PRASS: Probabilistic Risk-averse Robust Learning with Stochastic Search

Tianle Zhang[1] , **Yanghao Zhang**[1] , **Ronghui Mu**[1] , **Jiaxu Liu**[1] ,
Jonathan Fieldsend[2] , **Wenjie Ruan**[1*]

[1]Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, UK
[2]Department of Computer Science, University of Exeter, Exeter, EX4 4QF, UK
{T.Zhang, Y.Zhang, R.Mu, J.Liu}@liverpool.ac.uk, J.E.Fieldsend@exeter.ac.uk, w.ruan@trustai.uk

Abstract

Deep learning models, despite their remarkable success in various tasks, have been shown to be vulnerable to adversarial perturbations. Although robust learning techniques that consider adversarial risks against worst-case perturbations can effectively increase a model's robustness, they may not always be the most suitable approach. This is due to the fact that in certain scenarios, perturbations are more likely to occur probabilistically rather than being intentionally crafted by attackers. To address this challenge, we propose a novel *risk-averse robust learning* method based on entropic value-at-risk, called PRASS (**P**robabilistical **R**isk-**A**verse Robust Learning with **S**tochastic **S**earch). Our approach leverages principles of stochastic optimisation and considers the use of perturbing distributions rather than solely worst-case adversaries. By applying adaptive stochastic search to parameterised distributions, we further enhance the scalability of PRASS to handle distributional robustness. Empirical experiments demonstrate that PRASS outperforms existing state-of-the-art baselines.

1 Introduction

Deep neural networks (DNNs) have seen widespread adoption across a multitude of domains [Zhang *et al.*, 2020b; Mu *et al.*, 2021]. Their success, however, has been accompanied by growing concerns about robustness, especially in safety-critical environments [Huang *et al.*, 2020; Huang *et al.*, 2017]. This is due to the vulnerability of neural networks to *adversarial perturbations*, where subtle modifications to benign inputs can significantly mislead the model's predictions without significantly affecting human perception. Considering the wide range of adversarial attacks, a variety of defense methods have arisen, with *adversarial training* [Madry *et al.*, 2018] notably taking the spotlight. However, existing research primarily focuses on improving adversarial robustness against the worst-case risks posed by explicit adversaries. Though highly suitable in some instances, these approaches are not universally applicable.

Firstly, there can be concerns related to robustness against naturally occurring corruptions or *random input perturbations*, as opposed to an explicit adversary. For example, in autonomous systems, input data may be derived from various sources, such as onboard sensors, GPS, or interaction or communication with a complex environment. These may contain noise, be subject to signal degradation, or other variations due to environmental factors such as weather or electromagnetic interference. We desire our neural networks in such situations to accurately process and analyse input data, making safe and robust decisions in the presence of such random variations. In this context, a classifier must account for these variations, but some degree of perturbation or risk will typically be deemed acceptable. The crux is not about achieving complete immunity from adversarial examples, as that might often be unrealistic or even excessive. Rather, there is a balance to be had: enabling robustness with acceptable risk thresholds, ensuring optimal performance even in the face of uncertainty.

Secondly, in practice, we are usually concerned with the overall network robustness, *i.e.*, robustness across the range of potential inputs, rather than its input-specific robustness. This has inspired the exploration of robust learning with adversarial risk as a metric of the model's worst-case performance on adversarial perturbations. Based on such an ideal robustness definition, a classifier is deemed satisfactory only if it can withstand the full gamut of potential perturbations. However, such stringent safety requirements are rarely achievable and applicable in real-world settings. As [ISO, 2014] posits: "*safety risks and dangers are inescapable; residual risks endure even after risk reduction measures have been executed*". Furthermore, prior work [Schmidt *et al.*, 2018; Yin *et al.*, 2019] has demonstrated that these robust learning approaches exhibit poor generalisation from training to testing phases, substantiated by both theoretical analyses and empirical tests on real networks, significantly diminishing their broad-scale applicability.

To address these limitations, we propose a risk-averse robust learning framework for DNNs, which originates from relaxing worst-case adversarial risks by introducing Entropic Value-at-Risk (EVaR) risk and taking perturbation distributions into account. The risk of a classifier, termed as the *probabilistic robustness*, is calculated with a very tight upper bound derived from the Chernoff inequality over an input perturbation distribution. Further expanding on this concept,

*Corresponding Author

the risk metric evolves into a *distributional robustness* definition by employing adaptive stochastic search to iteratively update the perturbation distribution. It's crucial to understand that our framework does not seek to replace adversarial risk or serve as a means to learn adversarially robust networks; instead, it pioneers a risk-aversion training scheme that capably mitigates the potential risk of perturbations overall, rendering it more appropriate and pragmatic in some scenarios.

In summary, our contributions in this work encompass: **(C1)** Ours is the *first* work expanding on risk-sensitive robust learning framework for DNNs from a distribution perspective. Inspired by risk-aware optimisation and stochastic optimisation, we present ***P**robabilistic **R**isk-**A**verse Robust Learning with **S**tochastic **S**earch* (***PRASS***). Here, the robustness risk is calculated with the tightest upper bound from the Chernoff inequality. **(C2)** We provide theoretical and empirical results showing that the proposed risk metric has superior generalisation performance to its corresponding adversarial risks, particularly in high-dimensions, with bounds on the generalisation error respectively scaling as $O(\log(d))$ and $\sqrt{d}$ in the size of the network, respectively. **(C3)** We propose a tractable algorithm for risk-sensitive robust learning generalised to the point-wise distribution with stochastic search. We conduct experiments on the MNIST, CIFAR-10 and CIFAR-100 datasets, and the results validate the effectiveness of our proposed methods on building risk-averse models. These models are robust to the majority of perturbations, and achieve superior performance compared to the alternative state-of-the-art robust learning methods.

2 Related Work

Robust learning is an an emerging research topic with various attack and defense methods being proposed. We now discuss at a high level some of the major approaches.

Empirical Adversarial Training. A widespread empirical defence approach, empirical adversarial training [Wang *et al.*, 2021; Jin *et al.*, 2022; Zhang *et al.*, 2020a; Gowal *et al.*, 2019; Balunovic and Vechev, 2020] endeavours to approximate the solution to a minimax problem, thereby identifying an optimal hypothesis f from a hypothesis class, $\mathcal{F}$. The inner maximisation problem is frequently approximated by empirical adversarial attacks, which include gradient descent-based methods such as FGSM [Wong *et al.*, 2020] and PGD [Madry *et al.*, 2018]. The outer minimisation problem, on the other hand, optimises model parameters using gradient descent-based optimisers, similar to conventional training frameworks. Adversarially trained neural networks, despite demonstrating empirical resilience to adversarial attacks, remain susceptible to advanced attack methodologies due to the lack of verifiable theoretical guarantees [Tjeng *et al.*, 2019].

Certified Adversarial Training. Certified adversarial training focuses on enhancing models' provable robust accuracy, validated by robustness verifiers [Balunovic and Vechev, 2020; Croce and Hein, 2020; Croce *et al.*, 2019; Gowal *et al.*, 2019; Zhang *et al.*, 2020a; Fan and Li, 2021]. This technique minimises an upper bound of the loss across all perturbations, as opposed to training with adversarial examples. Certified bounds can be derived from the dual optimisation problem, or via linear relaxation [Balunovic and Vechev, 2020; Croce and Hein, 2020; Mirman *et al.*, 2018; Croce *et al.*, 2019; Zhang *et al.*, 2020a], and interval bound propagation (IBP) [Zhang *et al.*, 2020a; Lyu *et al.*, 2021; Zhang *et al.*, 2021; Shi *et al.*, 2021]. An emerging branch of robust defence focuses on probability-based defence methods, which we now briefly discuss.

Probability-based Defense. A popular implementation of this technique is randomised-smoothing approaches, as methods using this are capable of providing probabilistic robustness guarantees. Several works [Cohen *et al.*, 2019; Lécuyer *et al.*, 2019; Zhai *et al.*, 2020; Salman *et al.*, 2019; Awasthi *et al.*, 2020] explore robust training approaches for robust randomised models. For instance, one can augment the training dataset with noise [Cohen *et al.*, 2019; Lécuyer *et al.*, 2019] – an augmentation approach that has proven effective in practice and is currently one of the most widely-used training approaches under probabilistic robustness settings. Very recently, Robey *et al.* 2022 establish a probabilistically robust learning paradigm capable of balancing accuracy and robustness by enforcing robustness for the majority of perturbations, rather than all. Li *et al.* 2023 propose tilted empirical risk minimisation (TERM) to use exponential tilting to flexibly tune the tradeoff between average-loss and worst-loss.

We introduce a risk-averse training framework aimed at improving the probabilistic robustness under an uncertain perturbing distribution. Our method *differs from current works* that only focus on tail samples, EVaR considers all samples across the *entire distribution*, leading to a more comprehensive risk assessment (see *Sec 4.1*); *instead of considering a specific predefined perturbation distribution*, e.g. the uniform or normal distribution, we adopt an adaptive stochastic search to identify worst-case perturbation *distributions*, providing a practical and feasible *closed-form* solution.

3 Preliminaries

Supervised Learning. In classification tasks with K categories, we consider data sourced from an unknown distribution $\mathcal{D}$, encompassing feature-label pairs $(\mathbf{x}, y)$, where $\mathbf{x} \in \mathcal{X} = \mathbb{R}^n$ are n-dimensional instances, and $y \in \mathcal{Y} = \{1, \dots, K\}$ are the associated labels. The goal of supervised learning is to identify an optimal hypothesis $f^* : \mathcal{X} \to \mathcal{Y}$ from a class of hypotheses $\mathcal{F}$, typically comprising models parameterised by $\boldsymbol{\theta} \in \Theta$. This hypothesis can be obtained by minimising the *expected risk* associated with f, as defined:

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathcal{L}(f(\mathbf{x}), y)], \tag{1}$$

where, $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ is a loss function. Ordinarily, as the distribution $\mathcal{D}$ is unknown, calculating the objective in Eq. 1 is infeasible. However, given a training set of N distinct i.i.d. samples $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$ drawn from $\mathcal{D}$, the objective can be approximated using *empirical risk*, the average loss over samples:

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathcal{L}(f(\mathbf{x}), y)] \approx \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}\left(f\left(\mathbf{x}_i\right), y_i\right). \tag{2}$$

Adversarial Training. Adversarial training aims to generate hypotheses $f \in \mathcal{F}$ that remain *resilient to input perturbations*. Specifically, if a hypothesis f correctly classifies an input $\mathbf{x}$, it should also correctly classify a slightly adjusted input $\widetilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\delta}$. Here hypotheses are developed using the essential supremum of the objective function across the perturbation space, reflecting the worst-case scenario:

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\sup_{\boldsymbol{\delta}\in\Delta} \mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)], \quad (3)$$

where $\Delta \subset \mathbb{R}^n$ denotes a set of "imperceptible" perturbations, e.g., $\Delta = \{\delta \in \mathbb{R}^n : \|\delta\|_\infty \leq \epsilon\}$. Eq. 1 can be considered a special case of Eq. 3 when $\epsilon = 0$. This minimax optimisation is typically approached in two stages: first, the inner maximisation is approximated to generate adversarial examples, followed by optimising the neural networks' parameters based on these adversarial examples. For instance, PGD approximately solves the inner maximisation involving multiple gradient steps, *i.e.*, $\boldsymbol{\delta}_i^{t+1} = \Pi\left(\boldsymbol{\delta}_i^t + \alpha \cdot \text{sign}\left(\nabla_{\mathbf{x}} \mathcal{L}\left(f_{\boldsymbol{\theta}}\left(\mathbf{x}_i + \boldsymbol{\delta}_i^t\right), y_i\right)\right)\right)$, where $\boldsymbol{\delta}_i^t$ is the adversarial perturbation at the t-th step, $\Pi(\cdot)$ is the projection function, and α is a small step size. In essence, $\boldsymbol{\delta}_i^t$ always converges to a local optima influenced by the initialisation $\boldsymbol{\delta}_i^0$.

Distributional Adversarial Training. Adversarial distributional training [Dong *et al.*, 2020] has been proposed to defend against perturbations of the captured distribution surrounding samples. Here, adversarial perturbations around each input sample $\mathbf{x}_i$ follow the corresponding distribution $p(\boldsymbol{\delta})$ within Δ. This method can expressed as a distribution-based minimax optimisation problem:

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \sup_{p(\boldsymbol{\delta})\in\mathcal{P}} \mathbb{E}_{p(\boldsymbol{\delta})}\left[\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)\right], \quad (4)$$

where $\mathcal{P} = \{p : supp(p) \subseteq \Delta\}$ denotes a set of distributions with support within the Δ neighbourhood of natural examples.

Probabilistically Robust Learning. The goal of probabilistically Robust Learning is to provide robustness against the majority of perturbations, with a tolerance for a small proportion of perturbations in regions of negligible volume in the perturbation space Δ. Consequently, the objective here is to upper bound the loss function for a proportion $1-\gamma$ of the mass of the perturbation space Δ, not for all $\boldsymbol{\delta} \in \Delta$. Thus the following provides a definition of the associated operator introduced in [Robey *et al.*, 2022]. This function operates on the loss random variable $\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)$ and yields the value r for which the loss does not exceed r for a proportion $1-\gamma$ of the perturbation space.

Definition 1. Let $\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)$ denote a random variable representing the loss of model f for input $\mathbf{x}$ perturbed by $\boldsymbol{\delta}$ and true label y. Let Δ be the perturbation space with a defined measure. Then, the γ-$ess\sup$ operator (or γ-essential supremum) is then formulated as an optimisation problem, *i.e.*, $\gamma\text{-}ess\sup = \min_{r\in\mathbb{R}} P_{\boldsymbol{\delta}\sim\Delta}[\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y) \leq r] > 1-\gamma$, where γ is a small positive value that represents a tolerable level of probabilistic robustness.

Here, this definition frames γ-$ess\sup$ as a tool that quantifies the robustness of model f with respect to the proportion γ when accounting for losses caused by perturbations. The objective function for probabilistically robust learning, which focuses on the misclassification probability across a perturbation distribution, can be expressed as:

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\gamma\text{-}ess\sup_{\boldsymbol{\delta}\in\Delta} \mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)]. \quad (5)$$

Adaptive Stochastic Search. The *adaptive stochastic search* method, proposed by Zhou and Hu 2014, is designed to solve a general maximisation problem: $z^* \in \arg\max_{z\in\mathcal{Z}} G(z)$, where $\mathcal{Z} \subseteq \mathbb{R}^n$ is a nonempty compact set in $\mathbb{R}^n$, and $G : \mathcal{Z} \to \mathcal{R}$ is a deterministic real-valued function. Traditional optimisation methods often falter when the objective function $G(\cdot)$ exhibits non-convex, discontinuous, and non-differentiable traits. This method overcomes such hurdles by stochastically approximating the function. A solution z is sampled from a probability distribution $p(z;\eta)$ from the exponential family, parameterised by η. The reformulated problem becomes:

$$\eta^* = \arg\max_{\eta} \int G(z)p(z;\eta)\mathrm{d}z. \quad (6)$$

This reformulation, thanks to its probabilistic nature, exhibits properties conducive to optimisation. The algorithmic implementation can be further simplified by introducing a continuous, non-decreasing shape function, $S(\cdot) : \mathcal{R} \to \mathcal{R}^+$, ensuring the optimal solution remains unaffected. Thus, the final problem becomes: $\eta^* = \arg_\eta\max \int S(G(z))p(z;\eta)\mathrm{d}z = \arg_\eta\max \mathbb{E}_\eta[S(G(z))]$.

To tackle this optimisation problem, candidate solutions, z, are drawn from $p(z;\eta)$ within the solution space $\mathcal{Z}$. Afterwards, a gradient ascent method can be applied to Eq. 6 to update the parameter η. Depending on the chosen probability distribution for sampling z, a closed-form solution for the gradient of the above objective function with respect to η may be available. (The derivation is included in the Appendix A.1.)

4 Methodology

This section proposes a novel and generalisable framework: *Probabilistic Risk-averse Robust Learning with Stochastic Search* (***PRASS***). Its primary goal is to characterise the perturbation distribution and train a risk-averse model via mitigating the Entropic Value-at-Risk (EVaR) risk over the captured worst-case perturbation distribution. This particular risk exhibits convexity, facilitating a straightforward and efficient resolution for risk-averse robust learning. Additionally, EVaR risk capitalises on the entire samples across sampling distribution, as opposed to solely relying on tail samples.

4.1 Probabilistic Risk-averse Robust Learning

Optimising the objective function for probabilistically robust learning as per Eq. 5 is very challenging. This is due to the γ-essential supremum operator associated with random perturbations being non-convex, non-smooth, and highly

stochastic. Note that the γ-essential supremum is alternatively termed Value-at-Risk (VaR) [Robey *et al.*, 2022], especially within the risk-aware control domain [Majumdar and Pavone, 2017; Ahmadi *et al.*, 2022].

Definition 2. Given a random variable r and a function $g[\cdot]$ yielding a scalar output (i.e., the real scalar value correlated with the loss function), and considering a risk level $\gamma \in [0, 1]$, the Value-at-Risk, denoted as $VaR_\gamma(g(r))$, is the infimum over $\zeta \in \mathbb{R}$ where $g(r)$ situates beneath ζ with a probability not less than $1 - \gamma$. Formally, this can be expressed as $VaR_\gamma(g(r)) = \inf_{\zeta \in \mathbb{R}} \zeta : P[g(r) \leq \zeta] \geq 1 - \gamma$.

Unfortunately, the tractability of VaR is hampered by several constraints, such as the non-coherent risk measure and non-convex function. Thus, VaR is frequently replaced with Conditional Value-at-Risk (CVaR), which calculates the expected value of those losses that lie in a tail region where the threshold VaR_γ is exceeded.

Definition 3. The Conditional-Values-at-Risk with a risk level $\gamma \in (0, 1]$, denoted as $CVaR_\gamma(g(r))$, is the expected value of f within the tail distribution that equals or surpasses the threshold VaR_γ, i.e., $CVaR_\gamma(g(r)) = \mathbb{E}[g(r) \mid g(r) \geq VaR_\gamma(g(r))] = \inf_{\zeta \in \mathbb{R}} \left\{ \zeta + \frac{\mathbb{E}[g(r)-\zeta]_+}{1-\gamma} \right\}$, where $[\cdot] = \max\{0, \cdot\}$.

The inf operator inside the $CVaR_\gamma(g(r))$ is convex w.r.t. ζ; this property ensures that the optimisation problem to estimate CVaR can be efficiently solved, since $[\cdot]$ is increasing and convex [Rockafellar *et al.*, 2000]. Despite its computational advantage and valuable insights into the extreme events (in the tail of loss distribution), it does so at the expense of overlooking the remainder of the loss distribution. Thus we employ Entropic Value-at-Risk (EVaR) that is the tightest upper bound derived from the Chernoff inequality for the VaR.

Definition 4. The Entropic Values-at-Risk with risk level $\gamma \in (0, 1]$ denoted as $EVaR_\gamma(g(r))$ is defined as the infimum over $\zeta > 0$ of the Chernoff bound for $g(r)$ w.r.t. random variable r, *i.e.*, $EVaR_\gamma(g(r)) = \inf_{\zeta > 0} \frac{1}{\zeta} \ln \left(\frac{\mathbb{E}\left[e^{\zeta g(r)}\right]}{\gamma} \right)$.

While EVaR serves as the tightest upper bound derived from the Chernoff inequality for the γ-essential supremum, it also incorporates the entirety of the loss distribution, not just the tail. The definition of EVaR uses the exponential moment of the entire loss function weighted by a scalar ζ. This implies that every sample from the loss distribution has a role in shaping the EVaR value, thereby ensuring a more comprehensive view of the risks embedded in the distribution.

Proposition 1. *The EVaR is an upper bound for the γ-ess sup and CVaR with the same risk γ, that is, γ-ess sup $g(r) \leq CVaR_{1-\gamma}(g(r)) \leq EVaR_{1-\gamma}(g(r))$, with equality when $\gamma = 0$ or $\gamma = 1$. Moreover, $\mathbb{E}[g(r)] \leq EVaR_{1-\gamma}(g(r)) \leq ess\sup(g(r))$, where $EVaR_0(g(r)) = \mathbb{E}[g(r)]$ and $\lim_{\gamma \to 0} EVaR_{1-\gamma}(g(r)) = ess\sup(g(r))$.*

The relationship between the three risk measures is illustrated in Fig. 1. As Proposition 1 shows, EVaR risk is a more risk-averse risk metric than both γ-ess sup and CVaR, and can enhance the overall robustness of the models with a balance between capturing average and worst-case scenarios. While γ-ess sup and CVaR hone in exclusively on the tail distribution—essentially the extreme scenarios—EVaR integrates the full breadth of the loss distribution, facilitating a more rounded understanding of risks.

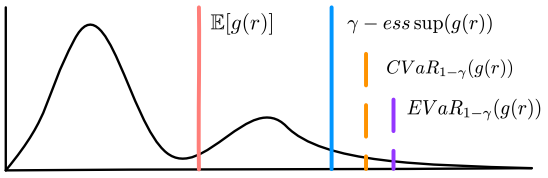

Figure 1: An illustration of three risk measures - γ-essential supremum, Conditional Value-at-Risk, and Entropic-Value-at-Risk. Per their definitions in Section 4.1, γ-ess sup $g(r) \leq CVaR_{1-\gamma}(g(r)) \leq EVaR_{1-\gamma}(g(r))$ as shown above.

Thus, our proposed risk-averse robust learning paradigm is to optimise the upper bound (*i.e.*, EVaR) of the objective in Eq. 5 to fortify models against probabilistic uncertainties. Mathematically, the new paradigm replaces the inner optimisation in Eq. 4 with EVaR, the objective function is:

$$\begin{aligned} \min_{f \in \mathcal{F}} \quad & \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[EVaR_{1-\gamma} \left(\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y); \boldsymbol{\delta} \in \Delta \right) \right], \\ \text{s.t.} \quad & EVaR_\gamma(g(r)) = \inf_{\zeta > 0} \frac{1}{\zeta} \ln \left(\mathbb{E}\left[e^{\zeta g(r)} \right] / \gamma \right), \end{aligned} \tag{7}$$

where γ is user-predefined risk-averse parameter.

Similar to adversarial training, the proposed learning paradigm constitutes a *composite optimisation* problem, i.e., an inner minimisation over ζ to compute EVaR and an outer minimisation to update the hypothesis parameters. However, different to the inner maximisation in adversarial training, the inner minimisation in this case is *convex* w.r.t. ζ regardless of other variables, including the parameters of the hypothesis. Consequently, this property allows the gradient of the inner objective to be obtained in a closed form.

4.2 Theoretical Generalisation Analysis

The inferior generalisation properties of adversarial training in high-dimensional spaces fundamentally constrain its applicability: independent of the solvability of the optimisation procedure during training, it lacks guarantees (or even a likelihood) that the risk-sensitive classifier will exhibit robustness at test-time. We now demonstrate that our risk metric is not subject to this limitation. As shown by Shalev-Shwartz and Ben-David 2014, the generalisation error of a learning algorithm can be probabilistically upper-bounded using statistical learning theory, employing concepts of complexity on the admissible set of hypotheses and loss function. Here we introduce the empirical Rademacher complexity below for a hypothesis class.

Definition 5. The empirical Rademacher complexity for a hypothesis class $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ and sample set $S = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is

$$\mathrm{Rad}_S(\mathcal{F}) := \frac{1}{N} \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}} \sum_{n=1}^{N} \sigma_n f(\mathbf{x}_n) \right], \tag{8}$$

where $\sigma_1, \ldots, \sigma_N$ are independent Rademacher random variables, with the value -1 or +1, each with probability 1/2. Intuitively, it measures the complexity of the class by determining how many different ways functions $f \in \mathcal{F}$ can classify the sample S.

Subsequently, given a neural network hypothesis class $\mathcal{F}$ and loss function class $\mathcal{L}_{\mathcal{F}} \triangleq \{(\mathbf{x}, y) \to \mathcal{L}(\mathbf{x}, y, f) : f \in \mathcal{F}\}$, we can bound the generalisation error of a classifier using the following theorem [Mohri *et al.*, 2018]:

Lemma 1. *Suppose* $\forall \mathbf{x}, y, f : \mathcal{L}(\mathbf{x}, y, f) \in [0, c]$. *For* $m \in \{1, \ldots, M\}$. *Additionally, assume further that the samples* $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ *are i.i.d. from a distribution* $\mathcal{D}$. *Then for any* $\delta \in (0, 1)$, *with probability no less than* $1-\delta$ *the following holds for all* $f \in \mathcal{F}$*:*

$$\begin{aligned} &\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathcal{L}(f(\mathbf{x}), y)] - \mathbb{E}_{P_N}[\mathcal{L}(f(\mathbf{x}), y)] \\ &\leq 2c\,\mathrm{Rad}_S\left(\mathcal{L}_{\mathcal{F}}\right) + 3c\sqrt{\log(2/\delta)/(2N)}, \end{aligned} \tag{9}$$

where $\mathbb{E}_{P_N}[\mathcal{L}(f(\mathbf{x}), y)] = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}\left(f\left(\mathbf{x}_i\right), y_i\right)$

This bound is probabilistic, data-dependent and uniform over all functions in the hypothesis class, $f \in \mathcal{F}$, meaning it holds for all such functions, including those trained on the dataset S. Informally, the empirical risk on the training dataset will be "close" to the true risk (*i.e.*, the difference bounded by the term on the right hand side) with high probability (in the formal sense).

To take advantage of this bound, we need to be able to compute $\mathrm{Rad}_S(\mathcal{L}_{\mathcal{F}})$. The empirical Rademacher complexity (see Eq. 8) of the hypothesis class $\mathrm{Rad}_S(\mathcal{F})$ can be upper bounded [Bartlett *et al.*, 2017; Yin *et al.*, 2019] by an expression $O\left(\log\left(d_{\max}\right)\right)$ that depends on the logarithm of $d_{\max}$ (*i.e.,* the maximum number of nodes in a single layer). Thus we simply need to relate $\mathrm{Rad}_S(\mathcal{L}_{\mathcal{F}})$ to $\mathrm{Rad}_S(\mathcal{F})$.

Lemma 2 (Talagrand's contraction principal)**.** *Let* g *be an* L*-Lipschitz continuous function, and* $\mathcal{F}$ *is a function class. Then,* $\mathrm{Rad}((g \circ \mathcal{F})) \leq L \cdot \mathrm{Rad}(\mathcal{F})$.

Considering the natural risk, if $\mathcal{L}(f(\mathbf{x}), y)$ is Γ-Lipschitz in the first argument, we can use Lemma 2. The lemma gives that $\mathrm{Rad}_S\left(\mathcal{L}_{\mathcal{F}}\right) \leq \Gamma\,\mathrm{Rad}_S\left(\mathcal{F}\right)$. Thus, substituting this inequality into Eq. 9, we have:

$$\begin{aligned} &\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathcal{L}(f(\mathbf{x}), y)] - \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}\left(f\left(\mathbf{x}_i\right), y_i\right) \\ &\leq 2c\Gamma\,\mathrm{Rad}_S\left(\mathcal{F}\right) + 3c\sqrt{\log(2/\delta)/(2N)}, \end{aligned}$$

such that our generalisation error bound scales as $O\left(\log\left(d_{\max}\right)\right)$ (as $\mathrm{Rad}_S(\mathcal{F})$ is $O\left(\log\left(d_{\max}\right)\right)$).

We now introduce an analogous result for our probabilistic risk in Theorem 1, and the detailed proof can be founded in Appendix B. In this case, the empirical risk we will use is the Monte Carlo estimate since this is what we actually compute.

Theorem 1. *Suppose* $\forall \mathbf{x}, y, f : \mathcal{L}(\mathbf{x}, y, f) \in [0, c]$. *For* $m \in \{1, \ldots, M\}$, *define* $S'_m = \left\{\left(\mathbf{x}'_{1,m}, y_1\right), \ldots, \left(\mathbf{x}'_{N,m}, y_N\right)\right\}$, *such that it contains the* m*-th perturbed point* $\mathbf{x}' = \mathbf{x} + \boldsymbol{\delta}$ *from each of the* N *original inputs. Then for any* $\delta \in (0, 1)$, *with probability at least* $1 - \delta$ *the following holds for all* $f \in \mathcal{F}$*:*

$$\begin{aligned} r_{\mathcal{D}}\left(f\right) - R_{N,M}\left(f\right) \leq\ & \frac{2e^{\zeta(c+\Gamma)}}{\zeta}\overline{\mathrm{Rad}_{S'}\left(\mathcal{F}\right)} \\ &+ 3e^{c\zeta}\sqrt{\log(1/\delta)/(2N)}, \end{aligned}$$

where

$$r_{\mathcal{D}}\left(f\right) \triangleq \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[EVaR_{1-\gamma}\left(\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y); \boldsymbol{\delta} \in \Delta\right)\right],$$

$$R_{N,M}\left(f\right) \triangleq \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\zeta}\ln\left(\frac{1}{\gamma M}\sum_{m=1}^{M}\left[e^{\zeta\mathcal{L}(f(\mathbf{x}'_{i,m}), y_i)}\right]\right),$$

$$\overline{\mathrm{Rad}_{S'}(\mathcal{F})} \triangleq \frac{1}{M}\sum_{m=1}^{M}\mathrm{Rad}_{S'_m}(\mathcal{F}).$$

We can see, the generalisation error is upper bounded by an expression that varies as $O\left(\log\left(d_{\max}\right)\right)$. In contrast, for the *adversarial risk*, defined as $\sup_{\boldsymbol{\delta}\in\Delta}\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)$, its empirical Rademacher complexity is **lower bounded** by an expression containing explicit dependence on $\sqrt{d_{in}}$, where d_{in} is the dimension of the input layer of network [Yin *et al.*, 2019]. While this lower bound does not allow us to directly bound the generalisation error using Eq. 9 or compare it with the upper bound of the complexity of the EVaR objective, it does suggest that in high dimensions the adversarial generalisation error can be much greater than the natural and the proposed EVaR risk gaps. This indicates it will typically be difficult to train networks that are adversarially robust at test time for high-dimensional datasets. Our experiments will show that with networks trained with the proposed objective it may be easier to get narrower train-test gaps for high-dimensional datasets compared with an adversarial objective.

4.3 Generalising to Distributional Robustness via Stochastic Search

To achieve more risk sensitivity, we compute the EVaR risk based on the worst-case scenarios for perturbation distributions. To this end, we enable it to generalise to distribution robustness with *adaption*. These perturbations are not static; they are constantly adjusted before any model updates occur in the backward pass. The uncertainty variable (η in Eq. 6) can be adapted using a rule such as gradient descent *w.r.t.* the inner-loop objective function for a fixed number of steps.

Assuming that $p(\boldsymbol{x}; \boldsymbol{\delta})$ is the continuous probability density around the input data $\boldsymbol{x}$, and its support is contained in Δ, then we rewrite the risk-averse robust learning framework (Eq. 7) in the form of distributional robustness by minimising the Entropic-VaR of the loss function $\mathcal{L}(\cdot)$ subject to perturbation δ following uncertain perturbation distributions as:

$$\begin{aligned} &\min_{f\in\mathcal{F}}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\sup_{p(\boldsymbol{\delta})\in\mathcal{P}} EVaR_{1-\gamma}\left(\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y); \boldsymbol{\delta}\right)\right] \\ &s.t.\ EVaR_{\gamma}(\mathcal{L}; \boldsymbol{\delta}) = \inf_{\zeta>0}\frac{1}{\zeta}\ln\left(\mathbb{E}_{p(\boldsymbol{\delta})}\left[e^{\zeta\mathcal{L}(\boldsymbol{\delta})}\right] / \gamma\right). \end{aligned} \tag{10}$$

Equivalently,

$$\min_{f\in\mathcal{F}}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\inf_{\zeta>0}\frac{1}{\zeta}\ln\left(\sup_{p(\boldsymbol{\delta})\in\mathcal{P}}\frac{\mathbb{E}_{p(\boldsymbol{\delta})}\left[e^{\zeta\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y)}\right]}{1-\gamma}\right)\right]. \tag{11}$$

Intuitively, we propose using an adaptive stochastic search to address the inherent stochastic nature of perturbations. The EVaR containing perturbation $\boldsymbol{\delta}$ in the inner loop is computed with respect to uncertainty distributions, and $G(z)$ can be of any functional form, i.e., loss function of a neural network. We then define a sampling distribution for these perturbations using the exponential family of densities defined below.

Definition 6. A family $\{p(z;\eta) : \eta \in \Theta\}$ is an exponential family of densities if it satisfies:

$$p(z;\eta) = h(z) \exp\left(\eta^\top T(z) - \phi(\eta)\right), \tag{12}$$

where $T(z) = [T_1(z), T_2(z), \ldots, T_d(z)]^\top$ is the vector of sufficient statistics, $\phi(\eta) = \ln\left\{\int \exp\left(\eta^\top T(z)\right) dz\right\}$ is a normalisation factor that ensures $p(z;\eta)$ to be a pdf. $\Theta = \{\eta : |\phi(\eta)| < \infty\}$ is the natural parameter space with a nonempty interior and $\eta = [\eta_1, \eta_2, \ldots, \eta_d]^\top$ is the vector of natural parameters.

The inner loop in Eq. 11 is performed by maximising $EVaR_{1-\gamma}(\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y); \boldsymbol{\delta})$ with $p(\boldsymbol{\delta}) = p(\boldsymbol{\delta}, \eta) \in \mathcal{P}$ with respect to the natural parameters:

$$\begin{aligned}
\eta^* &= \underset{\eta\in\Theta}{\operatorname{argmax}}\, EVaR_{1-\gamma}(\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}), y); \boldsymbol{\delta}), \\
&= \underset{\eta\in\Theta}{\operatorname{argmax}} \inf_{\zeta>0} \frac{1}{\zeta} \ln\left(\mathbb{E}_{p(\boldsymbol{\delta},\eta)}\left[e^{\zeta\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}),y)}\right]/(1-\gamma)\right), \\
&\propto \underset{\eta\in\Theta}{\operatorname{argmax}}\, \mathbb{E}_{p(\boldsymbol{\delta},\eta)}\left[e^{\zeta\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}),y)}\right].
\end{aligned} \tag{13}$$

With adaptive stochastic search, a shape function is always introduced $S : \mathbb{R} \to \mathbb{R}^+$ which allows for different weighing schemes of the cost levels, leading to different optimisation behaviours [Ollivier *et al.*, 2017]. Then, the problem is transformed into:

$$\eta^* = \underset{\eta\in\Theta}{\operatorname{argmax}}\, \mathbb{E}_{p(\boldsymbol{\delta},\eta)}\left[S\left(e^{\zeta\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}),y)}\right)\right] = \underset{\eta\in\Theta}{\operatorname{argmax}}\, \mathcal{J}(\eta), \tag{14}$$

where $S(x)$ is non-decreasing in x and bounded from above and below for bounded x (see Appendix C.1). Finally, we can obtain a scale-free gradient in a closed form (see Appendix A.2):

$$\nabla_\eta \mathcal{J}(\eta) = \frac{\mathbb{E}_{p(\boldsymbol{\delta},\eta)}\left[S\left(e^{\zeta\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}),y)}\right)\left(T(\boldsymbol{\delta}) - \nabla_\eta \phi(\eta)\right)\right]}{\mathbb{E}_{p(\boldsymbol{\delta},\eta)}\left[S\left(e^{\zeta\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}),y)}\right)\right]}. \tag{15}$$

With the analytical expression of the gradient we are ready to use a gradient-based approach to update the parameter η on an epoch basis during the training process. In practice, we approximate the inner expectation w.r.t. updated perturbation distribution in Eq. 15 with M Monte Carlo (MC) samples, and perform K steps of gradient ascent on δ to solve the inner problem. After obtaining the optimal parameters of the perturbation distribution, we use the adversarial distribution to update model parameters $\boldsymbol{\theta}$ through risk-averse robust learning. While any probability function of the exponential family will work, in this study we sample perturbations from a Gaussian distribution. The full algorithm is summarised in **Algorithm 1** in Appendix C.

Data	Algorithm	Evaluation Metric (%)				
		Nat. Acc.	Adv. Acc.	Aug. Acc.	Prob. Acc. $\gamma = 0.01$	Prob. Acc. $\gamma = 0.1$
MNIST	ERM	99.88/98.91	0.24/0.32	98.17/97.10	96.37/96.74	98.65/98.30
	FGSM	97.39/97.05	0.11/0.13	97.49/97.49	96.42/95.90	96.79/96.39
	PGD	99.33/98.35	96.38/**93.5**	99.27/98.28	98.20/97.42	98.33/98.07
	PRASS(0.01)	99.98/99.28	8.37/7.88	99.97/99.27	98.89/**98.37**	99.99/98.23
	PRASS(0.1)	99.99/**99.43**	3.98/3.70	99.98/ **99.35**	99.24/98.21	99.99/**98.41**
CIFAR-10	ERM	96.48/91.58	0.11/0.03	95.81/90.11	87.01/78.66	92.36/85.73
	FGSM	89.08/81.26	3.93/0.09	88.92/80.76	87.61/79.61	88.21/80.22
	PGD	95.03/80.31	75.23/**44.49**	94.93/80.28	94.11/78.32	94.48/79.24
	PRASS(0.01)	99.31/91.99	6.31/5.09	99.16/**91.55**	98.46/**87.01**	97.23/89.07
	PRASS(0.1)	99.20/**92.21**	5.23/4.27	99.14/91.36	97.08/86.65	98.04/**90.51**
CIFAR-100	ERM	96.96/65.52	0.01/0.01	88.35/59.09	80.14/50.11	74.72/45.30
	FGSM	90.61/49.19	3.99/0.42	89.03/58.68	76.01/45.67	80.65/49.85
	PGD	96.26/51.75	66.35/**17.04**	96.22/51.53	95.28/49.88	95.73/50.32
	PRASS(0.01)	98.44/69.48	6.11/4.32	98.31/**69.10**	96.12/**64.51**	96.94/59.71
	PRASS(0.1)	98.58/**70.06**	3.15/2.69	98.44/69.95	95.45/65.12	97.16/**60.24**

Table 1: Train/test set evaluations of different networks on MNIST and CIFAR. The best test set performance for each evaluation metric is highlighted in **Bold**. Values in bracket denote γ values.

5 Experiments

Experiment Setup. We conduct an extensive evaluation of the risk-averse robust learning method on three datasets: MNIST, CIFAR-10 and CIFAR-100. For MNIST, we adopt a ReLU network architecture with two convolutional layers, while for CIFAR-10 and CIFAR-100, we utilise an 18-layer residual network architecture. Moreover, the uncertainty set under consideration is a perturbation set, defined as $\Delta = \left\{\delta \in \mathbb{R}^d : \|\delta\|_\infty \leq \epsilon\right\}$, situated within a Gaussian distribution set $p(\delta) \in \mathcal{P}$. We set $\epsilon = 0.3$ for MNIST and $\epsilon = 8/255$ for CIFAR-10 and CIFAR-100. Full details are provided in Appendix C. All the experiments are executed on a system with a 32-Core AMD EPYC 7452 CPU and an NVIDIA A100 40GB GPU.

Evaluation Metrics. To evaluate the performance of the algorithms, we record the natural accuracy on the test set and the empirical robust accuracies of each algorithm. The latter is quantified on perturbed samples for each dataset, assessed via two distinct ways: **(i)** *Empirical Augmented Accuracy*: For each data point, we randomly draw 200 perturbations around inputs, subsequently recording the average accuracy over these perturbed samples, denoted as *"Aug. Acc."*. **(ii)** *Empirical Probabilistic Accuracy*: To assess the models' probabilistic accuracy, we compute the empirical probabilistic accuracy, signifying the proportion of samples that are empirically probabilistically robust with a tolerance level γ. The mathematical expression is defined as $Prob.\ Acc.(\gamma) = \mathbb{1}[\mathcal{P}_{\boldsymbol{\delta}\sim\Delta}[f(\mathbf{x}+\boldsymbol{\delta}) \neq y] < \gamma]$ [Robey *et al.*, 2022] and, notably, the same number of perturbations are utilised to calculate the empirical probabilistic accuracy, denoted as *"Prob. Acc.(γ)"*.

5.1 Empirical Generalisation Error

As discussed, training neural networks to achieve high test-time adversarial accuracy on high-dimensional datasets is challenging. However, as implied by our analysis in Section 4.2, the generalisation gap for our EVaR based method should be closer to that obtained when using empirical risk. To empirically investigate this, we use four different methods

Algorithm	Test Acc. (%)			Prob. Acc. (%)			Cert. Acc. (%)		
	Nat.	Adv.	Aug.	γ=0.2	γ=0.1	γ=0.05	γ=0.2	γ=0.1	γ=0.05
ERM	98.91	0.32	97.10	98.83	98.30	97.31	91.53	90.42	87.06
FGSM	97.05	0.13	97.49	96.86	96.39	96.03	92.64	91.93	90.78
PGD	98.35	93.50	98.28	98.15	98.07	97.86	93.67	90.41	88.01
TRADES	99.04	94.29	99.04	98.70	98.56	98.24	97.74	97.51	97.19
DALE	99.18	**99.14**	93.82	98.90	98.71	98.50	98.67	98.50	98.26
TERM	98.96	11.26	98.57	97.95	97.26	96.50	97.87	97.13	96.29
PRoL(γ=0.01)	99.18	3.87	98.34	99.08	98.77	98.46	98.35	98.00	97.25
PRoL(γ=0.1)	98.90	4.15	98.86	98.48	98.16	97.91	98.07	97.74	96.87
PRASS(γ=0.01)	99.28	7.88	99.27	99.12	98.23	**98.95**	**99.05**	98.17	98.37
PRASS(γ=0.1)	**99.43**	3.70	**99.35**	**99.13**	**98.91**	98.76	99.03	**98.78**	**98.40**

Table 2: MNIST - PRASS vs. baselines in terms of empirical / certified robust accuracy with a confidence level of $1 - 10^{-10}$.

Algorithm	Test Acc. (%)			Prob. Acc. (%)			Cert. Acc. (%)		
	Nat.	Adv.	Aug.	γ=0.2	γ=0.1	γ=0.05	γ=0.2	γ=0.1	γ=0.05
ERM	91.58	0.03	90.11	86.09	85.73	82.34	83.64	80.83	76.84
FGSM	81.26	0.09	80.76	81.04	80.22	79.81	74.83	73.77	71.93
PGD	80.32	44.49	80.28	79.60	79.24	78.97	74.82	73.51	72.08
TRADES	74.77	**45.58**	74.62	73.77	73.53	73.16	70.37	69.38	68.03
DALE	82.03	39.67	81.92	81.36	81.09	80.75	75.26	73.02	72.53
TERM	89.46	0.01	86.32	83.05	80.94	79.06	77.83	73.76	70.25
PRoL(γ=0.01)	88.47	1.07	87.13	84.23	81.18	78.67	81.33	76.82	74.69
PRoL(γ=0.1)	90.00	0.02	90.05	86.34	84.12	82.07	84.08	81.10	79.28
PRASS(γ=0.01)	91.99	5.09	**91.55**	90.98	89.95	**89.07**	**91.33**	88.17	**87.90**
PRASS(γ=0.1)	**92.21**	4.27	91.36	**91.37**	**90.51**	88.52	89.38	**88.42**	85.48

Table 3: CIFAR-10 - PRASS vs. baselines in terms of empirical / certified robust accuracy with a confidence level of $1 - 10^{-10}$.

Algorithm	Test Acc. (%)			Prob. Acc. (%)			Cert. Acc. (%)		
	Nat.	Adv.	Aug.	γ=0.2	γ=0.1	γ=0.05	γ=0.2	γ=0.1	γ=0.05
ERM	65.52	0.01	59.09	53.19	50.11	45.30	42.67	38.90	33.94
FGSM	49.19	3.99	58.68	52.96	49.85	47.32	47.66	46.36	44.44
PGD	51.75	17.04	51.53	50.69	50.32	50.05	49.80	49.23	48.36
TRADES	48.87	**22.94**	48.79	48.12	47.75	47.35	47.15	46.74	45.92
DALE	52.94	22.58	52.82	52.07	51.66	51.33	50.66	50.08	48.86
TERM	48.29	3.05	48.02	44.87	43.11	41.41	41.44	38.60	35.06
PRoL(γ=0.01)	66.78	2.1	66.54	62.95	60.97	59.29	51.13	49.44	45.83
PRoL(γ=0.1)	67.13	0.01	67.47	63.53	61.50	59.78	53.09	50.43	46.79
PRASS(γ=0.01)	69.48	4.32	69.10	66.23	64.51	62.97	57.31	54.50	51.04
PRASS(γ=0.1)	**70.06**	2.69	**69.95**	**67.08**	**65.12**	**63.35**	**58.80**	**56.35**	**52.80**

Table 4: CIFAR-100 - PRASS vs. baselines in terms of empirical / certified robust accuracy with a confidence level of $1 - 10^{-10}$.

to train networks: standard training (*i.e.*, ERM as defined in Eq. 2), adversarial training techniques (*i.e.*, FGSM and PGD with 7 gradient steps) and the proposed training approaches (*i.e.*, PRASS) with tolerance levels $\gamma = 0.01$ and $\gamma = 0.1$. In accordance with these methods, we evaluate the performance of the networks using natural accuracy, the previously mentioned probabilistic robust accuracy, and adversarial accuracy by using a 20-step PGD adversary.

The results are presented in Table 1. These findings corroborate our theoretical analysis in Section 4.2: **1)** For the low-dimensional MNIST dataset, all training methodologies exhibit exceptional generalisation performance, with train-test generalisation gaps being quite small, around 3% for all evaluation metrics, *i.e.,* clean accuracy, adversarial accuracy, and probabilistic accuracy. **2)** For the high-dimensional CIFAR-10 dataset, ERM and PRASS show fairly small generalisation gaps, about 8%. Conversely, adversarial training approaches, particularly on the PGD-trained network, show a markedly larger gap, nearing 30%. **3)** For CIFAR-100, which is even more challenging due to more classes, the generalisation gap increases across methods. Still, PRASS outperforms the PGD-trained network by a significant margin (around 20%). This underscores a clear limitation of adversarial training compared to our proposed risk-averse training.

5.2 Effectiveness of PRASS

Baselines & Evaluation Metrics. To validate the efficacy of PRASS, we consider baselines including ERM, TRADES [Zhang *et al.*, 2019], DALE [Robey *et al.*, 2021], TERM [Li *et al.*, 2023] and PRL [Robey *et al.*, 2022]. Besides the aforementioned metric, we also adopt a certified probabilistically robust accuracy with a tolerance level γ and confidence level $\delta = 10^{-10}$, computed with PRoA [Zhang *et al.*, 2022], denoted as *"Cert. Acc."*. Note that we consider probabilistically robust accuracy, i.e., *Cert. Acc.*(γ) and *Prob. Acc.*(γ), as the key metrics for assessing the level of probabilistic robustness that models trained with various algorithms can achieve. The highest and second highest performances are highlighted in **Bold** and Underlined. The most critical metric is emphasised in blue.

As shown in Tables 2, 3 & 4, PRASS shows significant improvements in risk reduction for random perturbations. For both empirical and certifiable probabilistic metrics, our method consistently showcases the best test-set performance across all risk levels. This improvement is most conspicuous on CIFAR-10 and CIFAR-100, the higher-dimensional datasets, and, to some degree, it aligns with our generalisation result. Another noteworthy observation is that our PRASS method does not suffer the same limitation as adversarial training does; in fact, PRASS's natural accuracy surpasses that of ERM by a margin of around 1% and 50%, respectively on these data. As expected, PRASS does not perform well on adversarial accuracy, given that its focus lies on risk aversion as opposed to explicit adversaries. Finally, we observe that all adversarial training approaches do not contribute positively to improving probabilistic robustness, despite achieving consistently commendable test-set adversarial performance. This indicates that adversarial risk is neither a *relevant* nor *optimal* objective for addressing risk reversion.

6 Conclusion

In this paper, motivated by scenarios where models' threats are not adversarially generated but arise probabilistically, we introduced a new framework called risk-aversion robust learning. Instead of focusing on the worst-case robustness, in this framework, robustness is enforced with high probability over perturbations by updating the perturbation distribution and minimising the upper bound of the γ-essential supremum across updated distributions. Our exploration of the practical and theoretical aspects of this framework led to a new algorithm that achieves superior generalisation performance compared to adversarial training and effectively enforces probabilistic robustness in practice.

References

[Ahmadi *et al.*, 2022] Mohamadreza Ahmadi, Xiaobin Xiong, and Aaron D. Ames. Risk-averse control via cvar barrier functions: Application to bipedal robot locomotion. *IEEE Control Systems Letters*, 6:878–883, 2022.

[Awasthi *et al.*, 2020] Pranjal Awasthi, Himanshu Jain, Ankit Singh Rawat, and Aravindan Vijayaraghavan. Adversarial robustness via robust low rank representations. In *Proceedings of the 34th Conference on Neural Information Processing Systems*, volume 33, pages 11391–11403, 2020.

[Balunovic and Vechev, 2020] Mislav Balunovic and Martin T. Vechev. Adversarial training and provable defenses: Bridging the gap. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[Bartlett *et al.*, 2017] Peter L. Bartlett, Dylan J. Foster, and Matus Telgarsky. Spectrally-normalized margin bounds for neural networks. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, pages 6240–6249, 2017.

[Cohen *et al.*, 2019] Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 1310–1320, 2019.

[Croce and Hein, 2020] Francesco Croce and Matthias Hein. Provable robustness against all adversarial l_p-perturbations for $p\geq 1$. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[Croce *et al.*, 2019] Francesco Croce, Maksym Andriushchenko, and Matthias Hein. Provable robustness of relu networks via maximization of linear regions. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, volume 89, pages 2057–2066, 2019.

[Dong *et al.*, 2020] Yinpeng Dong, Zhijie Deng, Tianyu Pang, Jun Zhu, and Hang Su. Proceedings of the 34th conference on neural information processing systems. In *Advances in Neural Information Processing Systems*, pages 8270–8283, 2020.

[Fan and Li, 2021] Jiameng Fan and Wenchao Li. Adversarial training and provable robustness: A tale of two objectives. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, volume 35, pages 7367–7376, 2021.

[Gowal *et al.*, 2019] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Arthur Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, pages 4841–4850, 2019.

[Huang *et al.*, 2017] Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu. Safety verification of deep neural networks. In *Proceedings of the 29th International Conference on Computer Aided Verification*, volume 10426, pages 3–29. Springer, 2017.

[Huang *et al.*, 2020] Xiaowei Huang, Daniel Kroening, Wenjie Ruan, James Sharp, Youcheng Sun, Emese Thamo, Min Wu, and Xinping Yi. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37:100270, 2020.

[ISO, 2014] IEC ISO. Iso/iec guide 51: Safety aspects-guidelines for their inclusion in standards. *Geneva, Switzerland*, 2014.

[Jin *et al.*, 2022] Gaojie Jin, Xinping Yi, Wei Huang, Sven Schewe, and Xiaowei Huang. Enhancing adversarial training with second-order statistics of weights. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15252–15262, 2022.

[Lécuyer *et al.*, 2019] Mathias Lécuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *IEEE Symposium on Security and Privacy*, pages 656–672, 2019.

[Li *et al.*, 2023] Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. On tilted losses in machine learning: Theory and applications. *Journal of Machine Learning Research*, 24:142:1–142:79, 2023.

[Lyu *et al.*, 2021] Zhaoyang Lyu, Minghao Guo, Tong Wu, Guodong Xu, Kehuan Zhang, and Dahua Lin. Towards evaluating and training verifiably robust neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 4308–4317, 2021.

[Madry *et al.*, 2018] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

[Majumdar and Pavone, 2017] Anirudha Majumdar and Marco Pavone. How should a robot assess risk? towards an axiomatic theory of risk in robotics. In *Proceedings of the 18th International Symposium on Robotics Research*, volume 10, pages 75–84, 2017.

[Mirman *et al.*, 2018] Matthew Mirman, Timon Gehr, and Martin T. Vechev. Differentiable abstract interpretation for provably robust neural networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 3575–3583, 2018.

[Mohri and Rostamizadeh, 2008] Mehryar Mohri and Afshin Rostamizadeh. Rademacher complexity bounds for non-i.i.d. processes. In *Proceedings of Conference on Neural Information Processing Systems*, pages 1097–1104, 2008.

[Mohri *et al.*, 2018] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive computation and machine learning. 2018.

[Mu *et al.*, 2021] Ronghui Mu, Wenjie Ruan, Leandro Soriano Marcolino, and Qiang Ni. Sparse adversarial video attacks with spatial transformations. In *Proceedings of the 32nd British Machine Vision Conference*, page 101, 2021.

[Ollivier *et al.*, 2017] Yann Ollivier, Ludovic Arnold, Anne Auger, and Nikolaus Hansen. Information-geometric optimization algorithms: A unifying picture via invariance principles. *Journal of Machine Learning Research*, 18:18:1–18:65, 2017.

[Robey *et al.*, 2021] Alexander Robey, Luiz F. O. Chamon, George J. Pappas, Hamed Hassani, and Alejandro Ribeiro. Adversarial robustness with semi-infinite constrained learning. In *Proceedings of the 35th Conference on Neural Information Processing Systems*, pages 6198–6215, 2021.

[Robey *et al.*, 2022] Alexander Robey, Luiz F. O. Chamon, George J. Pappas, and Hamed Hassani. Probabilistically robust learning: Balancing average and worst-case performance. In *Proceedings of International Conference on Machine Learning*, volume 162, pages 18667–18686, 2022.

[Rockafellar *et al.*, 2000] R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2000.

[Salman *et al.*, 2019] Hadi Salman, Jerry Li, Ilya P. Razenshteyn, Pengchuan Zhang, Huan Zhang, Sébastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *Proceedings of the 33rd Conference on Neural Information Processing Systems*, pages 11289–11300, 2019.

[Schmidt *et al.*, 2018] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Proceedings of the 32nd Conference on Neural Information Processing Systems*, pages 5019–5031, 2018.

[Shalev-Shwartz and Ben-David, 2014] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.

[Shi *et al.*, 2021] Zhouxing Shi, Yihan Wang, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh. Fast certified robust training via better initialization and shorter warmup. *CoRR*, 2021.

[Tjeng *et al.*, 2019] Vincent Tjeng, Kai Yuanqing Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

[Wang *et al.*, 2021] Fu Wang, Yanghao Zhang, Yanbin Zheng, and Wenjie Ruan. Gradient-guided dynamic efficient adversarial training. *CoRR*, 2021.

[Wong *et al.*, 2020] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[Yin *et al.*, 2019] Dong Yin, Kannan Ramchandran, and Peter L. Bartlett. Rademacher complexity for adversarially robust generalization. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 7085–7094, 2019.

[Zhai *et al.*, 2020] Runtian Zhai, Chen Dan, Di He, Huan Zhang, Boqing Gong, Pradeep Ravikumar, Cho-Jui Hsieh, and Liwei Wang. MACER: attack-free and scalable robust training via maximizing certified radius. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[Zhang *et al.*, 2019] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 7472–7482, 2019.

[Zhang *et al.*, 2020a] Huan Zhang, Hongge Chen, Chaowei Xiao, Sven Gowal, Robert Stanforth, Bo Li, Duane S. Boning, and Cho-Jui Hsieh. Towards stable and efficient training of verifiably robust neural networks. In *Proceedings of the 8th International Conference on Learning Representations*, 2020.

[Zhang *et al.*, 2020b] Tianle Zhang, Muzhou Hou, Tao Zhou, Zhaode Liu, Weirong Cheng, and Yangjin Cheng. Land-use classification via ensemble dropout information discriminative extreme learning machine based on deep convolution feature. *Computer Science and Information Systems*, 17(2):427–443, 2020.

[Zhang *et al.*, 2021] Bohang Zhang, Tianle Cai, Zhou Lu, Di He, and Liwei Wang. Towards certifying l-infinity robustness using neural networks with l-inf-dist neurons. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 12368–12379, 2021.

[Zhang *et al.*, 2022] Tianle Zhang, Wenjie Ruan, and Jonathan E. Fieldsend. Proa: A probabilistic robustness assessment against functional perturbations. In *Proceedings of Machine Learning and Knowledge Discovery in Databases - European Conference*, volume 13715, pages 154–170, 2022.

[Zhou and Hu, 2014] Enlu Zhou and Jiaqiao Hu. Gradient-based adaptive stochastic search for non-differentiable optimization. *IEEE Transactions on Automatic Control*, 59(7):1818–1832, 2014.