information about its successive derivatives at the point of interest (even the first derivatives are, however, extremely expensive to compute).

Although the higher-order approach is very similar to the framework we develop in Chapter 5 Section 5.4, it exhibits a major limitation, namely its lack of modularity. The results only stand for Gaussian smoothing (which as we will see in Chapter 5 Section 5.2 is not suited for large dimensions), and cannot be easily combined with additional information gathering techniques. As the higher-order informations are exponentially more costly to compute, this method unfortunately has no chance of scaling in very high dimensions.

4.3.5 Specificity of our work

In the next chapter, we will analyze the limitations of randomized smoothing and offer a framework to bypass them. Here, we will discuss the novelty of our approach, and how our results differ from the state of the art.

Linking information to the limitations of RS As stated in several papers such as [Yang et al., 2020], current no-go results only apply to certificates that use information from a single noise distribution at test time. The intuition was that gathering more information *may* lead to better results, but with no proof or even intuition on why this would be the case. We are to our knowledge the first to provide a clear link between the lack of information and the bad scaling of single-noise certificates. Furthermore, we identify a cause for that poor performance in high dimension, namely that single-noise certificates are blind to the curvature of the decision boundary. We show that for toy decision boundary that exhibit a high curvature, the gap between the perfect certificate and single-noise ones can become arbitrarily large.

Multiple-noise certificates We then introduce a framework that leverages the generalized Neyman-Pearson lemma to use information from several noise distributions at the same time. The novelty of this framework is twofolds : first, we separate the information gathering from the smoothing itself, thus incurring no additional loss of accuracy. Second, this works with any combination of noise distributions, which allows for tuning and optimization that previous frameworks did not permit. Finally, we hint at how prior information on the classifier may be used to drastically reduce the number of noises required to achieve a given precision in the certificates.

High-probability certification We introduce a new type of certificates, called *high-probability certificates*, which provide a guarantee with high probability over some randomization process that is inherent to the certification. In our case, we gather information from several Gaussian noises whose centers are drawn randomly around the point of interest. This ensures that, in high dimension, any attack will be orthogonal to the vector space generated by these points with high probability. We use that to derive dimension-independent certificates, which may scale to any problem once an efficient implementation is found. To our knowledge, this is the first time that such a certification method is derived.

5 A theoretical analysis of Randomized smoothing certification

Contents

In Chapter 3, we studied conditions for the existence and computability of optimal classifiers. Furthermore, our game theory analysis gave us two key insights on the adversarial example problem:

- Randomization is the most promising direction of research, if we ever wish to devise an optimally robust algorithm, that works against any attack;
- When adding randomization to an existing classifier to make it more robust, the shape of the decision boundary plays a crucial role. Hence we should look for classifier-specific randomization when possible.

In this chapter, we will study a different problem. If we ever find a good candidate classifier, how can we certify its performance ? To tackle that question, we need a way of obtaining certificates from any base classifier, and a way of collecting information from that classifier to make the certificates specific. Randomized smoothing provides both of these advantages, but as we saw in Chapter 4, currently suffers from strong impossibility results, and is thus widely considered as a doomed method.

Our main goal will be to show that these impossibility results can be bypassed when using stronger, classifier-specific certificates, and that these can be derived with no further loss of natural accuracy. More specifically, we will tackle the following two questions :

Q1: *Are the impossibility results for Randomized smoothing intrisical, or a byproduct of the current certification methods ?*

Q2: *Is it possible to devise certificates that use more classifier-specific information, without any loss of natural accuracy ?*

To answer **Q1**, we will focus on uniform noise distributions, and show in Section 5.2 that current certificates gather suboptimal information on classifier, and are in particular blind to the local curvature of the decision boundary. We show that this underestimation

can become arbitrarily bad as dimension increases, justifying the impossibility results. We then introduce a new framework for deriving certificates in Section 5.4, and show in Section 5.5 that this framework allows to approximate the perfect certificate with arbitrary precision, without any further loss of natural accuracy, thus answering **Q2**. Finally, in Section 5.6 we will begin an analysis on the choice of noise distributions to efficiently gather information from the classifier.

5.1 A general framework to study Randomized smoothing certification

5.1.1 Probabilities and certificates

Recall that, for a binary classification problem, randomized smoothing consists in computing the probability $p(x, h, q_0) = \mathbb{P}_{u\sim q_0}[h(x+u) = 1]$, and then returning 1 iff $p(x, h, q_0) \geq \frac{1}{2}$, thus performing a majority voting over the classes.

In this chapter, we will consider the points x such that $p(x, h, q_0) > \frac{1}{2}$, so where the smoothed classifier returns 1. The other case is exactly symmetrical. We will now give a general definition of the robustness guarantees provided by randomized smoothing.

Definition 50 (ϵ-certificate). *An ϵ-certificate for the q_0-randomized smoothing of h at point x is any lower bound on the probability after attack, i.e., some value $v \in \mathbb{R}$ such that:*

$$v \leq \inf_{\delta\in B(0,\epsilon)} p(x+\delta, h, q_0)$$

A certificate v is said to be successful if $v > \frac{1}{2}$.

A successful ϵ-certificate means that no attack of norm at most ϵ can fool the classifier. This definition allows us to compare different certificates for the same smoothed classifier. In the next subsection, we will focus on a particular class of certificates.

5.1.2 Partial information certificates

Certificates for randomized smoothing are usually "black-box", *i.e.* we can only access the classifier h through limited queries. This means giving a bound on the worst-case scenario for some class of functions $\mathcal{G}$ that we know contains h.

Definition 51 (Partial-information certificate). *Let q_0 be a probability density function, and $\mathcal{G}$ a family of classifiers. The $\mathcal{G}-$partial-information ϵ-certificate for the q_0-randomized smoothing of h at point x is:*

$$\mathrm{PIC}(h, q_0, x, \epsilon, \mathcal{G}) = \inf_{g \in \mathcal{G}} \inf_{\delta \in B(0,\epsilon)} p(x + \delta, g, q_0)$$

When the infimum over $\mathcal{G}$ is attained by some g, we call g a $\mathcal{G}$-worst case classifier.

Definition 52 (Noised-based certificate). *Let $\mathcal{Q}$ be a finite family of probability density functions, q_0 a probability density function. The $\mathcal{Q}-$noise-based ϵ-certificate for the q_0-randomized smoothing of h at point x is:*

$$\mathrm{NC}(h, q_0, x, \epsilon, \mathcal{Q}) = \mathrm{PIC}(h, q_0, x, \epsilon, \mathcal{G}_{\mathcal{Q}})$$

where:

$$\mathcal{G}_{\mathcal{Q}} = \{g \in \mathcal{H} \mid \forall q \in \mathcal{Q},\ p(x, g, q) = p(x, h, q)\}$$

This is a special case of partial information certificate. We sometimes call a $\mathcal{G}_{\mathcal{Q}}$-worst case classifier a $\mathcal{Q}$-worst case classifier.

A noise-based certificate is a lower bound over all classifiers that exhibit the same response as h to every noise distribution in $\mathcal{Q}$. The certificate from [Cohen et al., 2019] is a particular type of noise-based certificate, where we only use one distribution to gather information, namely the same q_0 that is used for the smoothing, *i.e.*, $\mathcal{Q} = \{q_0\}$.

Note that there is a fundamental difference between q_0, the noise used for the smoothing, which is a part of the smoothed classifier h_{q_0} used at test time, and the family $\mathcal{Q}$, which are noises used to analyze the base classifier h, and so incur no loss of natural accuracy. Noises from $\mathcal{Q}$ are only used for information-gathering.

5.1.3 Comparing and evaluating certificates

Certificates being bounds, they can be compared as numbers. However, to evaluate the quality of a certificate, we now need a benchmark. For that, we will use the *perfect certificate*, *i.e.*, the tightest possible bound, that uses full information over the classifier h.

Definition 53 (Perfect certificate). *The perfect ϵ-certificate for the q_0-randomized smoothing of h at point x is:*

$$\mathrm{PC}(h, q_0, x, \epsilon) = \inf_{\delta \in B(0,\epsilon)} p(x + \delta, h, q_0)$$

The underestimation between prefect certificates and noise-based certificates can now be defined as the difference between both bounds.

Definition 54 (Underestimation of a noise-based certificate). *Let $\mathcal{Q}$ be a finite family of probability density functions and let $q_0 \in \mathcal{Q}$ and $\epsilon > 0$. We define the underestimation function ν as:*

$$\nu(h, q_0, x, \epsilon, \mathcal{Q}) = \mathrm{PC}(h, q_0, x, \epsilon) - \mathrm{NC}(h, q_0, x, \epsilon, \mathcal{Q})$$

The function ν computes the difference between the perfect ϵ-certificate and the noise-based ϵ-certificate for an classifier h with randomized smoothing q_0.

5.2 A theoretical analysis of the underestimation gap

In this section, we provide insight on the perceived limitations of randomized smoothing. Recall that single-noise certificates, *i.e.*, $\mathrm{NC}(h, q_0, x, \epsilon, \{q_0\})$, use the same noise q_0 for smoothing and information-gathering. This technique presents several weaknesses:

1. Since $\mathcal{Q}$ is small, the certificate is obtained as a worst-case over a large set of functions $\mathcal{G}_{\mathcal{Q}}$. This will often make it significantly poorer than the optimal certificate PC for our specific classifier.
2. The only way to gather more information on the decision boundary (and thus to obtain a better certificate) is by increasing the variance of the noise. But this automatically leads to a decrease in natural accuracy.
3. Such classifier-agnostic certificates considerably limit the possibilities of optimization through the choice of the base classifier h. In particular, single-noise certificates are blind to the "local curvature" of the decision boundary, as will be illustrated shortly.

5.2.1 Modelling the local curvature through toy decision boundaries

To illustrate these limitations, we provide a deeper analysis of the underestimation function defined in Definition 54. In the following, we focus on the case of uniform noise

distributions on an ℓ_2 ball. To illustrate the importance of the local curvature of the decision boundary, we use two families of parametric classifiers, that make this concept easy to observe, namely cones and $2-$piecewise linear sets. In both cases, the angle θ represents how "open" the set is, and low values of θ correspond to a sharp curvature of the associated decision boundary.

In the following, we will consider the dimension $d \geq 3$. Most decision boundaries we will use will have rotational symmetry, which will allow for simplified computations and parametrization. Hence, we will define them using hypercylindrical coordinates :

Definition 55 (Hyper-cylindrical coordinates). *This is an extension of the hyperspherical coordinates, defined in [Blumenson, 1960]] Let $e_1, \dots, e_d$ be an orthonormal base of $\mathbb{R}^d$, with corresponding Euclidean coordinates $(x_1, \dots, x_d)$. The hypercylindrical coordinates of axis e_1 are the following change of variable:*

$$z = x_1 \tag{5.1}$$

$$\rho = \sqrt{x_2^2 + \cdots + x_d^2} \tag{5.2}$$

$$\phi_i = \operatorname{arccot}\left(\frac{x_i}{\sqrt{x_n^2 + \cdots + x_i^2}}\right) \tag{5.3}$$

$$\phi_{d-1} = 2\operatorname{arccot}\left(\frac{x_{d-1} + \sqrt{x_{d-1}^2 + x_d^2}}{x_d}\right) \tag{5.4}$$

with the following reverse transformation:

$$x_1 = z \tag{5.5}$$

$$x_2 = r\cos(\phi_1) \tag{5.6}$$

$$x_i = r\left(\prod_{i=1}^{i-2}\sin(\phi_i)\right)\cos(\phi_{i-1}) \tag{5.7}$$

$$x_d = r\left(\prod_{i=1}^{d-2}\sin(\phi_{i-2})\right) \tag{5.8}$$

where $i \in \{2, \dots, d-1\}$. This is a bijection, where $\phi_i \in [0, \pi]$, $r \in \mathbb{R}_+$, and $\phi_{d-1} \in [0, 2\pi]$, with the convention that $\phi_k = 0$ when $x_k, \dots, x_n = 0$. Note that it is simply a change of variables to hyperspherical coordinates on the $d-1$ last variables.

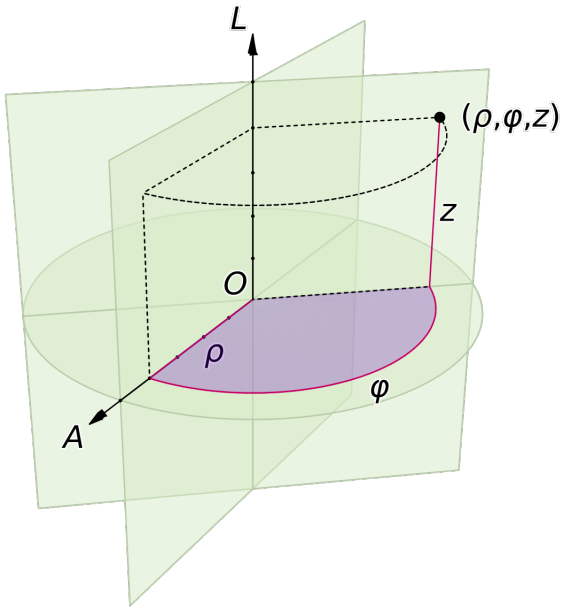


Figure 5.1: Illustration of hypercylindrical coordinates in 3 dimensions

Definition 56 (Cone of revolution). *Let $c \geq 0$. For any $x \in \mathbb{R}^d$, let $z, \rho, \phi_1, \ldots, \phi_{d-2}$ be the hyper-cylindrical coordinates of axis e_1 (see Definition 55). The cone of revolution of axis e_1, peaked at c and of angle $\theta \in \left[0, \frac{\pi}{2}\right]$ is the set $\mathcal{C}(c, \theta)$, defined by:*

$$\left\{ \begin{array}{l} z \in \mathbb{R} \\ \rho \in \mathbb{R}_+ \\ \phi_1, \ldots, \phi_{d-2} \in [0, \pi]. \end{array} \middle| \; z > c \text{ and } \rho \leq z \tan\theta \right\}$$

when $\theta \leq \frac{\pi}{2}$ (convex cone), and the set:

$$\mathcal{C}(c, \theta) = \{z \geq c \text{ or } \rho \geq -z \tan(\pi - \theta)\}.$$

for the concave cone ($\theta > \frac{\pi}{2}$).
We define a classifier with conical decision boundary as $h_\theta : x \mapsto \mathbb{1}\{x \notin \mathcal{C}(c, \theta)\}$.

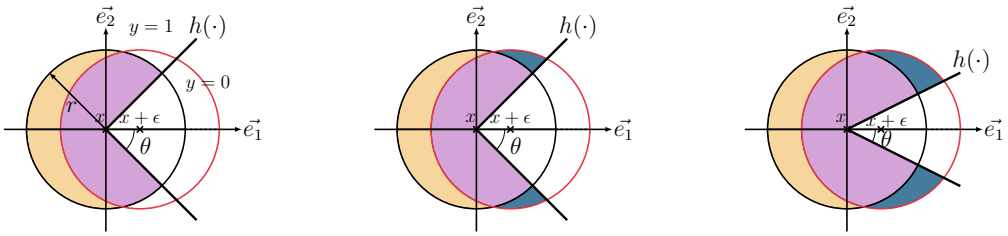


Figure 5.2: Illustration of Theorem 15. Figure (a) describes the probability $p(x, h, q_r) = q_r(\bullet) + q_r(\bullet)$. Figure (b) describes the perfect certificate as $\mathrm{PC}(h, q_r, x, \epsilon) = q_r(\bullet) + q_r(\bullet)$ whereas the single noised-based certificate is $\mathrm{NC}(h, q_r, x, \epsilon, \{q_r\}) = q_r(\bullet)$. Figure (c) shows that blue zone increases with θ, also, for $\theta = 0$, we have $q_r(\bullet) \xrightarrow[d\to\infty]{} 1$.

Definition 57 (2-piecewise Linear set). *Let $c \geq 0$. Let $x_1, \dots x_n$ be the euclidean coordinates in the base $(e_1, \dots, e_n)$. The 2-piecewise linear decision region of axis e_1 and e_2, of distance c and angle $\theta \in \left[0, \frac{\pi}{2}\right]$ is the set:*

$$\left\{ x_1, \dots, x_n \in \mathbb{R} \mid x_1 > c \text{ and } \arctan\left(\frac{x_2}{x_1}\right) \in [-\theta, \theta] \right\}$$

Remark 3. *We can generalize this definition into a $n-$piecewise linear set, whose decision boundary is locally linear except around "fracture points". This is obtained, for example, with neural networks that use ReLu activations, so their study is of particular interest for machine learning.*

Definition 58 (Linear half-space). *Let $c \geq 0$. The half-space of translation c is, in hypercylindrical coordinates of axis e_1, the set:*

$$H(c) = \left\{ \begin{array}{l} z \in \mathbb{R} \\ \rho \in \mathbb{R}_+ \\ \phi_2, \dots, \phi_{d-1} \in [0, \pi]. \end{array} \middle| \; z > c \right\}$$

5.2.2 Quantifying the underestimation of single-noise certificates

Intuition of Theorem 15. For both conical and 2-piecewise-linear decision boundaries, the gap between uniform single-noise certificates and the uniform perfect certificate increases with the local curvature of the decision boundary. For high local curvatures,

this gap becomes arbitrarily large as the dimension of the problem increases. Figure 5.2 illustrates this result in 2 dimensions.

Theorem 15 (Underestimation of single noise-based certificates). *Let $\epsilon, r \in \mathbb{R}_+^*$ such that $\epsilon \leq r$. Let $\mathcal{Q} = \{q_r\}$ where q_r is a uniform distribution over an ℓ_2 ball $B_2^d(0, r)$. We denote $\theta_m = \arccos\left(\frac{\epsilon}{2r}\right)$. For any $\theta \in [0, \theta_m]$, we denote by h_θ the classifier whose decision boundary is a cone of revolution of peak 0, axis e_1 and angle θ where $(e_1, \ldots, e_d)$ be any orthonormal basis of $\mathbb{R}^d$. Then, $\nu(h_\theta, q_r, 0, \epsilon, \mathcal{Q})$ is a continuous and decreasing function of θ. Furthermore, we have*

- $\nu(h_0, q_r, 0, \epsilon, \mathcal{Q}) = 1 - I_{1-(\frac{\epsilon}{2r})^2}\left(\frac{d+1}{2}, \frac{1}{2}\right)$
- $\nu(h_{\theta_m}, q_r, 0, \epsilon, \mathcal{Q}) = 0$

where $I_z(a, b)$ is the incomplete regularized beta function.
For any ϵ, r, $\nu(h_0, q_r, 0, \epsilon, \mathcal{Q}) \xrightarrow[d\to\infty]{} 1$. The same result holds for 2-piecewise linear sets.

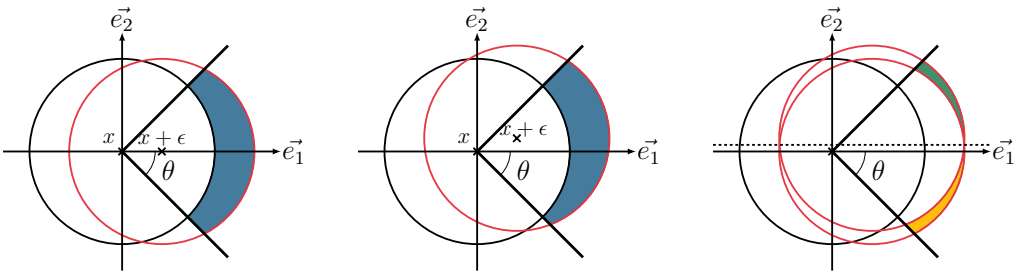


Figure 5.3: Illustration of the optimal attack with a cone of revolution as decision boundary. The optimal attack of norm ϵ is the vector $\delta = [\epsilon\vec{e_1}, 0, \ldots, 0]$. Figure (a) shows that there is always a gain by translating along e_1, Figure (b) shows the gain when translating along both e_1 and e_2, and finally, Figure (c) shows the difference. The loss incurred by the second translation, visible in the yellow zone, is greater than the gain (green zone). The argument of the proof is that the reflection of the blue zone through the dotted hyperplane is contained in the yellow zone.

Proof of Theorem 15.

First, in order to define the perfect certificate, we show that the optimal attack against a conical decision boundary is the translation along its axis. This means that the attack defined by $\delta = [\epsilon\vec{e_1}, 0, \ldots, 0]^\top$ is optimal. To prove that, we exploit the symmetry of the problem, as illustrated in Figure 5.3. To compute the difference between the perfect

certificate and single-noise certificate, we here again used the rotational symmetry of the problem around axis e_1, to compute the volume in *hyper-cylindrical coordinates* as defined Definition 55. □

Let us now prove theorem 15. We will first need two intermediary results. First of all, an asymptotic property of the regularized incomplete beta function, that is at the core of the last part of the theorem.

Lemma 7 (Limit of the regularized incomplete beta function). *Let $z \leq 1$, $b = \frac{1}{2}$ fixed. Then $I_z(a, b) \xrightarrow[a \to \infty]{} 0$.*

Proof. For any $z < 1, a > 1, b = \frac{1}{2}$, we have:

$$I_z(a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^z t^{a-1}(1-t)^{b-1}\,\mathrm{d}t \tag{5.9}$$

$$\leq \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} z^a (1-z)^{-\frac{1}{2}} \tag{5.10}$$

From [Olver et al., 2010], Equation 5.11.12, we have the following approximation:

$$\frac{\Gamma(a+b)}{\Gamma(a)} \underset{a \to +\infty}{\sim} a^b \tag{5.11}$$

from Equation (5.11), we can show that:

$$\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \underset{a \to +\infty}{\sim} \frac{a^b}{\Gamma(b)} \tag{5.12}$$

Finally, we have:

$$I_z(a,b) \leq \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} z^a (1-z)^{-\frac{1}{2}} \tag{5.13}$$

$$\underset{a \to +\infty}{\sim} \frac{a^b}{\Gamma(b)} z^a (1-z)^{-\frac{1}{2}} \tag{5.14}$$

$$\xrightarrow[a \to +\infty]{} 0 \tag{5.15}$$

which concludes the proof. □

Let us now dive in the proof itself. To compute certificates, we need to know the optimal attack against the cone. We will show that it is simply the translation along this

axis, although the proof is quite technical and relies heavily on the symmetries of the problem. In what follows, $V = \mathrm{Vol}(B_2^d(0, r))$.

Lemma 8. *The optimal attack of size $\epsilon < r$ against $\mathcal{C}(0, \theta)$ is the translation fully along its axis e_1,* i.e. ϵe_1.

Proof. Let $A = \mathrm{Span}(e_1), B = \mathrm{Span}(e_2, \dots, e_d)$. For any vector $u \in \mathbb{R}^d$, we write $u = u_A + u_B$ where u_A and u_B are the orthogonal projections of u on A and B respectively.

First we will show that since the cone is invariant by rotation around e_1, the orthogonal component of the attack is as well. Without any attack, the probability that the smoothed classifier returns 1 at point 0, is

$$p_1 = \int_{\mathbb{R}^d} \mathbb{1}\{x \in \mathcal{C}(0, \theta)\} q_0(x) dx \tag{5.16}$$

$$= \frac{1}{V} \int_{\mathbb{R}^d} \mathbb{1}\{\|x_B\| \leq \|x_A\| \tan(\theta)\} \mathbb{1}\left\{\|x_A - 0\|^2 + \|x_B - 0\|^2 \leq r^2\right\} \mathrm{d}x \tag{5.17}$$

$$\tag{5.18}$$

where V is the volume of the ball of radius r and center 0.

Let δ be any attack vector of norm $\|\delta\| = \epsilon$. Attacking the smoothed classifier by δ means that the classifier is now smoothed using the distribution $q_0(\cdot - \delta)$. This amounts to shifting the center of the ball from $(0, 0)$ to (δ_A, δ_B). Let

$$f(x, \delta) = \mathbb{1}\left\{\|x_A - \delta_A\|^2 + \|x_B - \delta_B\|^2 \leq r^2\right\} \mathbb{1}\{\|x_B\| \leq \|x_A\| \tan(\theta)\}, \qquad \forall x \in \mathbb{R}^d \tag{5.19}$$

Then at point 0, under attack δ, the smoothed classifier returns 1 with probability $p(\delta) = \frac{1}{V} \int_{\mathbb{R}^d} f(x, \delta)\, \mathrm{d}x$ where V is independent of δ.

We will now show that the attack is invariant by rotation around e_1, which means that any attack that is the image of δ by an isometry preserving e_1 gives the same probability as δ. let g be any isometric mapping such that $g|_A = \mathrm{Id}_A$. Let $\tilde{\delta} = g(\delta)$. Recall that as g is an isometry, g and g^{-1} are also affine, hence we have :

$$\begin{cases} \forall x, y \in \mathbb{R}^d, g(x - y) = g(x) - g(y) & \text{since } g \text{ is affine} \\ \forall x, y \in \mathbb{R}^d, \|g(x)\| = \|x\| & \text{since } g \text{ is an isometry} \end{cases} \tag{5.20}$$

And the same is true for g^{-1}

We can use that to show the rotation invariance, namely :

$$
\begin{aligned}
f(g^{-1}(x), \delta) &= \mathbb{1}\left\{\|g^{-1}(x_A) - \delta_A\|^2 + \|g^{-1}(x_B) - \delta_B\|^2 \leq r^2\right\} \\
&\quad \mathbb{1}\left\{\|g^{-1}(x_B)\| \leq \|g^{-1}(x_A)\| \tan(\theta)\right\} && (5.21) \\
&= \mathbb{1}\left\{\|g^{-1}(x_A - \tilde{\delta}_A)\|^2 + \|g^{-1}(x_B - \tilde{\delta}_B)\|^2 \leq r^2\right\} \\
&\quad \mathbb{1}\left\{\|g^{-1}(x_B)\| \leq \|g^{-1}(x_A)\| \tan(\theta)\right\} && (5.22) \\
&= \mathbb{1}\left\{\|x_A - \tilde{\delta}_A\|^2 + \|x_B - \tilde{\delta}_B\|^2 \leq r^2\right\}\mathbb{1}\{\|x_B\| \leq \|x_A\| \tan(\theta)\} && (5.23) \\
&= f(x, \tilde{\delta}) = f(x, g^{-1}(\delta)) && (5.24)
\end{aligned}
$$

since g^{-1} is an isometry. It follows, by integrating and with a change of variable in the integral, that the probability is invariant by rotation :

$$
\begin{aligned}
p(\delta) &= \frac{1}{V}\int_{\mathbb{R}^d} f(x, \delta)\,\mathrm{d}x && (5.25) \\
&= \frac{1}{V}\int_{\mathbb{R}^d} f(g^{-1}(u), \delta)\,\mathrm{d}u && (5.26) \\
&= \frac{1}{V}\int_{\mathbb{R}^d} f(u, g^{-1}(\delta))\,\mathrm{d}u && (5.27) \\
&= p(\tilde{\delta}) && (5.28)
\end{aligned}
$$

We show that the second component is detrimental

In particular, we can always choose g such that the $g(\delta_B) = \delta_2 e_2$, isolating the orthogonal component into a single coordinate. In what follows, we will consider δ of the form $\delta = \delta_1 e_1 + \delta_2 e_2$ and show that the attack is optimal when $\delta_2 = 0$. For this, we will compute the difference between the attack translated by $\delta_1 e_1$ and the one translated by $\delta_1 e_1 + \delta_2 e_2$, to show that the orthogonal component actually reduces the efficiency of the attack. Let us first recall that

$$
\begin{aligned}
p(\delta) &= \frac{1}{V}\,\mathrm{Vol}(B(\delta, r) \cap \mathcal{C}(0, \theta)). && (5.29) \\
&= \frac{1}{V}\int_{\mathbb{R}^d} \mathbb{1}\left\{\|x_1 - \delta_1\|^2 + \|x_2 - \delta_2\|^2 + \|x_3\|^2 + \cdots + \|x_d\|^2 \leq r^2\right\} \\
&\quad \mathbb{1}\{\|x_B\| \leq \|x_A\| \tan(\theta)\}\,\mathrm{d}x && (5.30)
\end{aligned}
$$

(5.31)

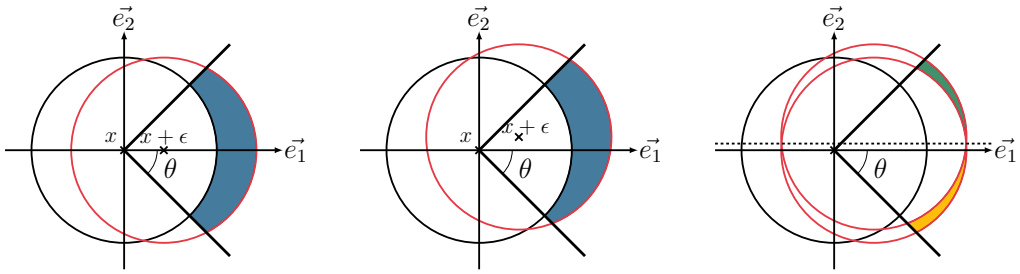


Figure 5.4: Illustration of the proof. The illustration on the left shows that there is always a gain by translating along e_1, the illustration in the middle shows the gain when translating along both e_1 and e_2, and finally, the illustration on the right shows the difference. The loss incurred by the second translation, visible in the yellow zone, is greater than the gain (green zone). The argument of the proof is that the symmetric of the blue zone is contained in the yellow zone.

Let $A = B(\delta, r) \cap \mathcal{C}(0, \theta) \setminus B(\delta_1 e_1, r)$ and $D = B(\delta_1 e_1, r) \cap \mathcal{C}(0, \theta) \setminus B(\delta, r)$. These represents respectively the points lost and gained when attacking along the coordinate e_2 after having already attacked along e_1, as we will show :

$$
\begin{aligned}
p(\delta_1 e_1) - p(\delta) &= \frac{1}{V}\operatorname{Vol}(B(\delta_1 e_1, r) \cap \mathcal{C}(0, \theta)) - \frac{1}{V}\operatorname{Vol}(B(\delta, r) \cap \mathcal{C}(0, \theta)) && (5.32)\\
&= \frac{1}{V}(\operatorname{Vol}(B(\delta, r) \cap \mathcal{C}(0, \theta) \cap B(\delta_1 e_1, r)) \\
&+ \operatorname{Vol}(B(\delta, r) \cap \mathcal{C}(0, \theta) \setminus B(\delta_1 e_1, r)) \\
&- \operatorname{Vol}(B(\delta_1 e_1, r) \cap \mathcal{C}(0, \theta)) \cap B(\delta, r)) \\
&- \operatorname{Vol}(B(\delta_1 e_1, r) \cap \mathcal{C}(0, \theta) \setminus B(\delta, r)) && (5.33)\\
&= \frac{1}{V}(\operatorname{Vol}(D) - \operatorname{Vol}(A)) && (5.34)
\end{aligned}
$$

We have $p(\delta_1 e_1) - p(\delta) = \frac{1}{V}(\operatorname{Vol}(D) - \operatorname{Vol}(A))$. To show that it is positive, i.e. that there are more points lost than gained with the translation along e_2 we will show that there is an isometry v (preserving volumes) such that $v(A) \subset D$.

Let v be the reflection across the hyperplan $\left\{x \in \mathbb{R}^d, x_2 = \frac{\delta_2}{2}\right\}$. We have $v(x_1, \ldots, x_d) = (x_1, \delta_2 - x_2, x_3, \ldots, x_d)$. For simplicity, for any $x \in \mathbb{R}^d$, we denote $v(x) = \tilde{x}$.

Let $x \in A$. We will show that $\tilde{x}$ is in D. As x is in A, we have

$$\begin{cases} (x_1 - \delta_1)^2 + (x_2 - \delta_2)^2 + x_3^2 + \cdots + x_d^2 \leq r^2 & (5.35) \\ x_1 > 0 & (5.36) \\ x_2^2 + \cdots + x_d^2 \leq x_1^2 \tan^2(\theta) & (5.37) \\ (x_1 - \delta_1)^2 + x_2^2 + x_3^2 + \cdots + x_d^2 > r^2 & (5.38) \end{cases}$$

Equation (5.35) states that $x \in B(\delta, r)$, Equation (5.36) and Equation (5.37) state that it is in the cone, whereas Equation (5.38) says that $x \notin B(\delta_1 e_1, r)$.

Let us first show that $\tilde{x} \in \mathcal{C}(0, \theta)$. $\tilde{x_1} = x_1 > 0$, and subtracting Equation (5.35) from Equation (5.38) gives us $x_2^2 > (x_2 - \delta_2)^2$. It follows:

$$\begin{aligned} \tilde{x_2}^2 + \cdots + \tilde{x_d}^2 &= (\delta_2 - x_2)^2 + x_3^2 + \cdots + x_d^2 && (5.39) \\ &< x_2^2 + \cdots + x_d^2 && (5.40) \\ &\leq x_1^2 \tan^2(\theta) \quad \text{(from Equation (5.37))} && (5.41) \\ &= \tilde{x_1}^2 \tan^2(\theta) && (5.42) \end{aligned}$$

Now we show that $\tilde{x} \in B(\delta_1 e_1, r)$.

$$\begin{aligned} (\tilde{x_1} - \delta_1)^2 + \tilde{x_2}^2 + \cdots + \tilde{x_d}^2 &= (x_1 - \delta_1)^2 + (x_2 - \delta_2)^2 + x_3^2 + \cdots + x_d^2 && (5.43) \\ &\leq r^2 && (5.44) \end{aligned}$$

Finally we show $\tilde{x} \notin B(\delta, r)$.

$$\begin{aligned} (\tilde{x_1} - \delta_1)^2 + (\tilde{x_2} - \delta_2)^2 + \cdots + \tilde{x_d}^2 &= (x_1 - \delta_1)^2 + x_2^2 + x_3^2 + \cdots + x_d^2 && (5.45) \\ &> r^2 \quad \text{(from Equation (5.38))} && (5.46) \end{aligned}$$

Combining the above, we get $\tilde{x} \in D$. As x was chosen arbitrarily in A, we get $v(A) \subset D$. As v is isometric, we finally get $p(\delta_1 e_1) - p(\delta) = \frac{1}{V}(\text{Vol}(A) - \text{Vol}(D)) = \frac{1}{V}(\text{Vol}(v(A)) - \text{Vol}(D)) \leq 0$. The component orthogonal to the axis is detrimental to the attack.

We show that the success of the attack grows with the first coordinate

We now only need to prove that $p(\delta)$ is strictly increasing with δ_1. For what follow, we consider an attack $\delta_1 e_1$, and another one $((\delta_1 + \Delta)e_1)$. We will use the same technique:

Let $A = B(\delta_1 e_1, r) \cap \mathcal{C}(0, \theta) \setminus B((\delta_1 + \Delta)e_1, r)$, and $D = B((\delta_1 + \Delta)e_1, r) \cap \mathcal{C}(0, \theta) \setminus B(\delta_1 e_1, r)$. We have $p((\delta_1 + \Delta)e_1) - p(\delta_1 e_1) = \frac{1}{V}(\text{Vol}(D) - \text{Vol}(A))$, and we will show that there is an isometry v such that $v(A) \subset D$.

Let v be the reflection across the hyperplane $\left\{x \in \mathbb{R}^d \mid x_1 = \delta_1 + \frac{\Delta}{2}\right\}$. Let $x = (x_1, \dots, x_d) \in A$. It verifies the following equations:

$$\begin{cases} (x_1 - \delta_1)^2 + x_2^2 + x_3^2 + \cdots + x_d^2 \leq r^2 & (5.47) \\ x_1 > 0 & (5.48) \\ x_2^2 + \cdots + x_d^2 \leq x_1^2 \tan^2(\theta) & (5.49) \\ (x_1 - \delta_1 - \Delta)^2 + x_2^2 + x_3^2 + \cdots + x_d^2 > r^2 & (5.50) \end{cases}$$

$v(x_1, \dots, x_d) = (2\delta_1 + \Delta - x_1, x_2, \dots, x_d) = \tilde{x}$.

First of all, subtracting Equation (5.50) from Equation (5.47) gives:

$$\begin{aligned} (x_1 - \delta_1 - \Delta)^2 > (x_1 - \delta_1)^2 \Rightarrow \Delta^2 - 2\Delta(x_1 - \delta_1) > 0 & \quad (5.51) \\ \Rightarrow x_1 < \delta_1 + \frac{\Delta}{2} & \quad (5.52) \end{aligned}$$

Let us show $\tilde{x} \in D$.

$$\begin{aligned} \tilde{x_1}^2 \tan^2(\theta) &= (2\delta_1 + \Delta - x_1)^2 \tan^2(\theta) & (5.53) \\ &\geq \left(2\delta_1 + \Delta - \delta_1 - \frac{\Delta}{2}\right)^2 \tan^2(\theta) & (5.54) \\ &= \left(\delta_1 + \frac{\Delta}{2}\right)^2 \tan^2(\theta) & (5.55) \\ &\geq x_1^2 \tan^2(\theta) & (5.56) \\ &\geq x_2^2 + \cdots + x_d^2 & (5.57) \\ &= \tilde{x_2}^2 + \cdots + \tilde{x_d}^2 & (5.58) \end{aligned}$$

Hence $\tilde{x} \in \mathcal{C}(0, \theta)$. Then:

$$\begin{aligned} (\tilde{x_1} - \delta_1 - \Delta)^2 + \tilde{x_2}^2 + \cdots + \tilde{x_d}^2 = (x_1 - \delta_1)^2 + x_2^2 + \cdots + x_d^2 & \quad (5.59) \\ \leq r^2 & \quad (5.60) \end{aligned}$$

Hence $\tilde{x} \in B((\delta_1 + \Delta)e_1, r)$. Finally,

$$\begin{aligned} (\tilde{x_1} - \delta_1)^2 + \tilde{x_2}^2 + \cdots + \tilde{x_d}^2 = (x_1 - \delta_1 - \Delta)^2 + x_2^2 + \cdots + x_d^2 & \quad (5.61) \\ > r^2 & \quad (5.62) \end{aligned}$$

and we have $\tilde{x} \notin B(\delta_1 e, r)$. We have thus shown that $\mathrm{Vol}(D) \geq \mathrm{Vol}(A)$, and so the attack is increasing in δ_1.

Increasing along the first coordinate cannot diminish the attack

We will now show that the increase is strict. For that, we show that there exists points in D whose image by v is not in A.

Recall that $D = B(\delta_1 + \Delta, r) \cap \mathcal{C}(0, \theta) \setminus B(\delta_1, r)$ is defined by the following set of equations:

$$\begin{cases} (x_1 - \delta_1 - \Delta)^2 + x_2^2 + x_3^2 + \cdots + x_d^2 \leq r^2 & (5.63) \\ x_1 > 0 & (5.64) \\ x_2^2 + \cdots + x_d^2 \leq x_1^2 \tan^2(\theta) & (5.65) \\ (x_1 - \delta_1)^2 + x_2^2 + x_3^2 + \cdots + x_d^2 > r^2 & (5.66) \end{cases}$$

Let us reason by contradiction, and assume that $v(D) \subset A$

This means that for all points verifying the previous set of equations, we also have $x_2^2 + \cdots + x_d^2 \leq \tilde{x}_1^2 \tan^2(\theta)$, i.e.

$$x_2^2 + \cdots + x_d^2 \leq (2\delta_1 + \Delta - x_1)^2 \tan^2(\theta) \tag{5.67}$$

Let us define $u = x_1 - \delta$ and $b = \delta + \Delta$. Combining eq. (5.67) and eq. (5.66) gives:

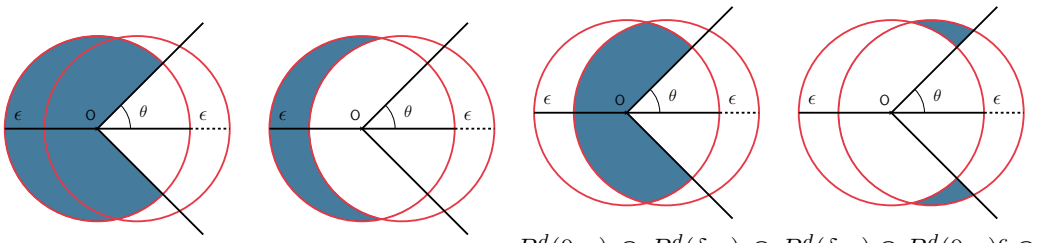
$$\begin{aligned} r^2 &\leq (x_1 - \delta)^2 + (2\delta + \Delta - x_1)^2 \tan^2(\theta) \\ &\leq u^2 + (b - u)^2 \tan^2(\theta) \\ &\leq u^2 + (b^2 + u^2 - 2bu) \tan^2(\theta) \end{aligned}$$

This implies:

$$\begin{aligned} &(1 + \tan^2(\theta))u^2 - 2b \tan^2(\theta)u + b^2 - r^2 > 0 \\ &\Rightarrow b^2 - r^2 > b^2 \frac{tan^2(\theta)}{1 + \tan^2(\theta)} \\ &\Rightarrow r^2 < b^2(1 - \sin^2(\theta) \tan^2(\theta)) \\ &\Rightarrow r^2 < \delta^2(1 - \tan^2(\theta)) \\ &\Rightarrow r^2 < \delta^2 \end{aligned}$$

Which is a contradiction since we consider attacks of size $\delta = \epsilon < r$.

We have shown that any component of the attack that is orthogonal to e_1 is detrimental to the attack, and that an increase along e_1 benefits the attack. It follows that the optimal attack of size at most ϵ is ϵe_1. □



$B_2^d(0,r) \cap \mathcal{C}(0,\theta)^c$ $B_2^d(0,r) \setminus B_2^d(\delta,r)$ $B_2^d(0,r) \cap B_2^d(\delta,r) \cap \mathcal{C}(0,\theta)^c$ $B_2^d(\delta,r) \cap B_2^d(0,r)^c \cap \mathcal{C}(0,\theta)^c$

Figure 5.5: Illustration of proof of Theorem 15. The worst-case classifier using only the information $p(\theta)$ assumes that the zone in the second figureis entirely lost, whereas for the perfect certificate, the blue zone in the fourth figure is not lost. That zone grows as θ shrinks.

We can now proove the theorem itself :

Proof of Theorem 15. Let $r > 0, 0 < \epsilon \leq r, \delta = [\epsilon, 0, \ldots, 0]^\top \in \mathbb{R}^d, \theta \in [0, \theta_m]$ with $\theta_m = \arccos\left(\frac{\epsilon}{2r}\right)$. Let $\mathcal{C}(0,\theta)$ be a cone of revolution of peak 0, axis e_1 and angle θ. Let us consider all functions h_θ whose decision boundary is the cone of revolution $\mathcal{C}(0,\theta)$. The probability $p(x, h_\theta, q_0)$ of returning class 1 at the point 0 for the classifier h_θ after smoothing by q_0 is:

$$p(x, h_\theta, q_0) = \int_{\mathbb{R}^d} \frac{\mathbb{1}\{x \in B_2^d(0,r)\}}{\mathrm{Vol}(B_2^d(0,r))} \mathbb{1}\{x \in \mathcal{C}(0,\theta)^c\} \, \mathrm{d}x \tag{5.68}$$

$$= \frac{\mathrm{Vol}(B_2^d(0,r) \cap \mathcal{C}(0,\theta)^c)}{\mathrm{Vol}(B_2^d(0,r))} \tag{5.69}$$

Single-noise certificates uses the fact that in the worst-case scenario, all the volume lost during the translation was in class 1, and all the volume gained is in the class 0 (see proof of [Yang et al., 2020] [Theorem I.19]) This gives:

$$\mathcal{C}(h_\theta, q_0, x, \epsilon, \{q_0\}) = p(x, h_\theta, q_0) - \frac{1}{V}\left(\mathrm{Vol}(B_2^d(0,r) \setminus B_2^d(\delta,r))\right) \tag{5.70}$$

$$= \frac{1}{V}\left(\mathrm{Vol}(B_2^d(0,r) \cap \mathcal{C}(0,\theta)^c) - \left(\mathrm{Vol}(B_2^d(0,r) \setminus B_2^d(\delta,r))\right)\right) \tag{5.71}$$

$$= \frac{1}{V} \mathrm{Vol}(B_2^d(0,r) \cap B_2^d(\delta,r) \cap \mathcal{C}(0,\theta)^c) \tag{5.72}$$

In Equation (5.70), we say that in the worst case scenario all the volume lost during the translation was in class 1, and all volume gained was in class 0, hence we loose everything outside the intersection.

This corresponds to Section 5.2.2: the zone that is preserved after translation for the noise certificate is the blue one in the third figure. We will show that the perfect certificate also preserves the blue zone in the fourth figure.

Since by Lemma 8 the optimal attack against the cone is the translation along its axis, the perfect certificate for the probability p will be defined under the attack δ:

$$\P\mathcal{C}(h_\theta, q_0, x, \epsilon) = \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c) \tag{5.73}$$

The difference between the perfect certificate and the single-noise based certificate (as in Definition 54) is:

$$\nu(h_\theta, q_0, x, \epsilon, \{q_0\}) = \frac{1}{V}\big(\operatorname{Vol}(B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c) - \operatorname{Vol}(B_2^d(0, r) \cap B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c)\big)$$

(5.74)

$$= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c \setminus \big(B_2^d(0, r) \cap B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c\big)) \tag{5.75}$$

$$= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c \cap \big(B_2^d(0, r) \cap B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c\big)^c) \tag{5.76}$$

$$= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \cap \mathcal{C}(0,\theta)^c \cap \big(B_2^d(0, r)^c \cup B_2^d(\delta, r)^c \cup \mathcal{C}(0,\theta)\big)) \tag{5.77}$$

$$= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \cap B_2^d(0, r)^c \cap \mathcal{C}(0,\theta)^c) \tag{5.78}$$

$$= \frac{A}{V}\int_{x=0}^{\infty}\int_{\substack{\rho= \\ x\tan(\theta)}}^{\infty} \mathbb{1}\{(x-\epsilon)^2 + \rho^2 \leq r^2\}\mathbb{1}\{x^2+\rho^2 > r^2\}\rho^{d-2}\,\mathrm{d}x\,\mathrm{d}\rho \tag{5.79}$$

where $A = \int_{\substack{\phi_1,\dots,\phi_{d-3}\in[-\pi,\pi] \\ \phi_{d-2}\in[0,2\pi]}} \prod_{k=1}^{d-2}\sin^k\phi_{d-1-k}\,\mathrm{d}\phi_1\dots\mathrm{d}\phi_{d-2}$.

It follows that ν is a continuous function with respect to $\theta \in [0, \theta_m]$. It is decreasing, since $\mathcal{C}(0,\theta_1) \subset \mathcal{C}(0,\theta_2)$ when $\theta_1 < \theta_2$.

Furthermore, when $\theta = 0$:

$$\begin{aligned}
\nu(h_0, q_0, x, \epsilon, \{q_0\}) &= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \cap B_2^d(0, r)^c) && (5.80)\\
&= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r) \setminus B_2^d(0, r)) && (5.81)\\
&= \frac{1}{V}\operatorname{Vol}(B_2^d(\delta, r)) - \operatorname{Vol}(B_2^d(0, r) \cap B_2^d(\delta, r)) && (5.82)\\
&= \frac{1}{V}\Big(\operatorname{Vol}(B_2^d(\delta, r) - 2\operatorname{Vol}(\operatorname{Cap}(r - \frac{\epsilon}{2}, r, d))\Big) && (5.83)\\
&= 1 - I_{1-\left(\frac{\epsilon}{2r}\right)^2}\left(\frac{d+1}{2}, \frac{1}{2}\right) && (5.84)
\end{aligned}$$

where the step from Equation (5.82) to Equation (5.83) is due because the intersection of both spheres is the union of two spherical caps.

Moreover, from Lemma 7, we have $\nu(h_0, q_0, x, \epsilon, \{q_0\}) \xrightarrow[d\to\infty]{} 1$:

And, when $\theta = \theta_m$, we are going to prove that $\nu(h_{\theta_m}, q_0, x, \epsilon, \{q_0\}) = 0$. Equivalently, we want to show that the set defined by:

$$\left\{(x, \rho) \in \mathbb{R} \;\middle|\; x < \frac{\epsilon}{2} \text{ and } (x-\epsilon)^2 + \rho^2 \leq r^2 \text{ and } \rho > x\tan\theta_m\right\} \tag{5.85}$$

is an empty set. Let (x, ρ) in this set. We have:

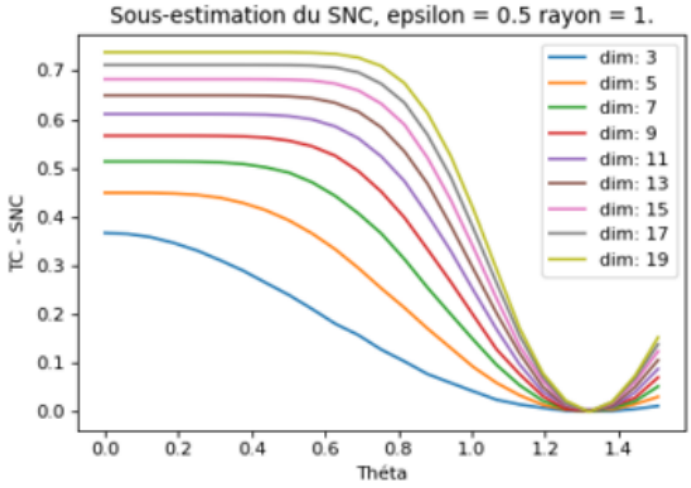
$$\begin{aligned}
\rho &> x\tan\left(\arccos\left(\frac{\epsilon}{2r}\right)\right) && (5.86)\\
&= \frac{2rx}{\epsilon}\sqrt{1 - \frac{\epsilon^2}{4r^2}} && (5.87)
\end{aligned}$$

due to the equality: $\tan(\arccos(x)) = \frac{\sqrt{1-x^2}}{x}$. Then, we have:

$$\begin{aligned}
r^2 &\geq (x-\epsilon)^2 + \rho^2 && (5.88)\\
&\geq x^2 - 2x\epsilon + \epsilon^2 + \frac{4r^2x^2}{\epsilon^2}\left(1 - \frac{\epsilon^2}{4r^2}\right) && (5.89)\\
&= x^2 - 2x\epsilon + \epsilon^2 + \frac{4r^2x^2}{\epsilon^2} - x^2 && (5.90)\\
&= \frac{4r^2}{\epsilon^2}x^2 - 2x\epsilon + \epsilon^2 && (5.91)
\end{aligned}$$

Hence we have:

$$\frac{4r^2}{\epsilon^2}x^2 - 2x\epsilon + \epsilon^2 - r^2 \leq 0 \quad \text{and} \quad x \leq \frac{\epsilon}{2} \tag{5.92}$$



h

Figure 5.6: Difference between the perfect certificate and the single-noise certificate, for several dimensions, depending on the angle θ of the cone. As we can see, the difference is high for a higher range of thetas as dimension increases.

But the minimum of the right hand side is: $\frac{\epsilon^3}{4r^2} \leq \frac{\epsilon}{2}$ because $r \leq \epsilon$. Therefore, the r.h.s is increasing on the interval $[\frac{\epsilon}{2}, \infty]$ and is equal to 0 when $x = \frac{\epsilon}{2}$, which proves that no point verifies Equation (5.92). That allows us to conclude that: $\nu(h_{\theta_m}, q_0, x, \epsilon, \{q_0\}) = 0$.

□

5.2.3 Numerical evaluation of the underestimation

As the uniform distribution has a finite support, the corresponding single-noise certificates will be blind to everything outside that support. It will thus always consider the worst-case scenario, namely that every point outside the ball is of the opposite class. Since it has no information on the precise repartition of points in the ball, the single-noise certificate will also assume that every point "lost" after the translation (see the green left crescent zone outside the intersection in Section 5.2.2) was of class 1.

Hence, the difference between the perfect certificate and the single-noise certificate is the relative area of the blue zone. The last part of the result shows that as the dimension gets higher, the single noise certificate can become arbitrarily bad. In the extreme case ($\nu(\theta) = 1$), it returns 0 even though the classification task is trivial. This is due to the fact that the volume of balls tends to concentrate on their surface in high dimensions, and so the relative weight of the crescent zone increases. We used simulations to plot the evolution of the underestimation with the parameter θ and the dimension. We used Monte Carlo sampling with $200k$ samples for each θ and each dimension.

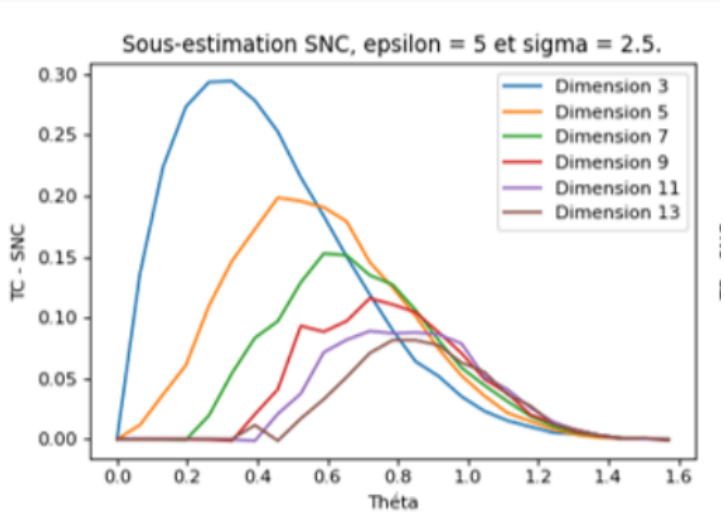


Figure 5.7: Underestimation of single-noise certificates for a normal distribution and a conical decision boundary.

As we can see in Figure 5.6, as dimension increases the cone represents a smaller portion of the space for a given θ. It follows that the underestimation becomes large for almost every θ in high dimensions.

A natural follow-up question would be whether this result stays true for other noise distributions, such as the Gaussian one, that is the most widely used in practice. However, it seems that this is not the case : as we see on Figure 5.7

Underestimation for Gaussian noise As we can see on Figure 5.7, the situation is completely different for Gaussian noise. The underestimation has a bell curve shape, getting lower and flatter as dimension increases, and seemingly converging to 0. This suggests that Gaussian noise is, in a sense, "doomed" for Randomized smoothing certification, as there is no gap to be exploited between single-noise certificates (which suffer from the no-go results) and the perfect one in high dimension.

Focusing on uniform noise, we now ask how likely we are to encounter these kinds of high local curvature situations when models are trained on real datasets?

5.3 Empirical analysis with real-world decision boundaries

In the following, we show that we can identify the points where the single-noise classifier is suboptimal, by leveraging the information from several concentric noises. More precisely,

for the uniform distribution, at points where the single-noise certificate is optimal, the probability of being in class 1 will decrease as the radius of the noise increases.

To illustrate these limitations, we provide a deeper analysis of the underestimation function defined in Definition 54. In the following, we focus on the case of uniform noise distributions on an ℓ_2 ball.

5.3.1 Identifying points of underestimation in a dataset

The main difficulty of real-world models is their opacity. We cannot know the precise shape of the decision boundary, which makes it difficult to evaluate, for example, its local curvature. We bypass this issue by providing a way to identify points where single-noise certificates cannot be optimal. The intuition is the following : we know (Theorem 16) the shape of single-noise worst-case classifiers how they would react to successive injections of noise. If the true classifier exhibits a very different reaction, then it cannot have a locally similar shape.

Intuition of Theorem 16. For an unknown decision boundary, at any point where the single-noise certificate is optimal, the probability must locally decrease with the radius of the noise. It follows that at any point where the probability increases with the radius of the noise, the single-noise certificate cannot be optimal.

Theorem 16 (Identifying points of non-zero underestimation)**.** *For any $r > 0$, let q_r denote the uniform distribution over $B_2^d(0, r)$. Let $r_1 > 0$, and h a classifier. For any $x \in \mathbb{R}^d$, $\epsilon > 0$, if $p(x, h, q_r)$ is not a decreasing function of r over $[r_1, r_1 + \epsilon)$, then* $\mathrm{PC}(h, q_{r_1}, x, \epsilon) > \mathrm{NC}(h, q_{r_1}, x, \epsilon, \{q_r\})$.

Proof. Let us denote q_r, the uniform distribution of radius $r > 0$, let $r_1 > 0$ and let g_{r_1} be the $\{q_{r_1}\}$-worst classifier. As we saw in the proof of Theorem 15, for any δ of norm ϵ, the decision region of g_{r_1}, $D_{r_1} = \{z \in \mathcal{X} \mid g_{r_1}(z) = 1\}$, is entirely contained in $B_2^d(x, r_1)$. Hence for any $r > r_1$,

$$p(x, g_{r_1}, q_r) = \mathbb{P}[U(x, r) \in D_{r_1}] \tag{5.93}$$

$$= \frac{\mathrm{Vol}(B_2^d(0, r) \cap D_{r_1})}{\mathrm{Vol}(B_2^d(x, r))} \tag{5.94}$$

$$= \frac{\mathrm{Vol}(B_2^d(x, r_1) \cap D_{r_1})}{\mathrm{Vol}(B_2^d(x, r))} \tag{5.95}$$

because $B_2^d(x, r) \cap D_{r_1} \subseteq B_2^d(x, r_1)$. Since r_1 is constant, this is a decreasing function in r.

A similar proof works for 2-piecewise linear decision boundaries. □

5.3.2 Evaluating the suboptimality of certificates on state-of-the art models

We leverage Theorem 16 to evaluate the number of points where single-noise certificates might be optimal, by "probing" the decision boundary with different distributions. Let q_0 and q_1 be two uniform distributions with r_0 and r_1 as their respective radius such that $r_0 < r_1$ and let $\mathcal{D}$ the set containing all the images of the CIFAR10 dataset [Krizhevsky et al., 2009]. We aim at computing the proportion ζ of points where the probability increases with the radius of the noise:

Table 5.1

r_0 / r_1	$\zeta(r_0, r_1)$
0.25 / 0.30	40.9%
0.50 / 0.55	41.9%
0.75 / 0.80	41.3%

$$\zeta(r_0, r_1) = \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \mathbb{1}\{p(x, h, q_0) < p(x, h, q_1)\}$$

This quantity measures the proportion of points where a better information-gathering scheme could improve the certificate. In the context of randomized smoothing, it is common practice to also add noise during training in order to avoid distribution shift at test time. Therefore, to perform this experiment, we use three models trained by [Yang et al., 2020] with uniform distribution with respective radius of 0.25, 0.50 and 0.75. We use the same radius r_0 as the one used during training and we use a $r_1 = r_0 + 0.05$. Table 5.1 shows the results of this experiment. We observe that nearly half of the corrected classified points have the probability growing suggesting that single-noise certificates would underestimate the real bound.

5.4 A new framework for separating smoothing and information gathering

As we saw in Section 5.2, the main downside of single-noise certificates is the coupling of smoothing and information-gathering. We will now introduce the generalized Neyman-Pearson lemma, which allows to compute the worst-case decision boundary under constraint information.

5.4.1 The generalized Neyman-Pearson Lemma for obtaining worst-case decision boundaries

Theorem 17 (Generalized Neyman-Pearson lemma [Chernoff and Scheffe, 1952]). *Let $q_0, \dots, q_n$ be probability density functions. For any $k_1, \dots, k_n > 0$, we define the Neyman-Pearson set*

$$\mathcal{S}_K = \left\{ q_0(x) \leq \sum_{i=1}^{n} k_i q_i(x) \right\}$$

and the associated Neyman-Pearson function:

$$\Phi_K = \mathbb{1}\{\mathcal{S}_K\}$$

Then for any function $\Phi : \mathcal{X} \to [0,1]$ such that $\int \Phi q_i \, \mathrm{d}\mu \geq \int \Phi_K q_i \, \mathrm{d}\mu$ for all $i \in \{1, \dots, n\}$, we have:

$$\int \Phi_K q_0 \, \mathrm{d}\mu \leq \int \Phi q_0 \, \mathrm{d}\mu$$

Proof. By definition of Φ_K, we have:

$$\int (\Phi - \Phi_K)(q_0 - \sum_{k=1}^{n} k_i q_i) \, \mathrm{d}\mu \geq 0 \tag{5.96}$$

since the integrand is always positive. Hence:

$$\int (\Phi - \Phi_K) q_0 \, \mathrm{d}\mu \geq \sum_{k=1}^{n} k_i \int (\Phi - \Phi_K) q_i d\mu \tag{5.97}$$

Since $\int (\Phi - \Phi_K) q_i \, \mathrm{d}\mu \geq 0$, we have:

$$\int (\Phi - \Phi_K) q_0 \, \mathrm{d}\mu \geq 0 \tag{5.98}$$

which is the desired result. □

The generalized Neyman-Pearson lemma states that if we can fit the set $\mathcal{S}$ to have certain reactions $\int \Phi_K q_i \, \mathrm{d}\mu \geq 0$ to the noise distributions q_i, then it will be the worst-case decision boundary under the noise distribution q_0. The main idea to obtain certificates from this lemma is to notice that when the noise distributions are all isotropic, and q_0 is of the form $q(\cdot - \delta)$, then the whole problem will be invariant by rotation in δ. This means that the response will be the same for any δ of norm ϵ, thus giving a certificate.

Corollary 1 (Noise-based certificates). *Let $\mathcal{Q} = \{q_0, \dots, q_n\}$ be a finite family of isotropic probability density functions, of same center. Let $\epsilon > 0$, and any δ of norm ϵ. If the k_i are such that $\forall i, p(x, \Phi_{\mathcal{K}}, q_i) \leq p(x, h, q_i)$, then we have:*

$$\mathrm{NC}(h, q_0, x, \epsilon, \mathcal{Q}) \geq p(x + \delta, \Phi_{\mathcal{K}}, q_0)$$

Furthermore, this becomes an equality if $\forall i, p(x, \Phi_{\mathcal{K}}, q_i) = p(x, h, q_i)$

This means that by choosing the k_i such that $p(x, \Phi_{\mathcal{K}}, q_i)$ is as close as possible from $p(x, h, q_i)$ while remaining lower, we can get arbitrary close to $\mathrm{NC}(h, q_0, x, \epsilon, \mathcal{Q})$ using the Neyman-Pearson classifier $\Phi_{\mathcal{K}}$.

Note an important difference between multiple-noise certificates and single-noise ones: we use many noise distributions of varying amplitude *at certification time*, but the noise used for smoothing at test time, q_0, remains constant. That is the strength of our framework: dissociating the information gathering process and the smoothing itself.

An advantage of this method is that since the function class $\mathcal{G}$ can only shrink with the number of noises used, a bound obtained with several noises will always be at least as good as the one proposed by [Cohen et al., 2019].

5.4.2 Deriving certificates with information-gathering from several noise distributions

Using the Generalized Neyman-Pearson lemma, we can extract information about the decision boundary using noise distributions $q_0, \dots, q_n$, via Monte-Carlo sampling. We then exploit that information to compute a worst-case decision boundary, which can be used to obtain a certificate:

1. Compute $p(x, h, q_i)$ for all noises q_i via Monte-Carlo sampling.
2. Compute the generalized Neyman-Pearson function $\Phi_{\mathcal{K}}$ by fitting the constants k_i such that $p(x, \Phi_{\mathcal{K}}, q_i) \leq p_i$ with the closest possible approximation.
3. Fix some δ of norm ϵ. If the noises used are all isotropic, we can directly compute the certificate $p(x + \delta, \Phi_{\mathcal{K}}, q_0)$ via sampling.

If the noise distributions are not isotropic or have different centers (breaking the isotropy of the problem), then we must take a lower bound over all $p(x + \delta, \Phi_{\mathcal{K}}, q_0)$ (where $\Phi_{\mathcal{K}}$ depends on δ), as we will do in the proof of Theorem 18. We will also show later that introducing randomness in the certification process allows to obtain certificate even in the non-isotropic case.

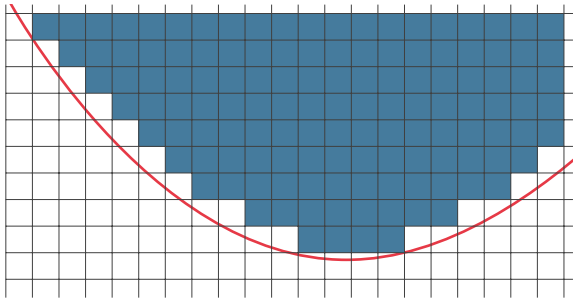


Figure 5.8: Illustration of the Theorem 18. By querying the classifier with uniform noises on the squares of the grid, we can compute an approximation of the true certificate using the blue squares. As we refine the grid with smaller squares, the approximation becomes increasingly good, and converges to the perfect certificate.

We will now show that this framework is enough to derive up to perfect certificates, and does not incur any additionl loss of natural accuracy when compared to the corresponding single-noise certificate.

5.5 Bypassing the limitations of single-noise certificates

5.5.1 General approximation result

Intuition of Theorem 18. For any continuous decision boundary, it is possible to approximate the perfect certificate with arbitrary precision, using information from a finite family of noise distributions. The size of the family used increases with the desired precision. An illustration of this theorem is shown in Figure 5.8.

Theorem 18 (General approximation theorem). *Let q_0 be the uniform noise on the ℓ_∞ ball $B_\infty(0, r)$ for some $r > 0$. For any $\epsilon > 0$, $\xi > 0$ and $x \in \mathbb{R}^d$, there exists a finite family $\mathcal{Q}$ of probability density functions such that:*

$$\mathrm{NC}(h, q_0, x, \epsilon, \mathcal{Q}) \geq \mathrm{PC}(h, q_0, x, \epsilon) - \xi$$

Sketch of proof of Theorem 18. The idea of this proof is to define a grid of disjoint squares covering the space. Then, we can construct a noise-based classifier that returns 1 only on the squares that are entirely contained in the decision region, *i.e.*, a strict underestimate of the true classifier. As the grid gets thinner, the approximation will then converge to the true classifier as a Riemann sum. □

Proof of Theorem 18. Let $n > 0$, and some $x \in \mathcal{X}$. We can construct a grid of $(n(r + 2\epsilon))^d$ disjoint squares of side size $\frac{1}{n}$, of the form $\left[\frac{a_1}{n}, \frac{a_1+1}{n}\right[\times \cdots \times \left[\frac{a_d}{n}, \frac{a_d+1}{n}\right[$ (except the ones on the border of the ball that are closed) that will cover the ball $B^d_\infty(x, r)$, as well as its translation by ϵ in any direction.

Let us call the squares in this grid A_j for $j = 1 \dots m$ and $m = (\frac{d+2\epsilon}{n})^d$. They all have the same volume $V_n = (\frac{1}{n})^d$.

Let q_j denote the probability density function of the uniform noise over A_j:

$$\forall j \in \{1, \dots, m\}, z \in \mathbb{R}^d, q_j(z) = \frac{1}{V_n} \mathbb{1}_{z \in A_j} \tag{5.99}$$

Let $V = \mathrm{Vol}(B^d_\infty(0, r))$. For $j \in \{1, \dots, m\}$, let $p_j = \int h(z) q_j(z) dz$ be the expected response of the true classifier h to noise q_j (i.e. what is observed), and the coefficients k_j such that:

$$k_j = \begin{cases} \frac{V_n}{V} & \text{if } p_j = 1, \textit{i.e.}, h = 1 \text{ almost surely on } A_j \\ 0 & \text{otherwise.} \end{cases} \tag{5.100}$$

We choose these specific coefficients to only "activate" the squares where $h = 1$ almost surely, i.e. that are entirely inside of the decision region.

Let δ be any attack vector of norm ϵ, and $\tilde{q_0} = q_0(\cdot - \delta)$ be the distribution after attack by δ. The support of $\tilde{q_0}$ is $B^d_\infty(-\delta, r)$ which is fully contained in $\bigcup_{i=1}^{m} A_i$.

Let $\Phi_{\mathcal{K}}$ be the Neyman-Pearson function defined by $\mathcal{K} := \{k_1, \dots, k_n\}$:

$$\Phi_{\mathcal{K}}(x') = \begin{cases} 1 & \text{if } \tilde{q_0}(x') \leq \sum_{i=1}^{n} k_i q_i(x') \\ 0 & \text{otherwise} \end{cases} \tag{5.101}$$

We know that $\Phi_{\mathcal{K}} = 1$ outside of $B^d_\infty(x + \delta, r)$ since $\tilde{q_0} = 0$ there. Let $x' \in B^d_\infty(x + \delta, r)$. The A_i are disjoint and cover the ball, so there is exactly one j such that $x' \in A_j$. We then have for any $x' \in A_j$:

$$\Phi_{\mathcal{K}}(x') = \begin{cases} 1 & \text{if } h = 1 \text{ almost surely on } A_j \\ 0 & \text{otherwise} \end{cases} \tag{5.102}$$

Hence $\Phi_{\mathcal{K}}|_{A_j} = \operatorname*{ess\,inf}_{A_j}(h)$, since h has values in $\{0, 1\}$. It follows that:

$$\int \Phi_{\mathcal{K}}(z) \tilde{q_0}(z) dz = \sum_{i=1}^{m} (\operatorname*{ess\,inf}_{A_j}(h)) \, \mathrm{Vol}(A_i \cap B^d_\infty(x - \delta, r)) \tag{5.103}$$

That is a lower Riemann sum for the integral $\int_{B_\infty^d(x-\delta,r)} h$, and so converges to it when $m \to \infty$ as h is Riemann integrable. Hence we can choose n such that, for any δ,

$$\int \Phi_{\mathcal{K}}(z)\tilde{q_0}(z)dz \leq \int h(z)\tilde{q_0}(z)dz + \xi \tag{5.104}$$

which gives us the desired result, since this is true for any δ. □

Theorem 18 shows that it is possible to collect asymptotically perfect information on the decision boundary using only noise-based queries. The main improvement compared to the result of [Mohapatra et al., 2020] is that we reconstruct the base classifier itself, and not just the Gaussian smoothed version of it, hence it works for any smoothing scheme. This shows that the black-box approach to randomized smoothing certification is viable, and can bypass the theoretical limitations when using several noises instead of one.

Furthermore, if we have access to some prior information on the decision boundary, it will be possible to design much more efficient noise-based information gathering schemes. In the following, we present a result demonstrating that we can obtain full information in the case of conical or 2-piecewise linear decision boundaries by using only a few concentric uniform noises.

5.5.2 Adding prior information on the decision boundary

Definition 59 (Noise-based certificate with prior information). *Let $\mathcal{Q}$ be a finite family of probability density functions, $\mathcal{F}$ be a family of classifiers (typically parameterized). Let $q_0 \in \mathcal{Q}$. The $\mathcal{Q}-$noise-based ϵ-certificate with prior information $\mathcal{F}$ for the q_0-randomized smoothing of h at point x is:*

$$\mathrm{NCP}(h, q_0, x, \epsilon, \mathcal{Q}, \mathcal{F}) = \inf_{g \in \mathcal{G}_{\mathcal{Q},\mathcal{F}}} \inf_{\delta \in B(0,\epsilon)} p(x+\delta, g, q_0)$$

where:

$$\mathcal{G}_{\mathcal{Q},\mathcal{F}} = \{g \in \mathcal{F} \mid \forall q \in \mathcal{Q},\ p(x,g,q) = p(x,h,q)\}$$

We will now show that adding prior information about the classifier can drastically improve the power of noise-based certificate. In particular, if we know that the decision boundary is in a parametrized family of curves, we can reach the perfect certificate using a small amount of noise, by reconstituting the parameters. In the case of conical and $2-$piecewise linear decision boundaries, we can do that because of a $1-$to-1 correspondence between the volume growth under concentric noises, and the parameter θ.

Definition 60 (Volume growth for a decision boundary). *Let B_2^d, the ℓ_2-ball in dimension d, $\mathcal{A}$ a set and $r_1 > r_2 \geq 0$. We define the volume growth of $\mathcal{A}$ from r_1 to r_2 as:*

$$\Delta V(\mathcal{A}, r_1, r_2) = \mathrm{Vol}\big(\mathcal{A} \cap B_2^d(0, r_2)\big) - \mathrm{Vol}\big(\mathcal{A} \cap B_2^d(0, r_1)\big)$$

Definition 61 (Linear half-space)**.** *Let $c \geq 0$. The half-space of translation c is, in hypercylindrical coordinates of axis e_1, the set:*

$$H(c) = \left\{ \begin{array}{l} z \in \mathbb{R} \\ \rho \in \mathbb{R}_+ \\ \phi_2, \dots, \phi_{d-1} \in [0, \pi]. \end{array} \middle| \; z > c \right\}$$

Lemma 9 (Growth function for concentric noises). *Let $c \geq 0$, $r_2 > r_1 > c$. Then $\Delta V(C(c, \theta), r_1, r_2)$ is a continuous, increasing function of θ, that is a bijection from $[0, \pi]$ to $[0, \Delta V(H(c), r_1, r_2)]$.*
The same result holds for 2-piecewise linear sets of parameter θ.

Proof of Lemma 9. Let $r_2 > r_1 > 0$, $c > 0$. Let $\mathcal{C}(c, \theta)$ be the cone of revolution of peak c and angle θ. Let $\mathcal{B}_\theta(r_1, r_2) = \mathcal{C}(c, \theta) \cap B_2^d(0, r_2) \setminus B_2^d(0, r_1)$ and $\mathcal{A}(r_1, r_2) = H(c) \cap B_2^d(0, r_2) \setminus B_2^d(0, r_1)$.

First note that $\mathrm{Vol}(\mathcal{B}_\theta(r_1, r_2)) = \mathrm{Vol}(\mathcal{C}(c, \theta) \cap B_2^d(0, r_2)) - \mathrm{Vol}(\mathcal{C}(c, \theta) \cap B_2^d(0, r_1)) = \Delta V(\mathcal{C}(c, \theta), r_1, r_2)$, and similarly $\mathrm{Vol}(\mathcal{A}(r_1, r_2)) = \Delta V(H(c), r_1, r_2)$

To compute the volume of $\mathcal{B}_\theta(r_1, r_2)$, we must cut the integral into three zones: where the bound is the cone, and where it is the surface of either of the balls. There exist a constant K (dependent on the dimension, and containing the integration in all the variables ϕ_i in hyper-spherical coordinates) such as:

$$\mathrm{Vol}(\mathcal{B}_\theta(r_1, r_2)) = \mathrm{Vol}(\mathcal{C}(c, \theta) \cap B_2^d(0, r_2) \setminus B_2^d(0, r_1)) \tag{5.105}$$

$$= \frac{1}{K} \left[\int_{x=r_1 \cos\theta}^{r_1} \int_{\sqrt{r_1^2 - x^2}}^{x \tan\theta} \rho^{d-2} \, \mathrm{d}\rho \, \mathrm{d}x + \int_{r_1}^{r_2 \cos\theta} \int_{\rho=0}^{x \tan\theta} \rho^{d-2} \, \mathrm{d}\rho \, \mathrm{d}x + \int_{r_2 \cos\theta}^{r_2} \int_{\rho=0}^{\sqrt{r_2^2 - x^2}} \rho^{d-2} \, \mathrm{d}\rho \, \mathrm{d}x \right] \tag{5.106}$$

It follows that $\mathrm{Vol}(\mathcal{B}_\theta(r_1, r_2))$ is a continuous function of θ.

Furthermore, if $\theta_2 > \theta_1$, then $\mathcal{C}(c, \theta_1) \subset \mathcal{C}(c, \theta_2)$, so $\mathrm{Vol}(\mathcal{B}_{\theta_1}(r_1, r_2)) \leq \mathrm{Vol}(\mathcal{B}_{\theta_2}(r_1, r_2))$, and the function is increasing.

For $\theta = 0$, $\mathrm{Vol}(\mathcal{B}_\theta(r_1, r_2)) = 0$, and for $\theta = \frac{\pi}{2}$, $\mathcal{B}_{\theta_1}(r_1, r_2)) = \mathcal{A}(r_1, r_2)$. Hence the result. When $\theta = \pi$, the cone becomes the half-space $H(c)$.

□

Intuition of Theorem 19. If we know that the decision boundary is conical or 2-piecewise-linear, then its parameter θ can be perfectly identified using only the information from two concentric noises. This then allows us to compute the perfect certificate by Monte-Carlo sampling, knowing the true decision boundary. Figure 5.9 is an intuitive illustration of the proposition. We can see that the volume of the cone captured by the balls is a strictly non-decreasing function of θ, so that there is a $1-$to-1 correspondence between the gradient of the volume and the angle of the cone.

Theorem 19 (Perfect certificate for conical decision boundaries). *Let $\theta_0 \in \left[0, \frac{\pi}{2}\right]$. Let h be a classifier whose decision boundary is the cone $\mathcal{C}(0, \theta_0)$. Let $\mathcal{F}$ be the family of all classifiers with a decision boundary of the form $\mathcal{C}(0, \theta)$. Then there exists uniform noises q_1 and q_2 such that, for any noise q_0, and any $x \in \mathbb{R}^d, \epsilon > 0$:*

$$\mathrm{NCP}(h, q_0, x, \epsilon, \{q_1, q_2\}, \mathcal{F}) = \mathrm{PC}(h, q_0, x, \epsilon)$$

Proof of Theorem 19. This is an immediate consequence of Lemma 9 : From the information of two noises, we can perfectly identify the parameter θ, and thus compute the perfect certificate as $\mathbb{P}_{X \sim q_0(x)}[X + \epsilon e_1 \in \mathcal{C}(0, \theta)]$. □

Also, by a direct extension of Theorem 19, a small number of concentric noises are enough to obtain full information on general cones $\mathcal{C}(c, \theta)$, in two steps:

- Evaluate the distance c to the decision boundary by finding the threshold such that $p(x, h, q(r)) \neq 1$. This can be done using a binary search with a given precision;
- Use two noises of radius r_1 and r_2 to identify the angle θ as presented in Theorem 19.

This hints at a more general result for piecewise linear decision boundaries (which includes all neural networks with ReLUs activations): it may be possible to gather perfect information using only a limited number of concentric noises to "map" the fractures of the decision boundary.

Designing certificates thus shifts from a classifier-agnostic problem to a more classifier-specific one: any prior information on the decision boundary can help guide the choices

of noises used at certification time. This also suggests that we could also choose the base classifier not only because of its efficiency, but to obtain some desirable properties that facilitate the certification process. This opens up a wide area of research.

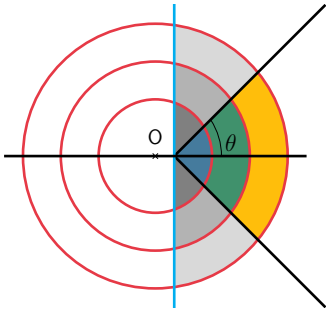


Figure 5.9: Illustration of Theorem 19. We see that the difference of volume captured between two balls (blue, green and yellow zones) grows with θ. For $\theta < \frac{\pi}{2}$, the volume growth is lower than for a hyperplane decision boundary (the cyan line) at the same distance. The difference is shown by the gray zones.

5.6 Choosing the noises for information collection

5.6.1 Discussion on computational cost

In this section, we analyze the computational challenges of implementing noise-based certificates, and explore some avenues to reduce them. There are currently three main obstacles to computing noise-based certificates using the generalized Neyman-Pearson Lemma:

1. Computing integrals via Monte Carlo sampling in high-dimension can become very costly as this technique suffers from the curse of dimensionality.
2. When computing the integrals in high dimensions, numbers can become very small or very large, leading to computational instability.[1]
3. Finally, fitting the k_i to compute the generalized Neyman-Pearson set is a hard stochastic optimization problem.

We show that we can bypass problems 1. and 2. when using Gaussian noise for information collection. Furthermore, uniform noise as an information gathering method considerably reduces problem 3, although suffering from problems 1 and 2.

[1] For example, the volume of an ℓ_2 ball in dimension 784 (MNIST dimension) is approximately equal to $\exp(-1503.90)$.

5.6.2 Combinatorial fitting with uniform noises

When using uniform noises, the Neyman-Pearson set takes a very interesting shape. Let $A_1, \dots, A_n$ be the sets on which the noise-gathering uniform distributions are taken, A_0 the set used for the smoothing uniform distribution, and $\tilde{A_0}$ the set translated by ϵ. Then the Neyman-pearson set becomes :

$$S = \left\{ x \in \mathbb{R}^d | \mathbb{1}_{x \in \tilde{A_0}} \leq \sum_{i=1}^{n} k_i \mathbb{1}_{x \in A_i} \right\} \tag{5.107}$$

If $x \in \tilde{A_0}$, the left hand side will be 0 and $x \in S$. If $x \in \tilde{A_0} \cap A_1$ and $x \notin A_2, \dots, A_n$, then we have :

$$x \in S \iff 1 \leq k_1 \tag{5.108}$$

since $\mathbb{1}_{x \in A_i} = 0$ for $i = 2 \dots n$. Each indicative function can only take two values, namely 0 and 1, and they each remain constant on the corresponding set. This means that the generalized Neyman-Pearson set is constant of each set of the from
$A_{i_0} \cap \dots \cap A_{i_k} \cap (A_{j_0} \cup \dots \cup A_{j_l})^{\complement}$.

It follows that the generalized Neyman-Pearson set can have at most 2^n values.

Theorem 20 (Computing the k_i for uniform noises). *Let $q_0, \dots, q_n$ be uniform distributions, where $n \ll d$. Then there are only at most 2^n possible values for the generalized Neyman-Pearson set S.*

This means that the exact values of the k_i do not matter, only the possible values of the Neyman-Pearson set S. The research of S thus shifts from a hard optimization problem to a combinatorial problem with only at most 2^n values to try where n correspond to the number of noise and is usually much lower than the input dimension. Note that by taking $n = 1$, the certificate reduces to the single-noise certificate, and increasing the number of noises can only improve it. Also, we should remark that smart choices of noises can make that combinatorial problem easier in practice, since the support of the distributions does not necessarily intersect with each other.

5.6.3 Lower dimension sampling with Gaussian noises

Theorem 21 (Gathering information from Gaussian noises). *Let q_0 be any isotropic probability distribution, $\sigma_1, \dots, \sigma_n > 0$. For $i = 1 \dots n$ let $q_i \sim \mathcal{N}(0, \sigma_i)$ be the noises used for information gathering. Let $S_{k_1 \dots k_n}$ be the corresponding Neyman-Pearson set, for any combination of parameters $k_1, \dots, k_n > 0$.*
Then $\mathbb{P}[\mathcal{N}(0, \sigma_i^2) \in S_{k_1 \dots k_n}]$ can be computed using a Monte Carlo sampling in dimension 2 from a χ distribution with $d-2$ degrees of freedom.

Sketch of proof for Theorem 21. The key of this proof is again the invariance by rotation of the generalized Neyman-Pearson set around the direction e_1 of the attack. This allows us to separate $\|z\|^2$ into two components, one along e_1, which follows a 1-dimensional normal distribution, and one in $e_1^\perp$, whose norm follows a χ distribution with $d-2$ degrees of freedom. □

Proof. Let $x \in \mathbb{R}^d$. q_0 is an isotropic probability density function, which means that there exists a function p_0 such that $\forall x \in \mathbb{R}^d, q_0(x) = p_0(\|x\|^2)$. For $i = 1 \dots n$, we have:

$$\mathbb{P}\left[\mathcal{N}(x, \sigma_i^2) \in \mathcal{S}_{k_1,\dots,k_n}\right] = \frac{1}{(2\pi)^{\frac{d}{2}}\sigma_0^d} \int_{\mathbb{R}^d} \exp\left(-\frac{\|u-x\|^2}{2\sigma_0^2}\right) \mathbb{1}_{u \in \mathcal{S}_{k_1,\dots,k_n}} \, \mathrm{d}u \quad (5.109)$$

where $\mathcal{S}$ is the Neyman-Person set defined as:

$$\mathcal{S}_{k_1,\dots,k_n} = \left\{ u \in \mathbb{R}^d \;\middle|\; p_0(\|u-x-\delta\|^2) \leq \sum_{i=0}^{n} k_i \exp\left(-\frac{\|u-x\|^2}{2\sigma_i^2}\right) \right\} \quad (5.110)$$

Where the k_i are defined as in Corollary 1.

By expressing Equation (5.109) with hypercylindrical coordinates of center x and axis δ, we have:

$$\mathbb{P}\left[\mathcal{N}(x, \sigma_i^2) \in \mathcal{S}_{k_1,\dots,k_n}\right] = \frac{1}{(2\pi)^{\frac{d}{2}}\sigma_i^d} \int_{\mathbb{R}^d} \exp\left(-\frac{\|u-x\|^2}{2\sigma_i^2}\right) \mathbb{1}_{x \in \mathcal{S}} \, \mathrm{d}x \quad (5.111)$$

$$= \frac{1}{(2\pi)^{\frac{d}{2}}\sigma_0^d} \int_{\substack{\mu \in \mathbb{R} \\ r \in \mathbb{R}^+ \\ \phi_i \in [-\pi, \pi] \\ \phi_{d-2} \in [0, 2\pi]}} \exp\left(-\frac{r^2+\mu^2}{2\sigma^2}\right) \mathbb{1}_{(r,\mu) \in \tilde{\mathcal{S}}} J \, \mathrm{d}r \, \mathrm{d}\mu \, \mathrm{d}\phi_1 \dots \mathrm{d}\phi_{d-2} \quad (5.112)$$

where from [Blumenson, 1960], the Jacobian J of the change of variables is:

$$J = r^{d-2} \prod_{k=1}^{d-2} \sin^k \phi_{d-1-k} \tag{5.113}$$

and where $\tilde{\mathcal{S}}$ is the updated Neyman-Person set:

$$\tilde{\mathcal{S}} = \left\{ r, \mu \in \mathbb{R} \;\middle|\; p_0(r^2 + (\mu - \epsilon)^2) \leq \sum_{i=0}^{n} k_i \exp\left(-\frac{r^2 + \mu^2}{2\sigma_i^2}\right) \right\} \tag{5.114}$$

Given that the indicator function is independent of the $\phi_1, \ldots, \phi_{d-2}$, we can rearrange the above equation as follows:

$$\begin{aligned} \mathbb{P}\big[\mathcal{N}(x, \sigma_i^2) \in \mathcal{S}_{k_1,\ldots,k_n}\big] = \frac{1}{(2\pi)^{\frac{d}{2}} \sigma_i^d} \left(\int_{\substack{\mu \in \mathbb{R} \\ r \in \mathbb{R}^+}} \exp\left(-\frac{r^2 + \mu^2}{2\sigma^2}\right) r^{d-2} \mathbb{1}_{(r,\mu) \in \tilde{\mathcal{S}}} \, \mathrm{d}r \, \mathrm{d}\mu \right) \\ \left(\int_{\substack{\phi_1,\ldots,\phi_{d-3} \in [-\pi,\pi] \\ \phi_{d-2} \in [0,2\pi]}} \prod_{k=1}^{d-2} \sin^k \phi_{d-1-k} \, \mathrm{d}\phi_1 \ldots \mathrm{d}\phi_{d-2} \right) \end{aligned} \tag{5.115}$$

By setting A as:

$$A = \int_{\substack{\phi_1,\ldots,\phi_{d-3} \in [-\pi,\pi] \\ \phi_{d-2} \in [0,2\pi]}} \prod_{k=1}^{d-2} \sin^k \phi_{d-1-k} \, \mathrm{d}\phi_1 \ldots \mathrm{d}\phi_{d-2} \tag{5.116}$$

we have:

$$\begin{aligned} &\mathbb{P}\big[\mathcal{N}(x, \sigma_i^2) \in \mathcal{S}_{k_1,\ldots,k_n}\big] \\ &= \frac{A}{(2\pi)^{\frac{d}{2}} \sigma_0^d} \left(\int_{\mu \in \mathbb{R}} \int_{r \in \mathbb{R}^+} \exp\left(-\frac{r^2}{2\sigma_0^2}\right) \exp\left(-\frac{\mu^2}{2\sigma^2}\right) r^{d-2} \mathbb{1}_{(r,\mu) \in \tilde{\mathcal{S}}} \, \mathrm{d}r \, \mathrm{d}\mu \right) \end{aligned} \tag{5.117}$$

Finally, we can express this probability with an expected value over a Gaussian and Chi distribution:

$$\mathbb{P}\left[\mathcal{N}(x, \sigma_i^2) \in \mathcal{S}_{k_1,\dots,k_n}\right] = \mathbb{E}_{\substack{\mu\sim\mathcal{N}(0,\sigma_i^2) \\ r\sim\chi(d-1,0,\sigma_i^2)}}\left[\mathbb{1}_{(r,\mu)\in\tilde{\mathcal{S}}}\right] \tag{5.118}$$

which concludes the proof. □

This means that whatever the noise used at smoothing time, we can easily gather information from Gaussian noises, since the Neyman-Pearson set needs only be sampled in dimension 2 to fit the k_i.

5.6.4 Toward dimension-independent certificates with high-probability certification

Definition 62 (High Probability Certification). *An α-probable ϵ-certificate for the q_0 randomized smoothing is a value $v \in \mathbb{R}$ and a probability distribution Q such that, for any δ of norm at most ϵ:*

$$\mathbb{P}_Q[p(x+\delta, h, q_0) \geq v] \geq 1 - \alpha$$

Note that the randomness is not on the attack, but on the noises used at certification time. We do not certify for "the majority of attacks", which would not work since attacks are engineered, thus worst-case guarantees are required.

Theorem 22 (Asymptotic dimension-independent certificate)**.** *Let $q_0, \dots, q_n$ are normal distribution of same variance σ, centered on points $x, z_1, \dots, z_n$, where the z_i constitute a random family of orthonormal vectors, at constant distance from x. For any $\alpha \in (0,1), \epsilon > 0$, there exists an α-probable ϵ-certificate v_d and a value $v \in [0,1]$ such that $v_d \xrightarrow[d\to\infty]{} v$ and v can be written as $\mathbb{E}[f(V)]$ for some function f and where V is a random variable in dimension at most n+1 such that both V and f are independent from the dimension d.*

Furthermore, that convergence is fast : for CIFAR10, with $\alpha = 0.05$, the approximation term $\frac{v_d}{v}$ is already of 0.99. This means that we can compute the value v (independent of the dimension) and use it as a high-probability certificate for the dimensions of real-world datasets.

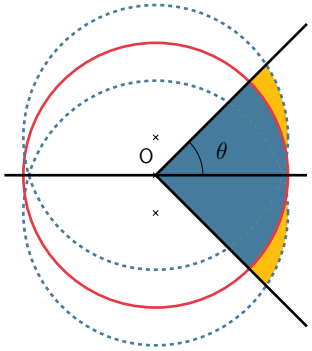


Figure 5.10: Illustration of Theorem 22. We can use translated noises to gather information (here identify the yellow zones), and in high dimensions the random translations will be almost orthogonal to the attack with high probability, whatever the direction of the attack.

The core argument of Theorem 22 is that in high dimension, for any attack vector δ, random directions will be almost orthogonal to δ with high probability. This allows us to obtain a certificate using the generalized Neyman-Pearson lemma, although our our noises do not have the same center so the problem is not isotropic.

This means that, by introducing some randomness during the information gathering, we can obtain a certificate using non-isotropic noise that is very easy to compute, since independent on the dimension. This removes one of the main barriers for our certificates to scale in very large dimension.

To prove this theorem, the main idea is to bound the dot product between δ and the x_i, using concentration inequalities as the dimension increases.

Lemma 10. *Let u be a random vector drawn uniformly on the unit sphere of $\mathbb{R}^d$, $\delta \in \mathbb{R}^d$ of norm ϵ. Then :*

$$\mathbb{P}\left[|\langle u, \delta\rangle| \geq \frac{\epsilon}{\sqrt{(d+2)\alpha}}\right] \leq \alpha \tag{5.119}$$

Proof. This is the direct consequence of Chebyshev's inequality, since $\langle u, \delta\rangle$ is a random vector of expectation 0 (by symmetry), and :

$$V(u, \delta) = \sum_{i=1}^{d} V(\delta_i u_i)$$

$$= \sum_{i=1}^{d} \delta_i^2 V(u_i)$$
$$= \frac{\|\delta\|^2}{d+2} = \frac{\epsilon^2}{d+2}$$

□

Lemma 11. *Let $v_1, \dots, v_n$ be random, orthogonal vectors, $e_i = \frac{v_i}{\|v_i\|}$ δ a vector of norm ϵ, and δ_P the orthogonal projection of δ on $P = \mathrm{Span}(e_1, \dots, e_n)$. Then:*

$$\mathbb{P}\left[\|\delta_P\|^2 \leq \frac{n\epsilon^2}{(d+2)\alpha}\right] \geq (1-\alpha)^n \tag{5.120}$$

Proof. $\|\delta_P\|^2 = \sum_{i=1}^{n} \langle \delta, v_i \rangle^2$, so we have :

$$\mathbb{P}\left[\|\delta_P\|^2 \leq \frac{n\epsilon^2}{(d+2)\alpha}\right] \geq \mathbb{P}\left[\forall i, |\langle v_i, \delta \rangle| \leq \frac{\epsilon}{\sqrt{(d+2)\alpha}}\right] \tag{5.121}$$
$$\geq (1-\alpha)^n \tag{5.122}$$

□

we can now prove the theorem :

Proof of Theorem 22. Let z_0 be the center of the Gaussian noise used at train and test time. Let z_i $(i = 1 \dots, n)$ be the centers of new Gaussian noises used to gather information. We choose the z_i as random vectors on the unit sphere, and orthonormalize them such that $(z_0 - z_i)$ are an orthogonal family.

Let $P = \mathrm{Span}(z_1, \dots, z_n)$, and $\delta = \delta_P + \delta_{P^\perp}$ where $\delta_P \in P$ and $\delta_{P^\perp} \in P^\perp$ and let $\tilde{\epsilon} = \|\delta_{P^\perp}\|$.

We use a coordinate system centered on z_0, with $\mu_i = \left\langle \frac{z_0 - z_i}{\|z_0 - z_i\|}, z - z_0 \right\rangle (i = 1 \dots m)$, $t = \left\langle \frac{\delta_{P^\perp}}{\epsilon}, x - z_0 \right\rangle$, and r, ϕ_i $(i = 1 \dots d - m - 2)$ the hypercylindrical coordinates of axis $\delta_{P^\perp}$ in $P^\perp$. Let $d_i = \|z_0 - z_i\|$. Let π_P be the orthogonal projection on P, $\pi_{P^\perp}$ on $P^\perp$. Then:

$$\|z - \delta\|^2 = \|\pi_P(z - \delta)\|^2 + \|\pi_{P^\perp}(z - \delta)\|^2 \tag{5.123}$$
$$= \|\pi_P(z) - \delta_P\|^2 + \|\pi_{P^\perp}(z) - \delta_{P^\perp}\|^2 \tag{5.124}$$
$$= \|\pi_P(z)\|^2 + \|\delta_P\|^2 - 2\langle \pi_P(z), \delta_P \rangle + r^2 + (t - \tilde{\epsilon})^2 \tag{5.125}$$

$$= \sum \mu_j^2 + r^2 + (t - \tilde{\epsilon})^2 + \|\delta_P\|^2 - 2\langle \pi_P(z), \delta_P \rangle \tag{5.126}$$

We can also write :

$$\forall i = 1 \dots m, \tag{5.127}$$
$$\|z - z_i\|^2 = \|z - z_0\|^2 + \|z_0 - z_i\|^2 - 2\langle z - z_0, z_0 - z_i \rangle \tag{5.128}$$
$$= r^2 + \sum_{j=1}^{m} \mu_j^2 + t^2 + d_i^2 - 2\mu_i d_i \tag{5.129}$$

The certificate given by the generalized Neyman-Pearson lemma for a given δ is:

$$\tilde{p} = \mathbb{P}\big[\mathcal{N}(z_0 + \delta, \sigma^2) \in S\big] \tag{5.130}$$
$$= K_1 \int \exp\big(-\|z - \delta\|^2\big) \mathbb{1}_S \, \mathrm{d}z \tag{5.131}$$
$$= K_1 \int \exp\left(-\frac{r^2}{2\sigma^2}\right) \exp\left(-\frac{\sum_{j=1}^{n} \mu_j^2}{2\sigma^2}\right) \exp\left(-\frac{(t - \tilde{\epsilon})^2}{2\sigma^2}\right)$$
$$\exp\left(-\frac{\|\delta_P\|^2}{2\sigma^2}\right) \exp\left(\frac{2\langle \pi_P(z), \delta_P \rangle}{2\sigma^2}\right) \mathbb{1}_S r^{d-n-2} \, \mathrm{d}r \, \mathrm{d}t \, \mathrm{d}\mu_i \tag{5.132}$$
$$\geq K_1 \int a(r, \mu_j, t) \exp\left(-\frac{n\epsilon^2}{2\sigma^2(d+2)\alpha}\right)$$
$$\exp\left(-\sum_j \mu_j \frac{\epsilon}{2\sigma^2\sqrt{(d+2)\alpha}}\right) \mathbb{1}_S r^{d-n-2} \, \mathrm{d}r \, \mathrm{d}t \, \mathrm{d}\mu_i \tag{5.133}$$

Where K_1 is the normalization constant for a normal distribution of variance σ^2. Hence, with probability at least $(1 - \alpha)^{n+1}$, by Lemma 10.

Where:

$$a(r, \mu_j, t) = \exp\left(-\frac{r^2}{2\sigma^2}\right) \exp\left(-\frac{\sum_{j=1}^{n} \mu_j^2}{2\sigma^2}\right) \exp\left(-\frac{(t - \tilde{\epsilon})^2}{2\sigma^2}\right) \tag{5.134}$$

and

$$b(r, \mu_i, t) = \sum_{i=1}^{m} k_i \exp\left(-\frac{r^2 + \sum_{i=1}^{m} \mu_j^2 + t^2 + d_i^2 - 2\mu_i d_i}{2\sigma^2}\right) \tag{5.135}$$

Then, we need to obtain a lower subset of S. with high probability.

$$S = \left\{ r > 0, \mu_i, t \in \mathbb{R}^d \;\middle|\; a(r, \mu_j, t) \leq \exp\left(-\frac{\|\delta_P\|^2}{2\sigma^2}\right) \exp\left(\frac{2\langle z - z_0, z_0 - x_i\rangle}{2\sigma^2}\right) b(r, \mu_j, t) \right\} \tag{5.136}$$

$$\supset \left\{ r > 0, \mu_i, t \in \mathbb{R}^d \;\middle|\; a(r, \mu_j, t) \leq \exp\left(-\frac{n\epsilon^2}{2\sigma^2(d+2)\alpha}\right) \right.$$
$$\left. \exp\left(-\sum_j \mu_j \frac{\epsilon}{2\sigma^2\sqrt{(d+2)\alpha}}\right) b(r, \mu_j, t) \right\} \tag{5.137}$$

We can further simplify $\tilde{S}$ by posing $c_0(n, d, \alpha) = \exp\left(-\frac{n\epsilon^2}{2\sigma^2(d+2)\alpha}\right)$, $a(d, \alpha) = \frac{\epsilon}{2\sigma^2\sqrt{(d+2)\alpha}}$ and remarking that :

$$\tilde{S} = \left\{ r > 0, \mu_i, t \in \mathbb{R}^d \;\middle|\; \exp\left(-\frac{r^2}{2\sigma^2}\right) \exp\left(-\frac{\sum_{j=1}^n (\mu_j - a(d,\alpha))^2}{2\sigma^2}\right) \exp\left(-\frac{(t-\tilde{\epsilon})^2}{2\sigma^2}\right) \right.$$
$$\left. \leq c_0 \sum_{i=1}^m k_i \exp\left(-\frac{r^2 + \sum_{i=1}^m \mu_j^2 + t^2 + d_i^2 - 2\mu_i d_i}{2\sigma^2}\right) \right\} \tag{5.138}$$
$$= \left\{ r > 0, \mu_i, t \in \mathbb{R}^d \;\middle|\; \exp\left(-\frac{(t-\tilde{\epsilon})^2}{2\sigma^2}\right) \leq c_0 \sum_{i=1}^m k_i \exp\left(-\frac{t^2 + d_i^2 - 2\mu_i d_i}{2\sigma^2}\right) \right\} \tag{5.139}$$

which only depends on variables $t, \mu_1, \ldots, \mu_n$, so we will write it $\tilde{S}(t, \mu_1, \ldots, \mu_n)$. Hence:

$$\tilde{p} \geq K_1 \int \exp\left(-\frac{r^2}{2\sigma^2}\right) \exp\left(-\frac{\sum_{j=1}^n \mu_j^2}{2\sigma^2}\right) \exp\left(-\frac{(t-\tilde{\epsilon})^2}{2\sigma^2}\right)$$
$$\exp\left(-\frac{n\epsilon^2}{2\sigma^2 d_0 \alpha(1-\zeta)}\right) r^{d-n-2} \mathbb{1}\left\{\tilde{S}(t, \mu_1, \ldots, \mu_n)\right\} \mathrm{d}r \, \mathrm{d}t \, \mathrm{d}\mu_i$$
$$\geq K_2 \exp\left(-\frac{n\epsilon^2}{2\sigma^2 d_0 \alpha(1-\zeta)}\right) \int \exp\left(-\frac{\sum_{j=1}^n (\mu_j - a(d,\alpha))^2}{2\sigma^2}\right) \exp\left(-\frac{(t-\tilde{\epsilon})^2}{2\sigma^2}\right)$$
$$\exp\left(-\frac{n\epsilon^2}{2\sigma^2 d_0 \alpha(1-\zeta(\alpha))}\right) \mathbb{1}_{\tilde{S}} \, \mathrm{d}t \, \mathrm{d}\mu_i \tag{5.140}$$
$$\geq \mathbb{E}_{\substack{\mu_j \sim \mathcal{N}(a(d,\alpha), \sigma^2) \\ t \sim \mathcal{N}(\epsilon, \sigma^2)}} \left[\mathbb{1}\left\{\tilde{S}(t, \mu_1, \ldots, \mu_n)\right\}\right] \tag{5.141}$$

Where $K_2 = \frac{K_1}{\int \exp\left(-\frac{r^2}{2\sigma^2}\right) r^{d-2}\,\mathrm{d}r}$, which corresponds to removing the normalization of the χ law.

When $d \to \infty$, we have $c_0(n, d, \alpha) \to 1$ and $a(d, \alpha) \to 0$. Let :

$$\mathcal{S}_0 = \left\{ r > 0, \mu_i, t \in \mathbb{R}^d \;\middle|\; \exp\left(-\frac{(t-\tilde{\epsilon})^2}{2\sigma^2}\right) \leq \sum_{i=1}^{m} k_i \exp\left(-\frac{t^2 + d_i^2 - 2\mu_i d_i}{2\sigma^2}\right) \right\} \tag{5.142}$$

Then

$$\mathbb{E}_{\substack{\mu_j \sim \mathcal{N}(a(d,\alpha),\sigma^2) \\ t \sim \mathcal{N}(\epsilon,\sigma^2)}} \left[\mathbb{1}\left\{ \tilde{S}(t, \mu_1, \ldots, \mu_n) \right\} \right] \xrightarrow[d\to\infty]{} \mathbb{E}_{\substack{\mu_j \sim \mathcal{N}(0,\sigma^2) \\ t \sim \mathcal{N}(\epsilon,\sigma^2)}} [\mathbb{1}\{\mathcal{S}_0(t, \mu_1, \ldots, \mu_n)\}] \tag{5.143}$$

which can be computed independently of the dimension. □

5.7 Summary of our study of Randomized smoothing certification

We have shown that the limitations of randomized smoothing are a byproduct of the certification method, namely the combination of the smoothing and information gathering steps. We show that by dissociating the two processes, and using multiple distributions for the information gathering, it is possible to circumvent these limitations without affecting the standard accuracy of the classifier. This opens up a whole new field of classifier-specific certification, with the guarantee of always performing better than single-noise certificates, and without any additional loss in standard accuracy. Furthermore, it is now possible to optimize the choice of the base classifier, and use prior information in the certification process. However, much remains to do for noise-based certificates to be viable:

Computing the k_i. Theorem 21 successfully reduces the difficulty of the problem. However, even with those simplifications, fitting the k_i of the generalized Neyman-Pearson set remains a difficult stochastic optimization problem. Indeed, each step requires the computation of an integral via Monte Carlo sampling, and many steps may be necessary to reach the desired precision. A potential direction of research would be to use the relaxation introduced by [Dvijotham et al., 2020] for an easier way to compute approximation of the Neyman-Pearson set. Both techniques from [Yang et al., 2020] to compute ordinary Neyman-Pearson sets can also be extended to our general sets, for more computational efficiency.

Choosing the base classifier h. Now that our certificates use more specific information on the classifier, it is possible to optimize the combination between the base classifier and the noise distributions used. For example, we may adjust our training to ensure that the decision boundary has the highest possible curvature, since it is where our new certificates will shine. The work from [Salman et al., 2019], which combined noise injection and adversarial training [Madry et al., 2018] during the training, suggest that different training schemes can have an important impact on the certification performance. Recently, this line of research has been studied and further improvements have been devised [Zhai et al., 2020, Jeong and Shin, 2020, Zhen et al., 2021, Wang et al., 2021]. In the context of our framework, new training schemes could be devised to improve the local curvature at each point by adjusting the amount of noise injected.

6 Conclusion and open problems

6.1 Summary of the results

In this thesis, we studied the problem of existence, computability and certifiation of optimal classifiers in the presence of adversaries, using the prisms of game theory, optimal transport and Neyman-Pearson certification. Our results can be summarized as follows :

- We studied the conditions under which "optimal" attacks and defenses can exist, and be computable in practice. We showed that the deterministic regime is not fit for stable equilibria, but randomization is a promising lead of research, that create new equilibria, increases their stability, and provides better performances under attack for the defender.
- We quantified the gap between single-noise certificates and the perfect one, showing that although gaussian noise smoothing cannot evade the impossibility results, uniform noise smoothing can, as that gap explodes with the dimension of the problem for zones where the local curvature of the decision boundary is high.
- We introduced a new framework to compute general noise-based certificates, separating the smoothing from the information gathering to induce no loss of accuracy. By using symmetries, invariances and high-probability certification, we devised a certificate that can be computed independently from the dimension of the problem, and provided several insights into computational techniques.

We hope that this thesis will help the community by providing a new perspective on adversarial classification. Our results open the way for several research areas, that we discussed in this thesis. We now wish to discuss three more general open problems, that we think are important for the development of the field.

Original Input	Connoisseurs of Chinese film will be pleased to discover that Tian's meticulous talent has not withered during his enforced hiatus.	Prediction: **Positive (77%)**
Adversarial example [Visually similar]	**Aonnoisseurs** of Chinese film will be pleased to discover that Tian's meticulous talent has not withered during his enforced hiatus.	Prediction: **Negative (52%)**
Adversarial example [Semantically similar]	Connoisseurs of Chinese **footage** will be pleased to discover that Tian's meticulous talent has not withered during his enforced hiatus.	Prediction: **Negative (54%)**

Figure 6.1: Adversarial examples for NLP. In this context, the imperceptibility makes less sense visually than semantically : the first attack would be immediately detected by a human as an "error", whereas the second, although very different in terms of norm, would go unnoticed.

6.2 Open problem 1 : A better understanding of the attacks and defenses

As mentioned in chapter 3, modeling the Attacker is not an easy task. We identify three major difficulties :

- **Imperceptibility :** what does it mean for a modification to be humanly imperceptible ? Norm constraints can be a very imprecise proxy, especially for structured applications such as text (see fig. 6.1). Field-specific notions of imperceptibility need to be modeled, and their influence on the existence and stability of equilibria studied;
- **Computation :** Imperceptibility constraints can be intractable, and require smoothed approximation. Any unified theory of adversarial attacks must take into account all "reasonable" smoothings of the problem, such as the Carlini&Wagner cost from chapter 3;
- **Additional constraints :** Adversaries are never only constrained by imperceptibility. There are many other factors that may come into play depending on the situation, from limited number of queries to computational resource, partial access to the input, probabilistic success of the attack, etc.

A more general study of the adversary would greatly benefit the field, as current models remain way too simple, often giving the theoretical opponent more power than it has in practice.

Similarly, real-life constraints on the classifier can considerably alter the analysis. This includes :

- **The search space :** Restricting the search to measurable, continuous, or even lipschitz functions can make a big difference, as highlighted in chapter A. Further refinements would require a deeper analysis of the classes of functions used for each application, like convolutional or residual neural networks.
- **Surrogate losses :** In a similar way, we have seen that the choice of the surrogate loss function may drastically alter the existence of equilibria. We have studied convex surrogates as well as the $0/1$ loss, but as seen in Appendix A, convex surrogates are not adversarially consistent. New losses need to be crafted, and the game they generate may behave very differently from the current one.

6.3 Open problem 2 : Increasing the stability of Nash equilibria via strategy restrictions

As we saw in chapter 3, restricting the defender to some class of functions has the potential to increase the stability of equilibria. We did that using randomization, but a variety of other schemes may be used :

- **Randomized Smoothing** seem like the next logical step to study, as we have shown that its certification process is still a promising leadof research. It works in a similar fashion as noise injection, while being deterministic. Smoothing the decision boundary of the classifier limits the local variation around any given point, and so the local margin of reaction to any given attack. This would be a source of stability for equilibria.
- **Lipschitz Networks** are a promising lead in the world of robust classification (see for instance [Meunier et al., 2022a] and [Gouk et al., 2021]). As they limit the local variation of a function, this restriction may be an interesting source of stability. Furthermore, we know that under certain conditions, lipschitz optimal solutions can exist, see chapter 3 chapter A.

More generally, will we always encounter an accuracy/stability tradeoff of some sorts ? In other words :

> Which class of functions allows for the best increase in the stability of equilibria with minimal loss of natural accuracy ?

6.4 Open problem 3 : A deeper understanding of the adversarial attack phenomenon

One of the major difficulties with theoretical studies of adversarial classification is how mysterious the phenomenon remains. We lack tools to analyze real-world decision boundaries, and understand the precise reasons behind the existence of attacks.

> Why do adversarial examples exist, and are they unavoidable ?

Several aspects may play an important role :

- **Curvature of the decision boundary.** Intuitively, overfitting can create "spike" zones in the classification region, which may go much further into the "territory" of another class as it should (see fig. 6.2). This lead has only slightly been studied, as evaluating the curvature of the decision boundary itself is a hard problem. The most common proxy used is the second derivative of the loss function, which hides much information. Using noise-based queries as in section 5.4 may however be a way of understanding the shape of the decision region better, and quantifying the impact that curvature has on the existence of attacks.
- **Representation space.** Another obstacle to our intuitive view of classification is that data are projected in an unknown "concept space", that is itslef learned from data by the network (see chapter 1). This projection may distort the distances between input points, and be one of the causes of the existence of attacks. This can be analyzed by computing the lipschitz constants of each layer in small networks, and seeing how the curvature of the decision region changes with each successive projection.

Understanding the root cause of adversarial examples will then lead to potential solutions, like using noise injection with several types of distributions during training to "model" the local curvature of the decision boundary, or focusing on building locally lipschitz networks.

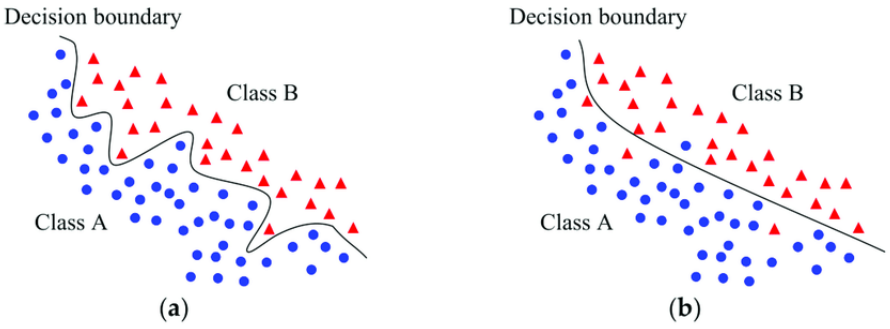


Figure 6.2: Overfitting lead to spikes in the decision region, which are zones especially vulnerable to attacks.

Bibliography

[Alzantot et al., 2018] Alzantot, M., Balaji, B., and Srivastava, M. (2018). Did you hear that? adversarial examples against automatic speech recognition. *arXiv preprint arXiv:1801.00554*.

[Athalye et al., 2018] Athalye, A., Carlini, N., and Wagner, D. (2018). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*.

[Awasthi et al., 2021a] Awasthi, P., Frank, N., Mao, A., Mohri, M., and Zhong, Y. (2021a). Calibration and consistency of adversarial surrogate losses. *Advances in Neural Information Processing Systems*, 34.

[Awasthi et al., 2021b] Awasthi, P., Frank, N. S., and Mohri, M. (2021b). On the existence of the adversarial bayes classifier (extended version). *arXiv preprint arXiv:2112.01694*.

[Awasthi et al., 2021c] Awasthi, P., Mao, A., Mohri, M., and Zhong, Y. (2021c). A finer calibration analysis for adversarial robustness. *arXiv preprint arXiv:2105.01550*.

[Bao et al., 2020] Bao, H., Scott, C., and Sugiyama, M. (2020). Calibrated surrogate losses for adversarially robust classification. In Abernethy, J. and Agarwal, S., editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 408–451. PMLR.

[Bartlett et al., 2006] Bartlett, P. L., Jordan, M. I., and McAuliffe, J. D. (2006). Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156.

[Bartlett and Mendelson, 2002] Bartlett, P. L. and Mendelson, S. (2002). Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482.

[Bhagoji et al., 2019] Bhagoji, A. N., Cullina, D., and Mittal, P. (2019). Lower bounds on adversarial robustness from optimal transport. *Advances in Neural Information Processing Systems*, 32.

[Blumenson, 1960] Blumenson, L. (1960). A derivation of n-dimensional spherical coordinates. *The American Mathematical Monthly*, 67.

[Blumer et al., 1989] Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. (1989). Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965.

[Brown, 1951] Brown, G. W. (1951). Iterative solution of games by fictitious play. *Activity analysis of production and allocation*, 13(1):374–376.

[Bungert et al., 2021] Bungert, L., Trillos, N. G., and Murray, R. (2021). The geometry of adversarial training in binary classification. *arXiv preprint arXiv:2111.13613*.

[Carlini et al., 2019] Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., and Madry, A. (2019). On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*.

[Carlini and Wagner, 2017a] Carlini, N. and Wagner, D. (2017a). 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.

[Carlini and Wagner, 2017b] Carlini, N. and Wagner, D. (2017b). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE.

[Cavonius and Estevez, 1975] Cavonius, C. and Estevez, O. (1975). Contrast sensitivity of individual colour mechanisms of human vision. *The Journal of physiology*, 248(3):649–662.

[Chernoff and Scheffe, 1952] Chernoff, H. and Scheffe, H. (1952). A generalization of the neyman-pearson fundamental lemma. *The Annals of Mathematical Statistics*.

[Cohen et al., 2019] Cohen, J., Rosenfeld, E., and Kolter, Z. (2019). Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning*.

[Dada et al., 2019] Dada, E. G., Bassi, J. S., Chiroma, H., Adetunmbi, A. O., Ajibuwa, O. E., et al. (2019). Machine learning for email spam filtering: review, approaches and open research problems. *Heliyon*, 5(6):e01802.

[Dhillon et al., 2018] Dhillon, G. S., Azizzadenesheli, K., Bernstein, J. D., Kossaifi, J., Khanna, A., Lipton, Z. C., and Anandkumar, A. (2018). Stochastic activation pruning for robust adversarial defense. In *International Conference on Learning Representations*.

[Diochnos et al., 2018] Diochnos, D., Mahloujifar, S., and Mahmoody, M. (2018). Adversarial risk and robustness: General definitions and implications for the uniform distribution. *Advances in Neural Information Processing Systems*, 31.

[Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[Dvijotham et al., 2020] Dvijotham, K. D., Hayes, J., Balle, B., Kolter, Z., Qin, C., Gyorgy, A., Xiao, K., Gowal, S., and Kohli, P. (2020). A framework for robustness certification of smoothed classifiers using f-divergences. In *International Conference on Learning Representations*.

[Fedus et al., 2021] Fedus, W., Zoph, B., and Shazeer, N. (2021). Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity.

[Foret et al., 2020] Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. (2020). Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*.

[Freund and Schapire, 1995] Freund, Y. and Schapire, R. E. (1995). A Decision Theoretic Generalization of On-Line Learning and an Application to Boosting. In Vitányi, P. M. B., editor, *Second European Conference on Computational Learning Theory (EuroCOLT-95)*, pages 23–37.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

[Goodfellow et al., 2015] Goodfellow, I., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

[Gouk et al., 2021] Gouk, H., Frank, E., Pfahringer, B., and Cree, M. J. (2021). Regularisation of neural networks by enforcing lipschitz continuity. *Machine Learning*, 110(2):393–416.

[Gozlan et al., 2018] Gozlan, N., Samson, P.-M., and Zitt, P.-A. (2018). Notes de cours sur le transport optimal.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[Hinton et al., 2012] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97.

[Jeong and Shin, 2020] Jeong, J. and Shin, J. (2020). Consistency regularization for certified robustness of smoothed classifiers. *Advances in Neural Information Processing Systems*, 33:10558–10570.

[Kermack and McKendrick, 1927] Kermack, W. O. and McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, 115(772):700–721.

[Khanal et al., 2020] Khanal, S. S., Prasad, P., Alsadoon, A., and Maag, A. (2020). A systematic review: machine learning based recommendation systems for e-learning. *Education and Information Technologies*, 25(4):2635–2664.

[Krizhevsky and Hinton, 2009] Krizhevsky, A. and Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical report, Citeseer.

[Krizhevsky et al., 2009] Krizhevsky, A., Nair, V., and Hinton, G. (2009). Cifar-10 (canadian institute for advanced research).

[Kumar et al., 2020] Kumar, A., Levine, A., Goldstein, T., and Feizi, S. (2020). Curse of dimensionality on randomized smoothing for certifiable robustness. In *Proceedings of the 37th International Conference on Machine Learning*.

[Kurakin et al., 2018] Kurakin, A., Goodfellow, I. J., and Bengio, S. (2018). Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC.

[Laraki et al., 2019] Laraki, R., Renault, J., and Sorin, S. (2019). *Mathematical foundations of game theory*. Springer.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[Lecuyer et al., 2018] Lecuyer, M., Atlidakais, V., Geambasu, R., Hsu, D., and Jana, S. (2018). Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*.

[Levine et al., 2021] Levine, A., Kumar, A., Goldstein, T., and Feizi, S. (2021). Tight second-order certificates for randomized smoothing.

[Li et al., 2018] Li, B., Chen, C., Wang, W., and Carin, L. (2018). Second-order adversarial attack and certifiable robustness.

[Li et al., 2019] Li, H., Zhou, S., Yuan, W., Li, J., and Leung, H. (2019). Adversarial-example attacks toward android malware detection system. *IEEE Systems Journal*, 14(1):653–656.

[Liu et al., 2016] Liu, Y., Chen, X., Liu, C., and Song, D. (2016). Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*.

[Long and Servedio, 2013] Long, P. and Servedio, R. (2013). Consistency versus realizable h-consistency for multiclass classification. In *International Conference on Machine Learning*, pages 801–809. PMLR.

[Madry et al., 2018] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

[Maschler et al., 2020] Maschler, M., Zamir, S., and Solan, E. (2020). *Game theory*. Cambridge University Press.

[McCulloch and Pitts, 1943] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

[Meunier et al., 2022a] Meunier, L., Delattre, B., Araujo, A., and Allauzen, A. (2022a). A dynamical system perspective for lipschitz neural networks. In *International Conference on Machine Learning*.

[Meunier et al., 2022b] Meunier, L., Ettedgui, R., Pinot, R., Chevaleyre, Y., and Atif, J. (2022b). Towards consistency in adversarial classification. *arXiv preprint arXiv:2205.10022*.

[Meunier et al., 2021] Meunier, L., Scetbon, M., Pinot, R. B., Atif, J., and Chevaleyre, Y. (2021). Mixed nash equilibria in the adversarial examples game. In *International Conference on Machine Learning*, pages 7677–7687. PMLR.

[Mistry et al., 2021] Mistry, D., Litvinova, M., Pastore y Piontti, A., Chinazzi, M., Fumanelli, L., Gomes, M. F., Haque, S. A., Liu, Q.-H., Mu, K., Xiong, X., et al. (2021). Inferring high-resolution human mixing patterns for disease modeling. *Nature communications*, 12(1):1–12.

[Mohapatra et al., 2021] Mohapatra, J., Ko, C.-Y., Weng, L., Chen, P.-Y., Liu, S., and Daniel, L. (2021). Hidden cost of randomized smoothing. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*.

[Mohapatra et al., 2020] Mohapatra, J., Ko, C.-Y., Weng, T.-W., Chen, P.-Y., Liu, S., and Daniel, L. (2020). Higher-order certification for randomized smoothing. In *Advances in Neural Information Processing Systems*.

[Moosavi-Dezfooli et al., 2019] Moosavi-Dezfooli, S.-M., Fawzi, A., Uesato, J., and Frossard, P. (2019). Robustness via curvature regularization, and vice versa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9078–9086.

[Mor-Yosef et al., 1990] Mor-Yosef, S., Samueloff, A., Modan, B., Navot, D., and Schenker, J. G. (1990). Ranking the risk factors for cesarean: logistic regression analysis of a nationwide study. *Obstetrics and gynecology*, 75(6):944–947.

[Morgulis et al., 2019] Morgulis, N., Kreines, A., Mendelowitz, S., and Weisglass, Y. (2019). Fooling a real car with adversarial traffic signs. *arXiv preprint arXiv:1907.00374*.

[Nair and Hinton, 2010] Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Icml*.

[Nakkiran et al., 2021] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2021). Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003.

[Neumann, 1928] Neumann, J. v. (1928). Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320.

[Olver et al., 2010] Olver, F. W., Lozier, D. W., Boisvert, R. F., and Clark, C. W. (2010). *NIST handbook of mathematical functions hardback and CD-ROM*. Cambridge university press.

[Pal and Vidal, 2020] Pal, A. and Vidal, R. (2020). A game theoretic analysis of additive adversarial attacks and defenses. *Advances in Neural Information Processing Systems*, 33:1345–1355.

[Pinot et al., 2020] Pinot, R., Ettedgui, R., Rizk, G., Chevaleyre, Y., and Atif, J. (2020). Randomization matters how to defend against strong adversarial attacks. In *International Conference on Machine Learning*, pages 7717–7727. PMLR.

[Pydi and Jog, 2020a] Pydi, M. S. and Jog, V. (2020a). Adversarial risk via optimal transport and optimal couplings. In *International Conference on Machine Learning*, pages 7814–7823. PMLR.

[Pydi and Jog, 2020b] Pydi, M. S. and Jog, V. (2020b). Adversarial risk via optimal transport and optimal couplings. In *International Conference on Machine Learning*.

[Pydi and Jog, 2021] Pydi, M. S. and Jog, V. (2021). The many faces of adversarial risk. *Advances in Neural Information Processing Systems*, 34.

[Qin et al., 2019] Qin, Y., Carlini, N., Cottrell, G., Goodfellow, I., and Raffel, C. (2019). Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *International conference on machine learning*, pages 5231–5240. PMLR.

[Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

[Salman et al., 2019] Salman, H., Li, J., Razenshteyn, I., Zhang, P., Zhang, H., Bubeck, S., and Yang, G. (2019). Provably robust deep learning via adversarially trained smoothed classifiers. In *Advances in Neural Information Processing Systems*.

[Salman et al., 2020] Salman, H., Sun, M., Yang, G., Kapoor, A., and Kolter, J. Z. (2020). Denoised smoothing: A provable defense for pretrained classifiers. In *Advances in Neural Information Processing Systems*.

[Shafahi et al., 2018] Shafahi, A., Huang, W. R., Studer, C., Feizi, S., and Goldstein, T. (2018). Are adversarial examples inevitable? *International Conference on Learning Representtation*.

[Shafahi et al., 2019] Shafahi, A., Najibi, M., Ghiasi, M. A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. (2019). Adversarial training for free! *Advances in Neural Information Processing Systems*, 32.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[Sitawarin et al., 2018] Sitawarin, C., Bhagoji, A. N., Mosenia, A., Mittal, P., and Chiang, M. (2018). Rogue signs: Deceiving traffic sign recognition with malicious ads and logos. *arXiv preprint arXiv:1801.02780*.

[Steinwart, 2007] Steinwart, I. (2007). How to compare different loss functions and their risks. *Constructive Approximation*, 26(2):225–287.

[Szegedy et al., 2014] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2014). Intriguing properties of neural networks. In *International Conference on Learning Representations*.

[Tramer et al., 2020] Tramer, F., Carlini, N., Brendel, W., and Madry, A. (2020). On adaptive attacks to adversarial example defenses. In *Advances in Neural Information Processing Systems*.

[Tramèr et al., 2017] Tramèr, F., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. (2017). The space of transferable adversarial examples. *arXiv*.

[Van Damme, 1991] Van Damme, E. (1991). *Stability and perfection of Nash equilibria*, volume 339. Springer.

[Vapnik and Chervonenkis, 2015] Vapnik, V. N. and Chervonenkis, A. Y. (2015). On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer.

[Villani, 2009] Villani, C. (2009). *Optimal transport: old and new*, volume 338. Springer.

[Villani, 2021] Villani, C. (2021). *Topics in optimal transportation*, volume 58. American Mathematical Soc.

[Wang et al., 2021] Wang, L., Zhai, R., He, D., Wang, L., and Jian, L. (2021). Pretrain-to-finetune adversarial training via sample-wise randomized smoothing.

[Wang et al., 2022] Wang, W., Wang, L., Wang, R., Ye, A., and Ke, J. (2022). Better constraints of imperceptibility, better adversarial examples in the text. *International Journal of Intelligent Systems*, 37(6):3440–3459.

[Wong and Kolter, 2018] Wong, E. and Kolter, Z. (2018). Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*.

[Xiao et al., 2018] Xiao, C., Li, B., Zhu, J.-Y., He, W., Liu, M., and Song, D. (2018). Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*.

[Yang et al., 2020] Yang, G., Duan, T., Hu, J. E., Salman, H., Razenshteyn, I., and Li, J. (2020). Randomized smoothing of all shapes and sizes. In *Proceedings of the 37th International Conference on Machine Learning*.

[Yang et al., 2021] Yang, L., Song, Q., and Wu, Y. (2021). Attacks on state-of-the-art face recognition using attentional adversarial attack generative network. *Multimedia tools and applications*, 80(1):855–875.

[Zagoruyko and Komodakis, 2016] Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press.

[Zhai et al., 2020] Zhai, R., Dan, C., He, D., Zhang, H., Gong, B., Ravikumar, P., Hsieh, C.-J., and Wang, L. (2020). Macer: Attack-free and scalable robust training via maximizing certified radius. In *International Conference on Learning Representations*.

[Zhang et al., 2018] Zhang, L., Wang, S., and Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.

[Zhang, 2004] Zhang, T. (2004). Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32(1):56–85.

[Zhen et al., 2021] Zhen, X., Chakraborty, R., and Singh, V. (2021). Simpler certified radius maximization by propagating covariances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7292–7301.

[Zhong and Deng, 2020] Zhong, Y. and Deng, W. (2020). Towards transferable adversarial attack against deep face recognition. *IEEE Transactions on Information Forensics and Security*, 16:1452–1466.

Appendices

A General study of the equilibrium under randomized attacks

This chapter is a collaboration with Prof. Guillaume Carlier, from the Ceremade Laboratory, who did the demonstrations of the theorems.

As we have shown the non-existence of Nash equilibrium in the deterministic regime, a natural way of looking for optimal attacks and defenses is to study randomization, i.e. a convex relaxation of the problem. This is also an extension from the work of [Pydi and Jog, 2020b], which through Strassen's lemma implicitly model attacks as Monge transport maps. In this section, we will study the case of randomized attacks as transport plans.

A.1 Problem statement : transport plans as randomized attacks

Recall the Attacker's problem we are looking for two distributions $\phi_y \# \mu_y$ that maximize the expected risk, ponderated by some lower-semicontinuous transport cost c. This is very similar to the Monge formulation of the optimal transport problem : transporting the distribution amounts to finding a determinstc coupling $(\mu_y, \phi_y \# \mu_y)$, i.e. we cannot split the mass that is allocated at a point when we displace it.

As we saw in Section 2.5.2, the natural relaxation of transport maps is to allow the Attacker to play a *coupling* between μ_y and another measure.

Definition 63 (Mixed strategy for the Attacker)**.** *A mixed strategy for the Attacker is a pair of probability measures $\gamma_i \in \mathcal{P}(\mathcal{X} \times \mathcal{X})$, such that $proj_1 \# \gamma_i = \mu_i$, for $i = \pm 1$. The set of all admissible strategies for the Attacker is thus :*

$$\Delta_{\mu_1, \mu_{-1}} = \left\{ \gamma_i \in \mathcal{P}(\mathcal{X} \times \mathcal{X}) : proj_1 \# \gamma_i = \mu_i, i = \pm 1 \right\}$$

As we have seen in Section 2.5.2, this is equivalent to the Attacker playing a randomized attack $p_x \in \mathcal{P}(\mathcal{X})$ at every point, distributing the mass on some support instead of

translating it all to a unique point. Let $l : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}_+$ denote the loss function used for classification, and $L_i = L(\cdot, i)$ for $i = \pm 1$. We define the mixed zero-sum game as follows:

Definition 64 (Mixed-Attacker zero-sum game)**.** *For any cost function c_i, we define the payoff of the zero-sum game as:*

$$R(h, \gamma_1, \gamma_{-1}) := \sum_{i=\pm 1} \int_{\mathcal{X}\times\mathcal{X}} [L_i(h(z)) - c_i(x, z)] d\gamma_i(x, z)$$

Remark 4. *Note that in this section, we incorporated the probabilities q_i inside of μ_i for more lisibility. The μ_i are therefore not necessarily probability measures.*

We thus have the two following problems :
Defender problem :

$$\overline{v} = \inf_{h\in\mathcal{C}(\mathcal{X})} \sup_{(\gamma_1,\gamma_{-1})\in\Delta_{\mu_1,\mu_{-1}}} R(h, \gamma_1, \gamma_{-1}). \tag{A.1}$$

Attacker problem :

$$\underline{v} = \sup_{(\gamma_1,\gamma_{-1})\in\Delta_{\mu_1,\mu_{-1}}} \inf_{h\in\mathcal{C}(\mathcal{X})} R(h, \gamma_1, \gamma_{-1}). \tag{A.2}$$

A.2 Duality result, existence of a mixed nash equilibrium

Let us first reformulate both problems in a way that is easier to analyze.

Hypothesis 1. *In that section, we will make the following hypothesis :*

- $L_i : \mathbb{R} \to \mathbb{R}_+$ *are convex (and so continuous) for* $i = \pm 1$.
- $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \cup \{\infty\}$ *is lower semicontinuous;*
- $\forall x \in \mathcal{X}, c(x, x) = 0$

Proposition 7. *For a fixed classifier $h \in \mathcal{C}(\mathcal{X})$, the set of the Attacker's best response always contains a deterministic attack. Furthermore,*

$$\overline{v} = \inf_{h \in \mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} \max_{z \in \mathcal{X}} [L_i(h(z)) - c_i(x,z)] d\mu_i(x) \tag{A.3}$$

Proof. As c is lower semicontinuous, so is the function $z \mapsto L_i(h(z)) - c(x,z)$ for any $x \in \mathcal{X}$. As $\mathcal{X}$ is compact, this function attains its maximum over $\mathcal{X}$, and $\arg\max_{z \in \mathcal{X}} \{L_i(h(z)) - c_i(x,z)\}$ is non-empty for every $x \in \mathcal{X}$.

It follows that $\gamma_i^* = (\mathrm{Id}, S_i)\#\mu_i$ with $S_i(x) \in \arg\max_{z \in \mathcal{X}} \{L_i(h(z)) - c_i(x,z)\}$ is well defined, and is a maximizer of $R(h, \dot{,}\dot{)}$ over $\Delta_{\mu_1,\mu_{-1}}$. Hence :

$$\max_{(\gamma_1,\gamma_{-1} \in \Delta_{\mu_1,\mu_{-1}}} R(h, \gamma_1, \gamma_{-1}) = \sum_{i=\pm 1} \int_{\mathcal{X}} \max_{z \in \mathcal{X}} [L_i(h(z)) - c_i(x,z)] d\mu_i(x)$$

The result immediately follows by taking the infimum in $h \in \mathcal{C}(h)$ □

Let us now simplify the Defender's problem :

Definition 65 (Transported distribution)**.** *We define the transported distributions ν_i ($i = \pm 1$) as the second marginal of the coupling γ_i :*

$$\nu_i = proj_2 \# \gamma_i$$

Proposition 8. *We can reformulate $\underline{v}$ using the transported distributions :*

$$\underline{v} = \sup_{(\nu_1,\nu_{-1}) \in \mathcal{P}(\mathcal{X})} \left[\inf_{h \in \mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} L_i \circ h d\nu_i - \sum_{i=\pm 1} T_c(\mu_i, \nu_i) \right] \tag{A.4}$$

Where $T_c(\mu_i, \nu_i)$ denotes the value of the Monge-Kantorovich problem (see Section 2.5.2):

$$T_c(\mu_i, \nu_i) = \min_{\gamma \in \Pi(\mu_i,\nu_i)} \int_{\mathcal{X} \times \mathcal{X}} c_i(x,z) d\gamma(x,z)$$

Proof.

$$
\begin{aligned}
\underline{\nu} &= \sup_{(\gamma_1,\gamma_{-1})\in\Delta_{\mu_1,\mu_{-1}}} \inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}\times\mathcal{X}} [L_i(h(z)) - c_i(x,z)] d\gamma_i(x,z) \\
&= \sup_{(\gamma_1,\gamma_{-1})\in\Delta_{\mu_1,\mu_{-1}}} \left[\inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} L_i(h(z)) d\nu_i(z) - \int_{\mathcal{X}\times\mathcal{X}} c_i(x,z) d\gamma_i(x,z) \right] \\
&= \sup_{(\nu_1,\nu_{-1})\in\mathcal{P}(\mathcal{X})} \sup_{\gamma\in\Pi(\mu_i,\nu_i)} \left[\inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} L_i(h(z)) d\nu_i(z) - \int_{\mathcal{X}\times\mathcal{X}} c_i(x,z) d\gamma_i(x,z) \right] \\
&= \sup_{(\nu_1,\nu_{-1})\in\mathcal{P}(\mathcal{X})} \left[\inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} L_i(h(z)) d\nu_i(z) - \inf_{\gamma\in\Pi(\mu_i,\nu_i)} \int_{\mathcal{X}\times\mathcal{X}} c_i(x,z) d\gamma_i(x,z) \right] \\
&= \sup_{(\nu_1,\nu_{-1})\in\mathcal{P}(\mathcal{X})} \left[\inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} L_i \circ h d\nu_i - \sum_{i=\pm 1} T_c(\mu_i,\nu_i) \right]
\end{aligned}
$$

Hence the result. □

We have thus divided the sup-inf into two portions : one that depends on the loss function, but not the coupling itself, and one that only depends on the coupling and the transport cost, but not the classifier or the loss function. We can further simplify the first part noticing that for the distributions ν_i, the only important thing for the classifier is their relative importance, i.e. how much more probable one class is from the other. In other words, we will express their densities relative to some dominant distribution, in that case $\nu_1 + \nu_{-1}$.

Lemma 12 ([Bartlett et al., 2006], page 141)**.** *Let $(\nu_1, \nu_{-1}) \in \mathcal{P}(\mathcal{X})$, and $\overline{\nu} = \nu_1 + \nu_{-1}$. Let α_i denote the density of ν_i with respect to $\overline{\nu}$ (so that $\alpha_1 + \alpha_{-1} = 1$ $\overline{\nu}$-almost everywhere. Then we have :*

$$\inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} (L_i \circ h) d\nu_i = \int_{\mathcal{X}} H(\alpha_1(z)) d\overline{\nu}(z) \tag{A.5}$$

Where

$$\forall \alpha \in [0,1], H(\alpha) := \inf_{t\in\mathbb{R}} \{\alpha L_1(t) + (1-\alpha) L_{-1}(t)\} \tag{A.6}$$

H is called the optimal conditional $L-$risk.

Using the previous lemmas, we can now rewrite :

$$\underline{v} = \sup_{(\nu_1,\nu_{-1})\in\mathcal{P}(\mathcal{X})} \left[\int_{\mathcal{X}} H(\frac{d\nu_1}{d\overline{\nu}}) d\overline{\nu} - \sum_{i=\pm 1} T_c(\mu_i, \nu_i) \right]$$

Theorem 23 (Existence of an asymptotic Nash equilibrium)**.** *Under Hypothesis 1, one has:*

$$\overline{v} = \underline{v} = \max_{(\gamma_1,\gamma_{-1})\in\Delta_{\mu_1,\mu_{-1}}} \inf_{h\in\mathcal{C}(\mathcal{X})} R(h, \gamma_1, \gamma_{-1}).$$

Proof. We have, from Equation (A.3):

$$\overline{v} = \inf_{h\in\mathcal{C}(\mathcal{X})} \sum_{i=\pm 1} \int_{\mathcal{X}} \max_{z\in\mathcal{X}} [L_i(h(z)) - c_i(x,z)] d\mu_i(x)$$

Seeing that $\max_{z\in\mathcal{X}}[L_i(h(z)) - c_i(x,z)]$ is a function of x that is always greater than $L_i(h(z)) - c_i(x,z)$, this expression can be rewritten equivalently as :

$$\inf_{(f_1,f_{-1},h)\in\mathcal{C}(\mathcal{X})^3} \left\{ \sum_{i=\pm 1} \int_{\mathcal{X}} f_i d\mu_i | \forall (x,z) \in \mathcal{X}^2, f_i(x) \geq L_i(h(z)) - c_i(x,z) \right\}$$

Let us now define the following operators to write that problem in Fenchel-Rockafeller form:

- Let F be the linear operator

$$\forall (f_1, f_{-1}, h) \in \mathcal{C}(\mathcal{X})^3, F(f_1, f_{-1}, h) := \sum_{i=\pm 1} f_i d\mu_i$$

- Λ is the linear continuous operator with values in $\mathcal{C}(\mathcal{X}^2) \times \mathcal{C}(\mathcal{X}^2) \times \mathcal{C}(\mathcal{X})$:

$$\forall (f_1, f_{-1}, h) \in \mathcal{C}(\mathcal{X})^3, \Lambda(f_1, f_{-1}, h) := (f_1 \circ \text{proj}_1, f_{-1} \circ \text{proj}_1, h)$$

- and G is the function such that, for $(g_1, g_{-1}, h) \in \mathcal{C}(\mathcal{X}^2) \times \mathcal{C}(\mathcal{X}^2) \times \mathcal{C}(\mathcal{X})$:

$$G(g_1, g_{-1}, h) := \begin{cases} 0 \text{ if } \forall (x,z) \in \mathcal{X}^2, \psi_i(x,z) \geq L_i(h(z)) - c_i(x,z) \\ +\infty \text{ otherwise.} \end{cases}$$

We thus have :

$$\overline{v} = \inf_{(f_1,f_{-1},h)\in\mathcal{C}(\mathcal{X})^3} F(f_1,f_{-1},h) + G(\Lambda(f_1,f_{-1},h))$$

By applying the Fenchel-Rockafeller theorem (Theorem 5), we get :

$$\overline{v} = \max_{(\gamma_1,\gamma_{-1},\nu)\in\mathcal{M}(\mathcal{X}^2)^2\times\mathcal{M}(\mathcal{X})} -F^*(\Lambda^*(\gamma_1,\gamma_{-1},\nu)) - G^*(-\gamma_1,-\gamma_{-1},-\nu)$$

Where the ajoints can be directly computed :

$$\forall(\gamma_1,\gamma_{-1},\nu)\in\mathcal{M}(\mathcal{X}^2)^2\times\mathcal{M}, \Lambda^*(\gamma_1,\gamma_{-1},\nu) = (\text{proj}_1\#\gamma_1, \text{proj}_1\#\gamma_{-1},\nu)$$

hence

$$F^*(\Lambda^*((\gamma_1,\gamma_{-1},\nu)) = \begin{cases} 0 \text{ if } \text{proj}_1\#\gamma_1 = \mu_1, \text{proj}_1\#\gamma_{-1} = \mu_{-1} \text{ and } \nu = 0 \\ +\infty \text{ otherwise.} \end{cases}$$

And finally:

$$-G^*(-\gamma_1,-\gamma_{-1},0) = \begin{cases} \inf_{h\in\mathcal{C}(\mathcal{X})} R(h,\gamma_1,\gamma_{-1}) \text{ if } \gamma_i \geq 0 \\ -\infty \text{ otherwise.} \end{cases}$$

Replacing, it follows that :

$$\overline{v} = \max_{(\gamma_1,\gamma_{-1})\in\Delta_{\mu_1,\mu_{-1}}} \inf_{h\in\mathcal{C}(\mathcal{X})} R(h,\gamma_1,\gamma_{-1}) = \underline{v}$$

□

A.3 Existence of a optimal clssifier

We now know, from Theorem 23, that under Hypothesis 1 there exists a sequence of classifiers whose risk under attack converge to the optimum. We know want to study under which conditions an optimal solution can be exactly computed.

Hypothesis 2. *On top of Hypothesis 1 we assume :*

- c_1 *and* c_{-1} *are continuous on* $\mathcal{X} \times \mathcal{X}$*;*
- L_1 *is nonincreasing and* L_{-1} *is nondecreasing;*
- $\lim_{t\to-\infty} L_1(t) = \lim_{t\to\infty} L_{-1}(t) = +\infty$.

Theorem 24 (Existence of a continuous solution)**.** *Under Hypothesis 2, there exists an optimal solution to Equation* (A.3)*, i.e. the Defender has an continuous optimal strategy.*

Proof. Let $(h^n)_n \in \mathcal{C}(\mathcal{X})^{\mathbb{N}}$ be a minimizing sequence for Equation (A.3), i.e. such that

$$\lim_{n\to\infty} \sum_{i=\pm1} \int_{\mathcal{X}} f_i^n(x) d\mu_i(x) = \overline{v} \tag{A.7}$$

where

$$\forall i = \pm1, x \in \mathcal{X}, f_i^n(x) = \max_{z\in\mathcal{X}} \{L_i(h^n(z)) - c_i(x,z)\}$$

As c_1 and c_{-1} are continuous, the modulus

$$\omega(t) = \max_{i=\pm1} \max\big\{|c_i(x_1,z_1) - c_i(x_2,z_2)|, (x_1,x_2,z_1,z_2) \in \mathcal{X}^4, d(x_1,x_2) + d(z_1,z_2) \le t\big\}$$

satisfies $\omega(t) \to 0$ as $t \to 0$. Hence, for every $n \in \mathbb{N}$, $i = \pm1$ and every $(x_1, x_2) \in \mathcal{X}^2$, we have:

$$\begin{aligned}
|f_i^n(x_1) - f_i^n(x_2)| &= |\max_{z_1\in\mathcal{X}} \{L_i(h^n(z_1)) - c_i(x_1,z_1)\} - \max_{z_2\in\mathcal{X}} \{L_i(h^n(z_2)) - c_i(x_2,z_2)\}| \\
&= \max_{z_1\in\mathcal{X}} \{L_i(h^n(z_1)) - c_i(x_1,z_1)\} - L_i(h^n(z_2)) - c_i(x_2,z_2)| \\
&\quad \text{where } z_2 \text{ attains the maximum in } f_i^n(x') \\
&= |L_i(h^n(z_2)) - c_i(x_1,z_2) - L_i(h^n(z_2)) - c_i(x_2,z_2)| \\
&\le \omega(d(x_1,x_2))
\end{aligned}$$

As ω does not depend in n, this means that both sequences $(f_i^n)_n$ are uniformly equicontinuous. Furthermore, since $L_i \ge 0$ and $c(x,x) = 0$, we know that $f_i^n \ge 0$. From Equation (A.3), we know that $\min_x f_i^n(x)$ is bounded by $\overline{v}$. Due to the equicontinuity,

we can conclude that $(f_i^n)_n$ is uniformly bounded. By Ascoli's theorem, this means that, up to a subsequence, f_i^n converges uniformly to some function f_i, and we have :

$$\overline{v} = \sum_{i=\pm 1} \int_{\mathcal{X}} f_i(x) d\mu_i(x) \tag{A.8}$$

We now need a way to reconstruct the function h from f. For that, we define a "dual" function to f_i : for $i = \pm 1$, any n and $z \in \mathcal{X}$, let

$$g_i^n(z) := \min_{x \in \mathcal{X}} \{f_i^n(x) + c_i(x, z)\}$$

The uniform convergence of f_i^n implies the uniform convergence of g_i^n to the function g_i defined by :

$$g_i(z) := \min_{x \in \mathcal{X}} \{f_i(x) + c_i(x, z)\} \tag{A.9}$$

By construction, we have :

$$g_1^n \geq L_1 \circ h^n, g_{-1}^n \geq L_1 \circ h^n \tag{A.10}$$

and since $\lim_{t \to -\infty} L_1(t) = \lim_{t \to \infty} L_{-1}(t) = +\infty$, this means that h^n must be uniformly bounded : there exists $C > 0$ such that $-C \leq h^n \leq C$ for all n.

We will now use the pseudo-inverse of the convex losses L_i. For $t > L_1(C)$, we define :

$$L_1^{-1}(t) := \inf\{u \in (-\infty, C], L_1(u) \leq t\} \tag{A.11}$$

Recall that L_1 is both nonincreasing and convex, so depending on whether its infimum is attained or not, L_1 is either decreasing on $\mathbb{R}$ or decreasing on some interval $(-\infty, a]$ then constant on $[a, +\infty)$. In any case, $t \geq L_1(u) \iff L_1^{-1}(t) \geq u$ and L_1^{-1} is nonincreasing continuous. Similarly, for $t \geq L_{-1}(-C)$, we define :

$$L_{-1}^{-1}(t) := \sup\{u \in [C, +\infty), L_{-1}(u) \leq t\} \tag{A.12}$$

L_{-1}^{-1} is nondecreasing continuous, and for $t \geq -C, t \geq L_{-1}(u) \iff L_{-1}^{-1}(t) \geq u$. We can thus rewrite equation Equation (A.10) as :

$$L_{-1}^{-1}(g_{-1}^n) \leq h^n \leq L_1^{-1}(g_1^n) \tag{A.13}$$

which by continuity passes to the limit to :

$$\overline{h} = L_{-1}^{-1}(g_{-1}) \geq \underline{h} = L_1^{-1}(g_1) \tag{A.14}$$

For $i = \pm 1$ and any $h \in \mathcal{C}(\mathcal{X})$, we have $g_i \geq L_i \circ h$.

Recall that $g_i(z) = \min_{x \in \mathcal{X}} \{f_i(x) + c_i(x, z)\}$. This means that we have, for all $x \in \mathcal{X}$,

$$\phi_i(x) \geq \max_{z \in \mathcal{X}} \{L_i(h(z)) - c_i(x, z)\}$$

It follows that :

$$\sum_{i=\pm 1} \int_{\mathcal{X}} \max_{z \in \mathcal{X}} [L_i(h(z)) - c_i(x, z)] d\mu_i(x) \leq \underline{v}$$

which is the infimum. Hence, h solves Equation (A.3). □

A.4 Existence of Lipschitz solutions

We have now shown the conditions for the existence of a continuous optimal solution. We would expect this solution to exhibit more powerful forms of continuity, as robustness naturally encourages controlled local variations. We will now investigate the conditions under which a Lipschitz solution can exist.

Proposition 9. *If, in addition to Hypothesis 2, we also have that :*

- *the transport costs c_i are Lipschitz on $\mathcal{X} \times \mathcal{X}$*
- *either L_1 is decreasing or L_{-1} is increasing*

Then there exists an optimal Lispschitz classifier.

Proof. Following the proof of Theorem 24, we find g_1 and g_{-1} as in Equation (A.9) (in the optimal transport terminology, we say that g_i is c_i-quasi-concave) such that $L_1^{-1}(g_1) \leq L_{-1}^{-1}(g_{-1})$ and any $h \in \mathcal{C}(\mathcal{X})$ such that $L_1^{-1}(g_1) \leq h \leq L_{-1}^{-1}(g_{-1})$ solves Equation (A.3).

Since the g_i are of the form Equation (A.9), the fact that the c_i are Lipschitz means that the g_i are as well. As L_1 is decreasing and convex, its subgradient is bounded away from zero on any compact interval, so its inverse is Lipschitz on compact sets, including $\mathcal{X}$, and so is $L_1^{-1}(g_1)$, which is thus a Lipschitz solution. The same reasoning gives $L_{-1}^{-1}(g_{-1})$ as a Lipschitz solution when L_{-1} is decreasing. □

A.5 Discussion on realistic transport costs and transport plans.

The general hypothesis of Theorem 23 encompass most realistic transport costs, in particular imperceptibiliy-enforcing costs, of the form $c(x,z) = \begin{cases} k(x,z) & \text{if } \|x-z\| \leq \epsilon \\ +\infty & \text{otherwise} \end{cases}$.

Allowing the use of randomized attacks, when using surrogate loss functions, thus create asymptotic Nash equilibria in realistic scenarios. Hypothesis 2 is however more constraining, as it does not hold with the imperceptibility constraint.

On imperceptibility this however leads to the question : is a strict imperceptibility condition really realistic ? Does the human perception work by thresholds, or in a more gradual way (in which case the relaxations of the imperceptibility constraints, such as the Carlini&Wagner cost, would be a better modeling hypothesis).

Most relaxations of the imperceptibility constraint lead to continuous, and even Lipschitz costs, which ensure the existence of an optimal Lipschitz solution for the Defender.

On transport plans for attacks However, let us keep in mind that the Attacker described in this section is very powerful. We allowed any randomized strategy, while transport plans for continuous distributions are in practice very difficult, if not impossible to compute. More research should be conduced to identify the realistic hypothesis for the Attacker, and on how to compute randomized attacks.

To summarize our results in this section, for surrogate loss functions and relaxed imperceptibility constraints, when the Attacker is allowed to play general randomized strategies, there is an optimal continuous, and even Lipschitz classifier.

B Towards consistency in adversarial classification

This chapter is the result of a work carried out in collaboration with Laurent Meunier, published at Neurips 2022, under the name "Towards consistency in adversarial classification". We refer the reader to the arxiv version of the paper for more details.

Contents

As we mentioned in chapter 2, a fundamental aspect of classification is the choice of the loss function. As the $0/1$ loss is not convex, it is unusable in practical implementations, so that convex surrogates are used in every neural network to this day. In the standard setting, this does not incur any problem, as a wide class of losses exhibit the property of *consistency*, i.e. minimizing them amounts to minimizing the $0/1$. This is possible because consistency can be reduced to a pointwise minimization property called *calibration*. However, this is not possible anymore in the adversarial setting, as the problem is by nature non-pointwise. Hence the question :

Can surrogate losses still be used as a proxy for minimizing the $0/1$ *loss in the presence of an adversary that alters the inputs at test-time?*

Different from the standard classification task, this cannot be reduced to a point-wise minimization problem, and calibration needs not to be sufficient to ensure consistency. In this paper, we expose some pathological behaviors specific to the adversarial problem, and show that no convex surrogate loss can be consistent or calibrated in this context. It is therefore necessary to design another class of surrogate functions that can be used to

solve the adversarial consistency issue. As a first step towards designing such a class, we identify sufficient and necessary conditions for a surrogate loss to be calibrated in both the adversarial and standard settings. Finally, we give some directions for building a class of losses that could be consistent in the adversarial framework.

B.1 Notions of Calibration and Consistency

Let us consider a classification task with input space $\mathcal{X}$ and output space $\mathcal{Y} = \{-1, +1\}$. Let $(\mathcal{X}, d)$ be a proper Polish (i.e. completely separable) metric space representing the inputs space. For all $x \in \mathcal{X}$ and $\delta > 0$, we denote $B_\delta(x)$ the closed ball of radius δ and center x. We also assume that for all $x \in \mathcal{X}$ and $\delta > 0$, $B_\delta(x)$ contains at least two points[1]. Let us also endow $\mathcal{Y}$ with the trivial metric $d'(y, y') = \mathbf{1}_{y \neq y'}$. Then the space $(\mathcal{X} \times \mathcal{Y}, d \oplus d')$ is a proper Polish space. For any Polish space $\mathcal{Z}$, we denote $\mathcal{M}_+^1(\mathcal{Z})$ the Polish space of Borel probability measures on $\mathcal{Z}$. We will denote $\mathcal{F}(\mathcal{Z})$ the space of real valued Borel measurable functions on $\mathcal{Z}$. Finally, we denote $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty, +\infty\}$.

B.1.1 Notations and Preliminaries

The $0/1$-loss is both non-continuous and non-convex, and its direct minimization is a difficult problem. The concepts of calibration and consistency aim at identifying the properties that a loss must satisfy in order to be a good surrogate for the minimization of the $0/1$-loss. In this section, we define these two concepts and explain the difference between them. First of all, we need to give a general definition of a loss function.

Definition 66 (Loss function)**.** *A loss function is a function $L : \mathcal{X} \times \mathcal{Y} \times \mathcal{F}(\mathcal{X}) \to \mathbb{R}$ such that $L(\cdot, \cdot, f)$ is measurable for all $f \in \mathcal{F}(\mathcal{X})$.*

Note that this definition is not specific to the standard or adversarial case. In general, the loss at point (x, y) can either depend only on $f(x)$, or on other points related to x (e.g. the set of points within a distance ε of x). We now recall the definition of the risk associated with a loss L and a distribution $\mathbb{P}$.

[1] For instance, for any norm $\|\cdot\|$, $(\mathbb{R}^d, \|\cdot\|)$ is a Polish metric space satisfying this property.

Definition 67 (L-risk of a classifier)**.** *For a given loss function L, and a Borel probability distribution $\mathbb{P}$ over $\mathcal{X} \times \mathcal{Y}$ we define the risk of a classifier f associated with the loss L and a distribution $\mathbb{P}$ as*

$$\mathcal{R}_{L,\mathbb{P}}(f) := \mathbb{E}_{(x,y)\sim\mathbb{P}}[L(x, y, f)].$$

We also define the optimal risk associated with the loss L as

$$\mathcal{R}^{\star}_{L,\mathbb{P}} := \inf_{f\in\mathcal{F}(\mathcal{X})} \mathcal{R}_{L,\mathbb{P}}(f)$$

Essentially, the risk of a classifier is defined as the average loss over the distribution $\mathbb{P}$. When the loss L is difficult to optimize in practice (e.g when it is non-convex or non-differentiable), it is often preferred to optimize a surrogate loss function instead. In the literature [Zhang, 2004, Bartlett et al., 2006, Steinwart, 2007], the notion of surrogate losses has been studied as a consistency problem. In a nutshell, a surrogate loss is said to be consistent if any minimizing sequence of classifiers for the risk associated with the surrogate loss is also one for the risk associated with L. Formally, the notion of consistency is as follows.

Definition 68 (Consistency)**.** *Let L_1 and L_2 be two loss functions. For a given $\mathbb{P} \in \mathcal{M}_1^+(\mathcal{X} \times \mathcal{Y})$, L_2 is said to be consistent for $\mathbb{P}$ with respect to L_1 if for all sequences $(f_n)_n \in \mathcal{F}(\mathcal{X})^{\mathbb{N}}$:*

$$\mathcal{R}_{L_2,\mathbb{P}}(f_n) \to \mathcal{R}^{\star}_{L_2,\mathbb{P}} \implies \mathcal{R}_{L_1,\mathbb{P}}(f_n) \to \mathcal{R}^{\star}_{L_1,\mathbb{P}} \tag{B.1}$$

Furthermore, L_2 is said consistent with respect to a loss L_1 the above holds for any distribution $\mathbb{P}$.

Consistency is in general a difficult problem to study because of its high dependency on the distribution $\mathbb{P}$ at hand. Accordingly, several previous works [Zhang, 2004, Bartlett and Mendelson, 2002, Steinwart, 2007] introduced a weaker notion to study a pointwise version consistency. This simplified notion is called *calibration* and corresponds to consistency when $\mathbb{P}$ is a combination of Dirac distributions. The main building block in the analysis of the calibration problem is the calibration function, defined as follows.

Definition 69 (Calibration function)**.** *Let L be a loss function. The calibration function $\mathcal{C}_L$ is*

$$\mathcal{C}_L(x, \eta, f) := \eta L(x, 1, f) + (1 - \eta) L(x, -1, f),$$

for any $\eta \in [0, 1]$, $x \in \mathcal{X}$ and $f \in \mathcal{F}(\mathcal{X})$. We also define the optimal calibration function as

$$\mathcal{C}_L^\star(x, \eta) := \inf_{f \in \mathcal{F}(\mathcal{X})} \mathcal{C}_L(x, \eta, f).$$

Note that for any $x \in \mathcal{X}$ and $\eta \in [0, 1]$, $\mathcal{C}_L(x, \eta, f) = \mathcal{R}_{L,\mathbb{P}}(f)$ with $\mathbb{P} = \eta\delta_{(x,+1)} + (1 - \eta)\delta_{(x,-1)}$. The calibration function thus corresponds then to a pointwise notion of the risk, evaluated at point x. η corresponds in this case to the conditional probability of $y = 1$ given x. We now define the calibration property of a surrogate loss.

Definition 70 (Calibration)**.** *Let L_1 and L_2 be two loss functions. We say that L_2 is* calibrated *with regards to L_1 if for every $\xi > 0$, $\eta \in [0, 1]$ and $x \in \mathcal{X}$, there exists $\delta > 0$ such that for all $f \in \mathcal{F}(\mathcal{X})$,*

$$\mathcal{C}_{L_2}(x, \eta, f) - \mathcal{C}_{L_2}^\star(x, \eta) \leq \delta \implies \mathcal{C}_{L_1}(x, \eta, f) - \mathcal{C}_{L_1}^\star(x, \eta) \leq \xi.$$

Furthermore, we say that L_2 is uniformly calibrated *with regards to L_1 if for every $\xi > 0$, there exists $\delta > 0$ such that for all $\eta \in [0, 1]$, $x \in \mathcal{X}$ and $f \in \mathcal{F}(\mathcal{X})$ we have*

$$\mathcal{C}_{L_2}(x, \eta, f) - \mathcal{C}_{L_2}^\star(x, \eta) \leq \delta \implies \mathcal{C}_{L_1}(x, \eta, f) - \mathcal{C}_{L_1}^\star(x, \eta) \leq \xi.$$

Connection between calibration and consistency. It is always true that calibration is a necessary condition for consistency. Yet there is no reason, in general, for the converse to be true. However, in the specific context usually studied in the literature (i.e., the standard classification with a well-defined $0/1$-loss), the notions of consistency and calibration have been shown to be equivalent. [Zhang, 2004, Bartlett et al., 2006, Steinwart, 2007]. In the next section, we come back on existing results regarding calibration and consistency in this specific (standard) classification setting.

B.1.2 Existing Results in the Standard Classification Setting

Classification is a standard task in machine learning that consists in finding a classification function $h : \mathcal{X} \to \mathcal{Y}$ that maps an input x to a label y. In binary classification, h is

often defined as the sign of a real valued function $f \in \mathcal{F}(\mathcal{X})$. The loss usually used to characterize classification tasks corresponds to the accuracy of the classifier h. When h is defined as above, this loss is defined as follows.

Definition 71 ($0/1$ loss). *Let $f \in \mathcal{F}(\mathcal{X})$. We define the* $0/1$ loss *as follows*

$$l_{0/1}(x, y, f) = \mathbf{1}_{y \times sign(f(x)) \leq 0}$$

with a convention for the sign, e.g. $sign(0) = 1$. We will denote $\mathcal{R}_{\mathbb{P}}(f) := \mathcal{R}_{l_{0/1},\mathbb{P}}(f)$, $\mathcal{R}^{\star}_{\mathbb{P}} := \mathcal{R}^{\star}_{l_{0/1},\mathbb{P}}$, $\mathcal{C}(x, \eta, f) := \mathcal{C}_{l_{0/1}}(x, \eta, f)$ and $\mathcal{C}^{\star}(x, \eta) := \mathcal{C}^{\star}_{l_{0/1}}(x, \eta)$.

Note that this $0/1$-loss is different from the one introduced by [Bao et al., 2020, Awasthi et al., 2021a, Awasthi et al., 2021c]: they used $\mathbf{1}_{y \times f(x) \leq 0}$ which is a usual $0/1$ loss but unadapted to consistency and calibrated study (see Section B.3 for details). Some of the most prominent works [Zhang, 2004, Bartlett et al., 2006, Steinwart, 2007] among them focus on the concept of margin losses, as defined below.

Definition 72 (Margin loss). *A loss L_{ϕ} is said to be a* margin loss *if there exists a measurable function $\phi : \mathbb{R} \to \mathbb{R}_+$ such that:*

$$L_{\phi}(x, y, f) = \phi(yf(x))$$

For simplicity, we will say that ϕ is a margin loss function and we will denote $\mathcal{R}_{\phi}$ and $\mathcal{C}_{\phi}$ the risk associated with the margin loss ϕ. Notably, it has been demonstrated in several previous works [Zhang, 2004, Bartlett et al., 2006, Steinwart, 2007] that, for a margin loss ϕ, we have always have $\mathcal{C}^{\star}_{\phi}(x, \eta) = \inf_{\alpha \in \mathbb{R}} \eta\phi(\alpha) + (1 - \eta)\phi(-\alpha)$. This is in particular one of the main observation allowing to show the following strong result about the connection between consistency and calibration.

Theorem 25 ([Zhang, 2004, Bartlett et al., 2006, Steinwart, 2007]). *Let $\phi : \mathbb{R} \to \mathbb{R}_+$ be a continuous margin loss. Then the three following assertions are equivalent: (i) ϕ is calibrated with regards to $l_{0/1}$, (ii) ϕ is uniformly calibrated $l_{0/1}$, (iii) ϕ is consistent with regards to $l_{0/1}$.*
Moreover, if ϕ is convex and differentiable at 0, then ϕ is calibrated if and only $\phi'(0) < 0$.

The Hinge loss $\phi(t) = \max(1 - t, 0)$ and the logistic loss $\phi(t) = \log(1 + e^{-t})$ are classical examples of convex consistent losses. Convexity is a desirable property for faster

optimization of the loss, but there exist other non-convex losses that are calibrated as the ramp loss ($\phi(t) = \max(1-t,0)+\max(1+t,0)$) or the sigmoid loss ($\phi(t) = (1+e^t)^{-1}$). In the next section, we present the adversarial classification setting for which Theorem 25 may not hold anymore.

Remark 5. *The equivalence between calibration and consistency is a consequence from the fact that, over the large space of measurable functions, minimizing the loss pointwisely in the input by desintegrating with regards to x is equivalent to minimize the whole risk over measurable functions. This result is very powerful and simplify the study of calibration in the standard setting.*

B.1.3 Calibration and Consistency in the Adversarial Setting.

We now consider the adversarial classification setting where an adversary tries to manipulate the inputs at test time. Given $\varepsilon > 0$, they can move each point $x \sim \mathbb{P}$ to another point x' which is at distance at most ε from x[2]. The goal of this adversary is to maximize the $0/1$ risk the shifted points from $\mathbb{P}$. Formally, the loss associated to adversarial classification is defined as follows.

Definition 73 (Adversarial $0/1$ loss)**.** *Let $\varepsilon \geq 0$. We define the adversarial $0/1$ loss of level ε as:*

$$l_{0/1,\varepsilon}(x,y,f) = \sup_{x' \in B_\varepsilon(x)} \mathbf{1}_{ysign(f(x)) \leq 0}$$

We will denote $\mathcal{R}_{\varepsilon,\mathbb{P}}(f) := \mathcal{R}_{l_{0/1,\varepsilon},\mathbb{P}}(f)$, $\mathcal{R}^\star_{\varepsilon,\mathbb{P}} := \mathcal{R}^\star_{l_{0/1,\varepsilon},\mathbb{P}}$, $\mathcal{C}_\varepsilon(x,\eta,f) := \mathcal{C}_{l_{0/1,\varepsilon}}(x,\eta,f)$ and $\mathcal{C}^\star_\varepsilon(x,\eta) := \mathcal{C}^\star_{l_{0/1,\varepsilon}}(x,\eta)$ for every $\mathbb{P}$, x, f and η.

Specificity of the adversarial case The adversarial risk minimization problem is much more challenging than its standard counterpart because an inner supremum is added to the optimization objective. With this inner supremum, it is no longer possible to reduce the distributional problem to a pointwise minimization as it is usually done in the standard classification framework. In fact, the notions of consistency and calibration are significantly different in the adversarial setting. This means that the results obtained in the standard classification may no longer be valid in the adversarial setting (e.g., the calibration need not be sufficient for consistency), which makes the study of consistency much more complicated. As a first step towards analyzing the adversarial classification problem, we now adapt the notion of margin loss to the adversarial setting.

[2]Note that after shifting x to x', the point need not be in the support of $\mathbb{P}$ anymore.

Definition 74 (Adversarial margin loss)**.** *Let $\phi : \mathbb{R} \to \mathbb{R}_+$ be a margin loss and $\varepsilon \geq 0$. We define the adversarial loss of level ε associated with ϕ as:*

$$\phi_\varepsilon(x, y, f) = \sup_{x' \in B_\varepsilon(x)} \phi(yf(x'))$$

We say that ϕ is adversarially calibrated (resp. uniformly calibrated, resp. consistent) at level ε if ϕ_ε is calibrated (resp. uniformly calibrated, resp. consistent) wrt $l_{0/1,\varepsilon}$.

Note that a first important sanity check to make is verify that ϕ_ε and $l_{0/1,\varepsilon}$ are indeed measurable and well defined. The arguments are not trivial since it uses advanced arguments from measure theory, but it is necessary to establish measurability before going further on. Proposition 10 states the measurability of ϕ_ε and $l_{0/1,\varepsilon}$.

Proposition 10. *Let $\phi : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ be a measurable function and $\varepsilon \geq 0$. For every $f \in \mathcal{F}(\mathcal{X})$, $(x, y) \mapsto \phi_\varepsilon(x, y, f)$ and $(x, y) \mapsto l_{0/1,\varepsilon}(x, y, f)$ are universally measurable.*

Now that, we proved that the adversarial setting is properly defined, we can make a first observation: the calibration functions for ϕ and ϕ_ε are actually equal. This property might seem counter-intuitive at first sight as the adversarial risk is most of the time strictly larger than its standard counterpart. However, the calibration functions are only pointwise dependent, hence having the same prediction for any element of the ball $B_\varepsilon(x)$ suffices to reach the optimal calibration $\mathcal{C}^\star_\phi(x, \eta)$.

Proposition 11. *Let $\varepsilon > 0$. Let ϕ be a continuous classification margin loss. For all $x \in \mathcal{X}$ and $\eta \in [0, 1]$, we have*

$$\mathcal{C}^\star_{\phi_\varepsilon}(x, \eta) = \inf_{\alpha \in \mathbb{R}} \eta\phi(\alpha) + (1 - \eta)\phi(-\alpha) = \mathcal{C}^\star_\phi(x, \eta) \quad .$$

The last equality also holds for the adversarial $0/1$ loss.

B.2 Solving Adversarial Calibration

In this section, we study the calibration of adversarial margin losses with regard to the adversarial $0/1$ loss. We first provide necessary and sufficient conditions under which margin losses are adversarially calibrated. We then show that a wide range of surrogate

losses that are calibrated in the standard setting are not calibrated in the adversarial setting. Finally we propose a class of losses that are calibrated in the adversarial setting, namely the *shifted odd losses*.

B.2.1 Necessary and Sufficient Conditions for Calibration

One of our main contributions is to find necessary and sufficient conditions for calibration in the adversarial setting. In a brief, we identify that for studying calibration it is central to understand the case where there might be indecision for classifiers (i.e. $\eta = 1/2$). Indeed, in this case, either labelling positively or negatively the input x would lead the same loss for x. Next result provides a necessary condition for calibration.

Theorem 26 (Necessary condition for Calibration)**.** *Let ϕ be a continuous margin loss and $\varepsilon > 0$. If ϕ is adversarially calibrated at level ε, then ϕ is calibrated in the standard classification setting and $0 \notin \arg\min_{\alpha \in \bar{\mathbb{R}}} \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha)$.*

While the condition of calibration in the standard classification setting seems natural, we need to understand why $0 \notin \arg\min_{\alpha \in \bar{\mathbb{R}}} \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha)$. The intuition behind this result is that a sequence of functions simply converging towards 0 in the ball of radius ε around some x can take positive and negative values thus leading to suboptimal $0/1$ adversarial risk. It turns out that, given an additional mild assumption, this condition is actually sufficient to ensure calibration.

Theorem 27 (Sufficient condition for Calibration)**.** *Let ϕ be a continuous margin loss and $\varepsilon > 0$. If ϕ is decreasing and strictly decreasing in a neighbourhood of 0 and calibrated in the standard setting and $0 \notin \arg\min_{\alpha \in \bar{\mathbb{R}}} \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha)$, then ϕ is adversarially uniformly calibrated at level ε.*

Remark 6 (Decreasing hypothesis)**.** *For the reciprocal, the additional assumption that ϕ is decreasing and strictly decreasing in a neighborhood of 0 is not restrictive for usual losses. In Theorem* **??***, this assumption is stated as a necessary and sufficient condition for convex losses to be calibrated.*

B.2.2 Negative results

Thanks to Theorem 26, we can present two notable corollaries invalidating the use of two important classes of surrogate losses in the standard setting. The first class of losses are convex margin losses. These losses are maybe the most widely used in modern day

machine learning as they comprise the logistic loss or the margin loss that are the building block of most classification algorithms.

Corollary 2. *Let $\varepsilon > 0$. Then no convex margin loss can be adversarially calibrated at level ε.*

A convex loss satisfies $\frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha) \geq \phi(0)$, hence $0 \in \arg\min_{\alpha\in\mathbb{R}} \phi(\alpha) + \phi(-\alpha)$. From Theorem 26, we deduce the result. Then, ϕ is not adversarially calibrated at level ε. This result seems counter-intuitive and highlights the difficulty of optimizing and understanding the adversarial risk. Since convex losses are not adversarially calibrated, one may hope to rely on famous non-convex losses such as sigmoid and ramp losses. But, unfortunately, such losses are not calibrated either.

Corollary 3. *Let $\varepsilon > 0$. Let $\lambda \in \mathbb{R}$ and ψ be a lower-bounded odd function such that for all $\alpha \in \mathbb{R}$, $\psi > -\lambda$. We define ψ as $\phi(\alpha) = \lambda + \psi(\alpha)$. Then ϕ is not adversarially calibrated at level ε.*

Indeed, $\frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha) = \lambda$, so that $\arg\min_{\alpha\in\mathbb{R}} \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha) = \mathbb{R}$. Thanks to Theorem 26, ϕ is not adversarially calibrated at level ε.

B.2.3 Positive results

Theorem 27 also gives sufficient conditions for ϕ to be adversarially calibrated. Leveraging this result, we devise a class of margin losses that are indeed calibrated in the adversarial settings. We call this class *shifted odd losses*, and we define it as follows.

Definition 75 (Shifted odd losses)**.** *We say that ϕ is a* shifted odd margin loss *if there exists $\lambda \geq 0$, $\tau > 0$, and a continuous lower bounded decreasing odd function ψ that is strictly decreasing in a neighborhood of 0 such that for all $\alpha \in \mathbb{R}$, $\psi(\alpha) \geq -\lambda$ and $\phi(\alpha) = \lambda + \psi(\alpha - \tau)$.*

The key difference between a standard odd margin loss and a shifted odd margin loss is the variations of the function $\alpha \mapsto \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha)$. The primary difference is that, in the standard case the optima of this function are located at 0 while they are located in $-\infty$ and $+\infty$ in the adversarial setting. Let us give some examples of margin shifted odd losses below.

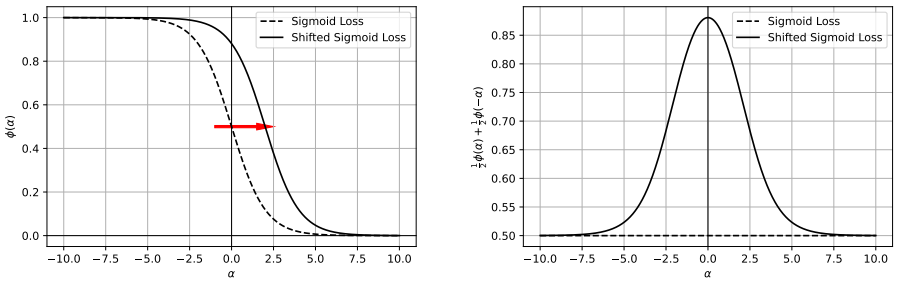


Figure B.1: Illustration of a calibrated loss in the adversarial setting. The sigmoid loss satisfy the hypothesis for ψ. Its shifted version is then calibrated for adversarial classification.

Example 3 (Shifted odd losses)**.** *For every $\varepsilon > 0$ and every $\tau > 0$, the shifted logistic loss, defined as follows, is adversarially calibrated at level ε: $\phi : \alpha \mapsto (1 + \exp\{(\alpha - \tau)\})^{-1}$ This loss is plotted on left in Figure B.1. We also plotted on right in Figure B.1 $\alpha \mapsto \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha)$ to justify that $0 \notin \arg\min_{\alpha\in\mathbb{R}} \frac{1}{2}\phi(\alpha) + \frac{1}{2}\phi(-\alpha)$. Also note that the shifted ramp loss also satisfies the same properties.*

A consequence of Theorem 27 is that shifted odd losses are adversarially calibrated, as demonstrated in Proposition 12 stated below.

Proposition 12. *Let ϕ be a shifted odd margin loss. For every $\varepsilon > 0$, ϕ is adversarially calibrated at level ε.*

B.3 Related Work and Discussions

We now explain the differences between our approach and the one proposed by [Bao et al., 2020, Awasthi et al., 2021a, Awasthi et al., 2021c]. The two main differences are the choice of the $0/1$ loss and the studied notion of consistency and calibration.

Alternative $0/1$ loss An alternative $0/1$ loss would the following: $l_{\leq}(f(x), y) = \mathbf{1}_{yf(x)\leq 0}$. This loss penalizes indecision: i.e. predicting 0 would lead to a pointwise risk of 1 for $y = 1$ and $y = -1$ while the $0/1$ loss $l_{0/1}$ returns 1 for $y = 1$ and 0 for $y = -1$. This definition was used by [Bao et al., 2020, Awasthi et al., 2021a, Awasthi et al., 2021c] to prove their calibration and consistency results. While [Bartlett et al., 2006] was not explicit on the choice for the $0/1$ loss, [Steinwart, 2007] explicitly mentions that the $0/1$ loss is not a margin loss. The use of this loss is not suited for studying consistency and leads to inaccurate results as shown in the following counterexample. On $\mathcal{X} = \mathbb{R}$,

let $\mathbb{P}$ defined as $\mathbb{P} = \frac{1}{2}(\delta_{x=0,y=1} + \delta_{x=0,y=-1})$ and $\phi : \mathbb{R} \to \mathbb{R}$ be a margin based loss. The ϕ-risk minimization problem writes $\inf_\alpha \frac{1}{2}(\phi(\alpha) + \phi(-\alpha))$. For any convex functional ϕ the optimum is attained for $\alpha = 0$. $f_n : x \mapsto 0$ is a minimizing sequence for the ϕ-risk. However $R_{l_\leq}(f_n) = 1$ for all n and $R^*_{l_\leq} = \frac{1}{2}$. Then we deduce that no convex margin based loss is consistent wrt $l_\leq$. Consequently, the $0/1$ loss to be used in adversarial consistency needs to be $l_{0/1,\varepsilon}(x, y, f) = \sup_{x' \in B_\varepsilon(x)} \mathbf{1}_{y\text{sign}(f(x)) \leq 0}$, otherwise the obtained results might be innacurate.

$\mathcal{H}$-consistency and $\mathcal{H}$-calibration [Bao et al., 2020, Awasthi et al., 2021a, Awasthi et al., 2021c] proposed to study $\mathcal{H}$-calibration and $\mathcal{H}$-consistency in the adversarial setting, i.e. calibration and consistency when minimizing sequences are in $\mathcal{H}$. However, even in the standard classification setting, the link between both notions in this extended setting is not clear at all since a pointwise minimization of the risk cannot be done. To our knowledge, there is only one research paper [Long and Servedio, 2013] that focuses on this notion in standard setting. They do it in the restricted case of realisability, i.e. when the standard optimal risk associated with the $0/1$ loss equals 0. We believe that studying $\mathcal{H}$-consistency and $\mathcal{H}$-calibration in the adversarial setting is a bit anticipated. For these reasons, we focus only on calibration and consistency on the space of measurable functions $\mathcal{F}(\mathcal{X})$. However, note that many of our results can be adapted to $\mathcal{H}$-calibration.

About the Adversarial Bayes Risk and Game Theory. A recent trend of work has focused on analyzing the adversarial risk from multiple point of views. [Bhagoji et al., 2019] as well as [Pydi and Jog, 2020b, Pydi and Jog, 2021] showed that the adversarial optimal Bayes classifier can be written as optimal transport for a well chosen cost. Another line of work [Pinot et al., 2020, Meunier et al., 2021, Pydi and Jog, 2021] have focused on a game theoretic approach for analyzing the adversarial risk having interest in the nature of equilibria between the classifier and the attacker. Recently, some researchers [Awasthi et al., 2021b, Bungert et al., 2021] proved encouraging results on the existence of an optimal Bayes classifier in the adversarial setting under mild assumptions.

C A general study of contact tracing for epidemics

Contents

This section contains a summary of some of my works on modeling the COVID-19 pandemic. We will focus on contact tracing, i.e. soft public policies where a central authority chooses to isolate a selected few individuals instead of using a global lockdown. Let us first give a few definitions and preliminary observations, so that we can state the problems we will tackle.

C.1 Introduction to compartimental models of epidemics

Consider a population of N individuals, where an epidemic circulates. To model the diffusion of the epidemic, we will first use the homogeneous SIR model, introduced by [Kermack and McKendrick, 1927]. In this model, we consider that the population is *homogeneous*, i.e. all individuals have the same characteristics relative to the disease. We will remove this hypothesis in appendix C.3 to analyze age-stratified models.

Figure C.1: Illustration of the SIR model

The idea of SIR models is to separate the population at any given point t into three categories : the **susceptibles** (whose number is called S(t)), who can catch the disease, the **infected**, which have it, and the **recovered**, who are immune and cannot catch it anymore. Furthermore, we assume that we study over a time period $T > 0$ that is short enough for the number of susceptible to have very little relative variation, i.e. $S(t) = S$ is constant over $[1, \dots, T]$.

Definition 76 (Homogeneous SIR model)**.** *An SIR model is defined by an initial situation* (S_0, I_0, R_0)*, coefficients* $\beta, \gamma \in (0, 1)$*, and a discrete process* $(S, I, R)(t)$ *such that* $(S, I, R)(0) = (S_0, I_0, R_0)$*, and for all* $t \in [1, \dots, T]$*:*

$$\begin{aligned} S(t+1) &= S(t) = S \\ I(t+1) &= \beta \frac{S}{N} I(t) - \gamma I(t) \\ R(t+1) &= \gamma I(t) \end{aligned}$$

where β *represents the average number of people contaminated by each infected, and* γ *the proportion of infected that recovers, all during a given period.*

We will further separate β into two components :

$$\beta = \underbrace{n}_{\substack{\text{average number} \\ \text{of contacts}}} \times \underbrace{r}_{\substack{\text{average risk} \\ \text{of contamination}}}$$

This framework can be generalized when the population is stratified into K groups. We call $S_1, \dots, S_K$ the number of susceptibles, $I_1, \dots, I_k$ the number of infected, and $R_1, \dots, R_k$ the number of recovered.

Definition 77 (Stratified SIR model)**.** *A stratified SIR model is defined by an initial situation* $(S_k(0), I_k(0), R_k(0)$ *for* $k \in \{1, \dots, K\}$*, coefficients* $\beta_{i,j}, \gamma \in (0,1)$ *for* $(i,j) \in \{1, \dots, K\}$*, and a discrete process* $(S_k, I_k, R_k)(t)$ *such that for all* $k \in \{1, \dots, K\}$*,* $t \in [1, \dots, T]$*:*

$$\begin{aligned} S_k(t+1) &= S(t) = S \\ I_k(t+1) &= \sum_j \beta_{j,k} \frac{I_j(t)}{N_j} - \gamma I_k(t) \\ R_k(t+1) &= \gamma I_k(t) \end{aligned}$$

where $\beta_{j,k}$ *represents the average number of infectious contacts from individuals in cohort* j *with susceptibles of cohort* k*, and* γ *the proportion of infected that recovers, all during a given period.*

Under this framework (and a multi-class generalization), we will study the following questions :

Q1: *What is the most efficient contact tracing policy at different stage of the epidemic ?*

Q2: *What are other, more global policies that can keep the epidemic under control while minimizing the lockdown time ?*

C.2 General theoretical study : optimal forms of contact tracing depending on the prevalence

C.2.1 Different forms of contact tracing

In this section, we will analyze and compare contact tracing policies, i.e. selective isolation of a portion of the population. Given an "isolation budget" b, what are efficient ways to choose the people that are isolated ? We will compare three kinds of contact tracing policies :

- **Random Sampling :** draw an individual at random in the cohort that exhibits the highest prevalence, ask him to test and isolate if positive.

- **Standard Contact Tracing :** ask all individuals that are tested positive for the list of their contacts, and send these contacts a message asking them to test and isolate. We are allowed to priorize contacts based on specific information such as age.
- **App-based Contact Tracing :** we assume that some application is adopted by some given proportion of the population. When someone is tested positive, all of its recent contact are automatically asked to test and isolate. The app does not have access to any specific information about people, so works as if the population was homogeneous.

Such contact tracing policies have two major objectives :

1. Prevent infections and/or ICUs;
2. Gather information on the contact matrices and epidemic coefficients;

To evaluate the efficiency of contact tracing mechanisms, several criterion can be considered :

- The number of ICUs avoided over a given period of time;
- The impact on the epidemic development (through for example the reproduction rate);
- The amount of information gathered (speed to adjust to changing contact matrices, convergence rate of the estimators...)

We will focus here on the first criterion, i.e. the number of ICUs avoided. Hence, we will assume that the contact tracing is limited in its scale, and does not directly affect the epidemic development. We will provide some insights on the information gathering later.

C.2.2 The different steps of contact tracing

Any contact tracing mechanism can be divided into two major steps :

1. Identify infectious individuals;
2. Priorize the ones to isolate based on their contact patterns.

Step 1. amounts to estimated the probability of being infected. Random sampling does that by using the *prior* probability $\frac{I_k}{N_k}$, which is the prevalence in a given cohort k, whereas other contact tracing forms use the posterior probability $\mathbb{P}[infected|contact]$ for the contacts of infected individuals.

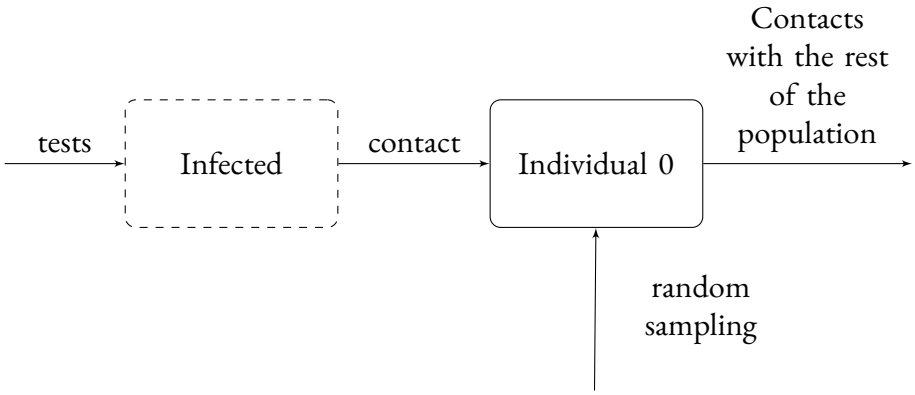


Figure C.2: The contact tracing process. The first step is to identify infectious individuals, either as a contact of someone tested positive, or through random sampling. Then, we can priorize these infected by how many other people they are susceptible to contaminate.

Step 2. amounts to evaluate the remaining fraction of the infectious window that remains after calling the individual, ant then to count the number of ICUs caused by an infected, depending on its cohort, over a given time span. We will study that second aspect in appendix C.3

C.2.3 Homogeneous population : the prevalence threshold

When the population is homogeneous, two things happen :

- As every individual has the same action toward the evolution of the epidemic, the performance of a contact tracing mechanism only depends on its ability to find infected individuals, as how late they are found in their infectious window;
- Contact tracing and app-based contact tracing become identical, as there can be no priorization.

Definition 78 (Contact tracing delay)**.** *The contact tracing delay of a policy is the average proportion of the infectious window that is already over when an individual is called.*

Proposition 13. *The contact tracing delay of Random Sampling is always* $\frac{1}{2}$.

We can thus compare both policies as follow :

Theorem 28. *In the homogeneous scenario, at a given time t, the ct-score of Standard Contact Tracing of delay δ and Random Sampling are given by :*

- *Standard CT :* δr
- *Random Sampling :* $\frac{I(t)}{2N}$

It follows that standard contact tracing performs better for $\frac{I(t)}{N} \leq 2\delta r$*, whereas random sampling performs better for* $\frac{I(t)}{N} > 2\delta r$*. We call* $2\delta r$ *the prevalence threshold.*

C.3 Analysis on real-world contact matrix : maximizing the efficiency per call

In this section, we focus on the age-stratified SIR model, and study the following question :

Q: *How should we divide our call budget between the different cohorts ?*

We make the following hypothesis :

- γ is constant across the different cohorts;
- all cohorts have the same population N
- $\beta_{i,j} = N n_{i,j} r$, where $n_{i,j}$ is the average number of contacts from one person in cohort i in cohort j

It follows that we only need the contact matrix $M = (n_{i,j})_{i,j}$, which represents the average number of contacts between two cohorts. Such contact matrices would usually be obtained through contact tracing during the epidemic, to encompass variations which inevitably occur at each change of public policy (opening the bars and restaurants, restricting the number of people in closed space, etc). In this paper, we will use the contact matrices provided by [Mistry et al., 2021] for France. Keep in mind that they reflect a "standard" behavior of the population, and that results would probably be very different depending on the period of the epidemic.

When using a vector notation for definition 77, we can easily see that :

$$I(t+1) = (M - \gamma Id)I(t) = GI(t)$$

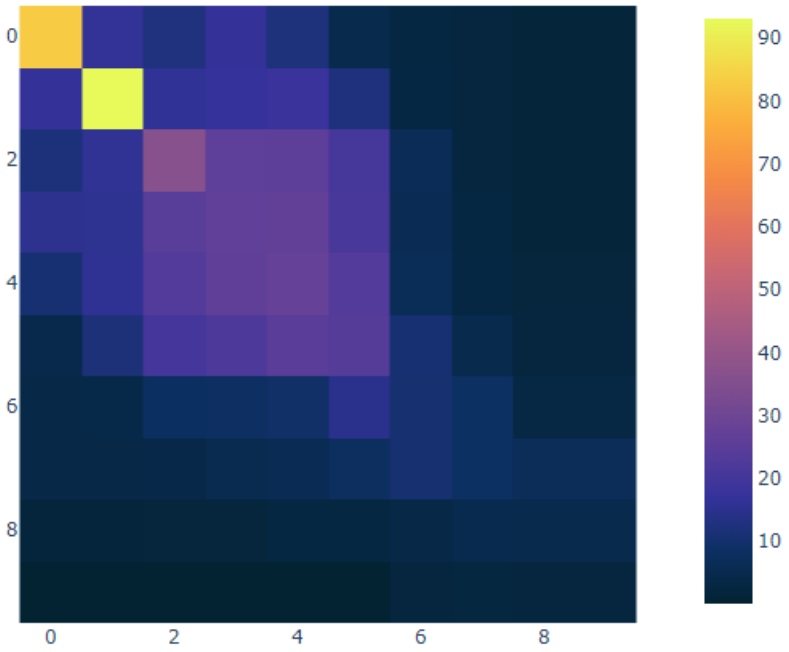


Figure C.3: Contact matrix from [Mistry et al., 2021], aggregated by cohorts of 10 years.

Where M is the contact matrix. It follows that the number of ICUs avoided by a call to someone in cohort j, over a period T, is :

Theorem 29 (Number of ICUs avoided by a call)**.** *The number of ICUs avoided by a call to someone in cohort j is :*

$$\sum_{t=1}^{T}\sum_{k=1}^{k}\alpha_k G_{j,k}^{t} \tag{C.1}$$

When plotting that for several values of t, we ca observe that although it is efficient on the short run to priorize calls to old people, it is much more efficient on the long run to call young people, as they have way more contacts and contaminate more people by ripple.

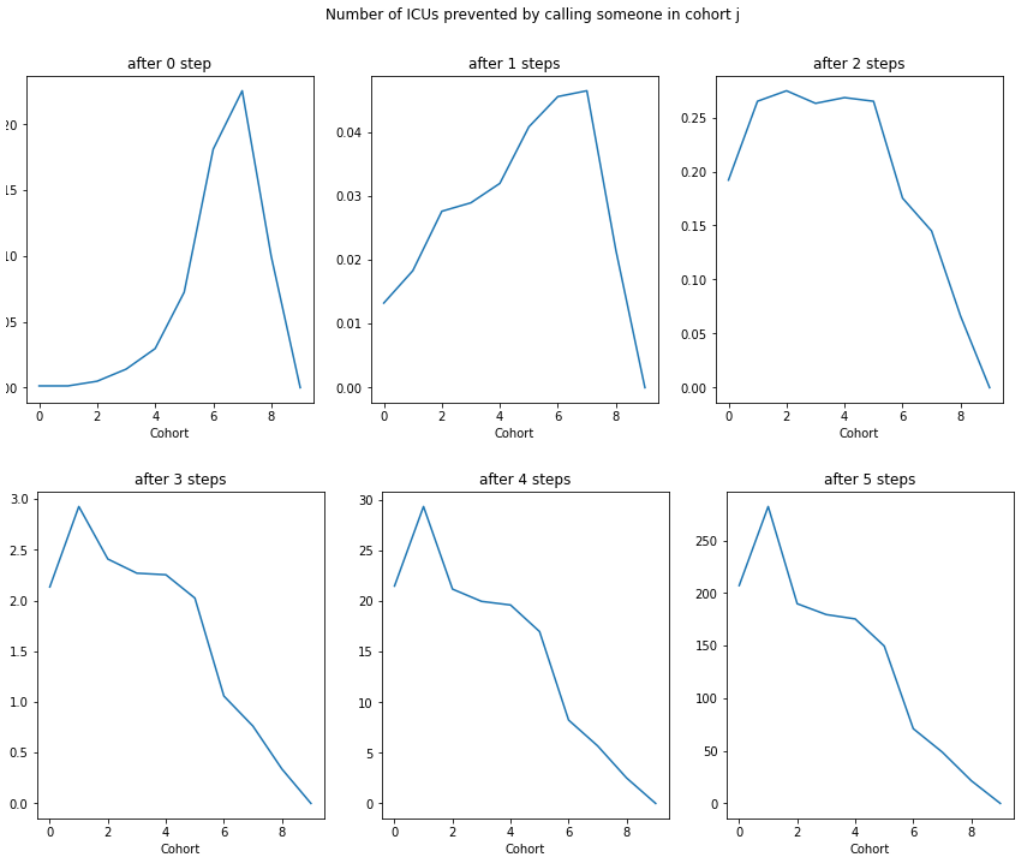


Figure C.4: Number of ICU avoided by a call over a period T, depending on the age group of the person called

RÉSUMÉ

Les modèles d'apprentissage automatique sont désormais au coeur de nombre d'applications, y compris les plus critiques comme les voitures autonomes et le diagnostique médical. Il est par conséquent important que les systèmes embarqués soient capables d'identifier, et si possible de neutraliser toute vulnérabilité de ces modèles, afin de garantir leur bon fonctionnement.

Cette thèse se concentre sur l'un des principaux problèmes de sécurité en machine learning : les attaques par exemples adversariaux. Nous étudions l'existence de modèles robustes contre ces attaques, en abordant la question sous l'angle de la théorie des jeux et du transport optimal. Nous proposons ensuite un cadre général permettant d'améliorer les garanties de robustesse que les algorithmes peuvent offrir.

MOTS CLÉS

Exemples adverses - Théorie des Jeux - Transport optimal - Théorie de l'apprentissage statistique

ABSTRACT

Machine learning models are now at the heart of many applications, including the most critical such as autonomous cars and medical diagnosis. It is thus important that embedded systems are able to identify, and if possible correct, any vulnerability of these models, to guarantee their performance.

This thesis focuses on adversarial example attacks, which are one of the main security issues in machine learning. We study the existence of robust models against such attacks, under the prism of game theory and optimal transport. We then provide a general framework to obtain better guarantees of robustness for algorithms.

KEYWORDS

Adversarial examples - Game theory - Optimal transport - Statistical learning theory