Provably Adversarially Robust Nearest Prototype Classifiers

Václav Voráček [1] **Matthias Hein** [1]

Abstract

Nearest prototype classifiers (NPCs) assign to each input point the label of the nearest prototype with respect to a chosen distance metric. A direct advantage of NPCs is that the decisions are interpretable. Previous work could provide lower bounds on the minimal adversarial perturbation in the ℓ_p-threat model when using the same ℓ_p-distance for the NPCs. In this paper we provide a complete discussion on the complexity when using ℓ_p-distances for decision and ℓ_q-threat models for certification for $p, q \in \{1, 2, \infty\}$. In particular we provide scalable algorithms for the *exact* computation of the minimal adversarial perturbation when using ℓ_2-distance and improved lower bounds in other cases. Using efficient improved lower bounds we train our **P**rovably adversarially robust **NPC** (PNPC), for MNIST which have better ℓ_2-robustness guarantees than neural networks. Additionally, we show up to our knowledge the first certification results w.r.t. to the LPIPS perceptual metric which has been argued to be a more realistic threat model for image classification than ℓ_p-balls. Our PNPC has on CIFAR10 higher certified robust accuracy than the empirical robust accuracy reported in (Laidlaw et al., 2021). The code is available in our repository.

1. Introduction

The vulnerability of neural networks against adversarial manipulations (Szegedy et al., 2014; Goodfellow et al., 2015) is a major problem for their real world deployment in safety critical systems such as autonomous driving and medical applications. However, the problem is not restricted to neural networks as it has been shown that basically all machine learning algorithms are vulnerable to adversarial perturbations e.g. nearest neighbor methods (NN) (Wang et al., 2018), kernel SVMs (Xu et al., 2009; Biggio et al., 2013; Russu et al., 2016; Hein & Andriushchenko, 2017), decision trees (Papernot et al., 2016; Bertsimas et al., 2018; Chen et al., 2019; Andriushchenko & Hein, 2019). In the area of neural networks this lead to an arm's race between novel empirical defenses and attacks and even initially promising defenses were broken later on (Athalye et al., 2018). This still happens for papers published at top machine learning conferences (Tramer et al., 2020; Croce & Hein, 2020a) despite more reliable attacks for adversarial robustness evaluation (Croce & Hein, 2020b) and guidelines (Carlini et al., 2019) being available.

Thus classifiers with provable adversarial robustness guarantees are highly desirable. For neural networks computation of the exact minimal perturbation turns out to be restricted to very small networks (Tjeng & Tedrake, 2017). Instead one derives either deterministic (Hein & Andriushchenko, 2017; Wong & Kolter, 2018; Gowal et al., 2018; Mirman et al., 2018; Zhang et al., 2020; Lee et al., 2020; Huang et al., 2021; Leino et al., 2021) or probabilistic guarantees (Cohen et al., 2019; Jeong et al., 2021) on the robust accuracy. We refer to (Li et al., 2020) for a recent overview. While provable adversarial robustness has been studied extensively for neural networks, the literature for standard classifiers is scarce, e.g. decision trees (Bertsimas et al., 2018), boosted decision stumps and trees (Chen et al., 2019; Andriushchenko & Hein, 2019), and nearest neighbour (Wang et al., 2018; 2019) and nearest prototype classifiers (NPC) (Saralajew et al., 2020). NPC are also known as *Learning Vector Quantization (LVQ)*, see (Kohonen, 1995), and are directly interpretable, can be used for all data where a distance function is available and have the advantage compared to a nearest neighbour classifier that the prototypes can be learned and thus they are more efficient and achieve typically better generalization performance. Moreover, NPC have a maximum margin nature (Crammer et al., 2003) and (Saralajew et al., 2020) showed recently how to derive lower bounds on the minimal adversarial perturbation which in turn yield lower bounds on the robust accuracy. (Wang et al., 2019) have shown how to compute the minimal adversarial perturbation for nearest neighbor classifiers using the ℓ_2-distance which applies to NPC as well.

Contributions: we show that the results of (Saralajew et al., 2020) can be improved in various ways leading to our PNPC

[1] University of Tübingen, Germany. Correspondence to: Václav Voráček <vaclav.voracek@uni-tuebingen.de>, Matthias Hein <matthias.hein@uni-tuebingen.de>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

which perform better both in clean and robust accuracy.

A) We generalize the lower bounds on the minimal adversarial perturbation (Saralajew et al., 2020) provided for distances induced by semi-norms to general semi-metrics, thus improving significantly over standard ℓ_p-based certification. The original proof of (Saralajew et al., 2020) used the absolute homogenity of semi-norms; thus, it do not generalize to semi-metrics.

B) For NPC using the ℓ_2-distance we show that the lower bounds of (Wang et al., 2019) can be quickly evaluated so that training with them is feasible and show that these bounds improve the ones of (Saralajew et al., 2020). Moreover, we improve the certification of (Wang et al., 2019) by integrating that the domain in image classification is $[0,1]^d$. For MNIST our ℓ_2-PNPC has the best ℓ_2-robust accuracy even outperforming randomized smoothing for large radii. Moreover, we show how to certify exactly ℓ_1- and ℓ_∞-robustness for ℓ_2-NPC and in this way can certify multiple-norm robustness and show that our ℓ_2-PNPC outperforms the multiple-norm robustness guarantees of (Croce & Hein, 2020a).

C) For the ℓ_1-and ℓ_∞-NPC we provide novel lower bounds and analyze their complexity. For ℓ_∞-NPCs we thus improve over the bounds given in (Saralajew et al., 2020).

D) As the ℓ_p-distances are not suited for image classification tasks, we use a neural perceptual metric (LPIPS) (Zhang et al., 2018) as a semi-metric for the NPC and provide robustness guarantees in the perceptual metric. We improve both in terms of clean and certified robust accuracy over the clean and empirical robust accuracy of the adversarially trained ResNet 50 of (Laidlaw et al., 2021)

2. Provably Robust NPC Classifiers

Nearest prototype classifiers require for a given input space $\mathcal{X}$ only a (semi)-metric. To compare with previous work, we introduce also a (semi)-norm, which requires a vector-space structure; thus, assuming the existence of a norm is a stronger assumption than the assumption of the existence of a metric.

Definition 2.1. A mapping $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a semi-metric if the following properties holds for any $x, y, z \in \mathcal{X}$:

- $d(x, y) \geq 0$ (non-negativity)
- $d(x, y) = d(y, x)$ (symmetry)
- $d(x, y) \leq d(x, z) + d(z, y)$ (triangle inequality)

If we further require that $d(x, y) = 0 \implies x = y$, then the semi-metric becomes a metric.

Definition 2.2. A mapping $\|\cdot\| : \mathcal{X} \to \mathbb{R}$ is a semi-norm if the following properties holds for any $x, y \in \mathcal{X}$, $\alpha \in \mathbb{R}$:

- $\|x\| \geq 0$ (non-negativity)
- $\|\alpha x\| = |\alpha| \, \|x\|$ (absolute homogeneity)
- $\|x + y\| \leq \|x\| + \|y\|$ (triangle inequality)

If we further require $\|x\| = 0 \implies x = \mathbf{0}$, then the semi-norm becomes a norm.

Note that any (semi)-norm $\|x\|$ induces a (semi)-metric d with $d(x, y) = \|x - y\|$.

We denote by $(w_i)_I$ the set of prototypes. Each prototype is assigned to one class. Then $z \in \mathbb{R}^d$ is classified as

$$f(z) = \underset{y=1,\dots,K}{\arg\min} \min_{i \in I_y} d(z, w_i),$$

where I_y are the prototypes of class y. A nearest neighbor classifier (1NN) can also be understood as NPC where one uses the training set as prototypes and thus are not learned. However, by training prototypes one can achieve better classification performance, and also robustness, see Table 5, with less prototypes meaning that NPC are significantly more efficient than 1NN. We note that the classification for a point z with label y is correct if

$$\min_{i \in I_y} d(z, w_i) - \min_{j \in I_y^c} d(z, w_j) < 0,$$

where I_y^c is the set of all prototypes not belonging to class y (the complement of I_y in I).

2.1. Provable robustness guarantees for semi-metrics

Next we define the minimal adversarial perturbation of a point z for a semi-metric on $\mathcal{X}$, that is the radius r of the smallest ball $B_d(z, r) = \{x \in \mathcal{X} \,|\, d(x, z) \leq r\}$ around z such at least one point in $B_d(z, r)$ is classified differently than is z. If a point z is misclassified then we define the minimal adversarial perturbation to be zero. We assume that there is a non-empty set of prototypes for every class; thus, there always exists an adversarial example.

Definition 2.3. The **minimal adversarial perturbation** $\epsilon_d(z)$ of $z \in \mathcal{X}$ of a NPC using semi-metric d is defined as

$$\epsilon_d(z) = \min\{r \,|\, \max_{x \in B_d(z,r)} \big(\min_{i \in I_y} d(x, w_i) - \min_{j \in I_y^c} d(x, w_j)\big) \geq 0\}$$

If $\min_{i \in I_y} d(z, w_i) - \min_{j \in I_y^c} d(z, w_j) \geq 0$ then we set $\epsilon_d(z) = 0$.

In (Saralajew et al., 2020) they derive for semi-norms a lower bound on ϵ_d and in this way get robustness certificates. We generalize this lower bound to semi-metrics which is considerably more general as $\mathcal{X}$ need not be a vector space. It turns out that the only necessary technical requirement for the proof is the triangle inequality. This is unlike (Saralajew et al., 2020), where the proof also required the absolute homogeneity of semi-norms.

Theorem 2.4. *Let $(\mathcal{X}, d)$ be a semi-metric space, then it holds for the minimal adversarial perturbation $\epsilon_d(z)$ of $z \in \mathcal{X}$ with correct label y:*

$$\epsilon_d(z) \geq \max\left\{0, \frac{\min\limits_{j \in I_y^c} d(z, w_j) - \min\limits_{i \in I_y} d(z, w_i)}{2}\right\}.$$

We note that if the semi-metric d can be written as $d(x, y) = \|x - y\|$ for some semi-norm $\|\cdot\|$, then our bound is equal to the one given in (Saralajew et al., 2020)

2.2. The minimal adversarial ℓ_q-perturbation of the ℓ_p-NPC and lower bounds

In this section we derive the minimal adversarial ℓ_q-perturbation for the ℓ_p-PNPC in $\mathbb{R}^d$ where our main interest is $p, q \in \{1, 2, \infty\}$. In contrast to the semi-metric case, here we treat the case where the ℓ_q-metric measuring the size of the adversarial perturbation is different from the ℓ_p-metric used in the NPC. In this section we use the notation

$$B_q(x, r) = \{z \in \mathbb{R}^d \mid \|z - x\|_q \leq r\}.$$

Thus we first define

Definition 2.5. The **minimal adversarial perturbation** $\epsilon_p^q(z)$ of $x \in \mathcal{X} \subset \mathbb{R}^d$ with respect to the ℓ_q-metric for the ℓ_p-NPC is defined as:

$$\begin{aligned} \epsilon_p^q(z)_j = \min_{r \in \mathbb{R}, x \in \mathcal{X}} \quad & r \\ \text{sbj. to:} \quad & \|x - w_i\|_p - \|x - w_j\|_p \geq 0 \\ & x \in B_q(z, r) \end{aligned}$$

If $\min\limits_{i \in I_y} \|x - w_i\|_p - \min\limits_{j \in I_y^c} \|x - w_j\|_p > 0$ we set $\epsilon_p^q(z) = 0$.

The following reformulation of the optimization problem for the computation of the minimal adversarial perturbation $\epsilon_p^q(z)$ allows us to provide a generic and direct way to derive efficiently computable lower bounds on $\epsilon_p^q(z)$. Note that in the following we always integrate the constraint $x \in \mathcal{X}$ as we will see that this significantly improves the guarantees, e.g. when $\mathcal{X} = [0, 1]^d$ in image classification, compared to $\mathcal{X} = \mathbb{R}^d$ as done in (Saralajew et al., 2020; Wang et al., 2019).

Theorem 2.6 (Exact computation of $\epsilon_p^q(z)$). *Let $z \in \mathcal{X} \subset \mathbb{R}^d$ and denote by I_y the index set of prototypes (w_j) of class y and by I_y^c its complement (the index set of prototypes not belonging to class y). Then define for every $j \in I_y^c$:*

$$\begin{aligned} r_p^q(z)_j = \min_{x \in \mathbb{R}^d} \quad & \|x - z\|_q \\ \textit{sbj. to:} \quad & \|x - w_i\|_p - \|x - w_j\|_p \geq 0 \quad \forall\, i \in I_y \\ & x \in \mathcal{X} \end{aligned} \tag{1}$$

Then $\epsilon_p^q(z) = \min\limits_{j \in I_y^c} r_p^q(z)_j$.

		ℓ_q-threat model		
		ℓ_1	ℓ_2	ℓ_∞
ℓ_p-distance	ℓ_1	NP-hard	NP-hard	$O(d\log(d))$
	ℓ_2	$\Theta(d)$	$\Theta(d)$	$\Theta(d)$
	ℓ_∞	$\Theta(d)$	$O(d\log(d))$	$\Theta(d)$

Table 1: Computational complexity of $\rho_p^q(z)_{i,j}$.

		ℓ_q-threat model		
		ℓ_1	ℓ_2	ℓ_∞
ℓ_p-distance	ℓ_1	NP-hard	NP-hard	Poly
	ℓ_2	Poly	Poly	Poly
	ℓ_∞	NP-hard	NP-hard	NP-hard

Table 2: Computational Complexity of $r_p^q(z)$ and $\epsilon_p^q(z)$.

While the corresponding optimization problems are often non-convex, we will see in the following that they are equivalent to convex optimization problems in the case where the ℓ_2-distance is used in the NPC ($p = 2$). Using the formulation of the exact problem as an optimization problem we can now simply derive lower bounds on $\epsilon_p^q(z)$ by relaxing the optimization problem (1).

We consider for this reason the following optimization problems. For $i \in I_y$ and $j \in I_y^c$ we define:

$$\begin{aligned} \rho_p^q(z)_{i,j} = \min_{x \in \mathbb{R}^d} \quad & \|x - z\|_q \\ \text{sbj. to:} \quad & \|x - w_i\|_p - \|x - w_j\|_p \geq 0 \\ & x \in \mathcal{X} \end{aligned} \tag{2}$$

In Theorem 2.7 we show that these simpler problems can often be solved efficiently, although the computation of ϵ_p^q is often intractable, as we show in Theorem 2.8.

Theorem 2.7. *The computational complexities of optimization problems $\rho_p^q(z)_{i,j}$ for $p, q \in \{1, 2, \infty\}$ for $\mathcal{X} = \mathbb{R}^d$ are summarized in Table 1.*

Theorem 2.8. *The computational complexities of optimization problems $r_p^q(z)_j$ in (1) for $p, q \in \{1, 2, \infty\}$ and $\mathcal{X} = [0, 1]^d$ are summarized in Table 2.*

Apart from the known ℓ_2-case (see (Wang et al., 2019)) we show that also ℓ_1-NPC can be certified efficiently for the ℓ_∞-threat model. Because of this theorem it is even more important that at least for the ℓ_∞-NPCs efficient lower bounds are available for all threat models in $q = \{1, 2, \infty\}$. We note that the optimization problem for $r_2^q(z)_j$ in (1) is equivalent to a quadratic program for $q = 2$ and to a linear programs for $q \in \{1, \infty\}$ for both with and without box constraints.

The following lemma shows that (2) can be used to get a lower bound on the minimal adversarial perturbation, and

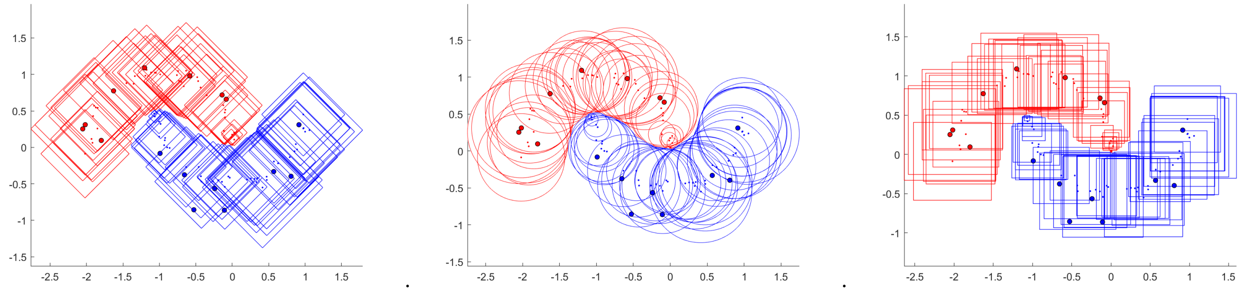


Figure 1: **Illustration of the ℓ_q-minimal adversarial perturbations of a ℓ_2-NPC for a binary classification problem.** The learned prototypes are shown as the larger red resp. blue dots. For each data point we draw the largest ℓ_1-(left), ℓ_2-(middle) and ℓ_∞-(right) ball which is fully classified as the same class. The radii are computed using Alg. 1. Though there is no specific optimization for multiple-norm robustness, ℓ_2-NPC possess non-trivial multiple-norm robustness.

subsequently we show that it improves on the previous bound given in Theorem 2.4 which has been derived by (Saralajew et al., 2020). In particular, this bound can be tight and we show in Table 4 in Section 5 that this happens frequently in practice and thus allows to avoid the significantly more complex problems in (1).

Lemma 2.9. *It holds*

$$\epsilon_p^q(z) \geq \min_{j\in I_c^y} \max_{i\in I_y} \rho_p^q(z)_{i,j}.$$

Moreover, let (j^*, i^*) be the prototype pair in $I_c^y \times I_y$ which realizes the lower bound and denote by x^* the minimizer of $\rho_p^q(z)_{i^*,j^*}$. Then if x^* fulfills

$$\|x^* - w_i\|_p - \|x^* - w_{j^*}\|_p \geq 0 \quad \forall i \in I_y,$$

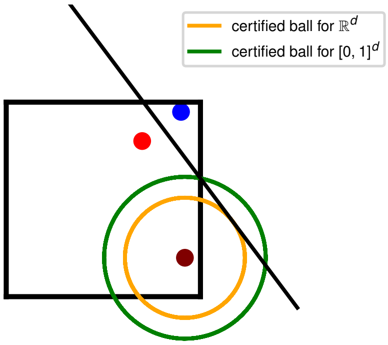
then $\epsilon_q^p(z) = \min_{j\in I_c^y} \max_{i\in I_y} \rho_p^q(z)_{i,j}$.

Theorem 2.10. *The lower bound on $\epsilon_p^p(z)$ of Lemma 2.9 is at least as good as the one of Theorem 2.4. That is,*

$$\begin{aligned}\min_{j\in I_c^y} \max_{i\in I_y} \rho_p^p(z)_{i,j} &\geq \min_{j\in I_c^y} \rho_p^p(z)_{i^*,j} \\ &\geq \max\left\{0, \frac{\min_{j\in I_y^c}\|z-w_j\|_p - \min_{i\in I_y}\|z-w_i\|_p}{2}\right\},\end{aligned}$$

where $i^* \in \arg\min_{i\in I_y} \|z - w_i\|_p$.

In order to be able to use these lower bounds for certified training of our PNPC, their efficient computation is of high importance which we discuss next.



certified ball for $\mathbb{R}^d$
certified ball for $[0, 1]^d$

Figure 2: Illustration for ℓ_2-NPC for two prototypes (red and blue): when taking into account that the data lies in $[0, 1]^d$ we can certify a larger ball than in $\mathbb{R}^d$.

For better intuition we discuss some cases in more detail. The ℓ_2-NPC have a nice geometric descriptions as the set

$$\begin{aligned}&\{z|\, \|z - w_i\|_2 = \|z - w_j\|_2\} \\ =&\{z|\, \langle w_j - w_i, z\rangle + \frac{\|w_i\|_2^2 - \|w_j\|_2^2}{2} = 0\}\end{aligned}$$

is a hyperplane. Thus the computation of $\rho_2^q(z)_{i,j}$ for $\mathcal{X} = \mathbb{R}^d$ corresponds to the computation of the ℓ_q-distance of a point to a hyperplane:

$$\rho_2^q(z)_{i,j} = \frac{\|z - w_j\|_2^2 - \|z - w_i\|_2^2}{2\,\|w_i - w_j\|_{q^*}},$$

where q^* denotes the dual norm of q. This has also been derived in (Wang et al., 2019). As illustration how the constraints $\mathcal{X} = [0, 1]^d$, e.g. in image classification, improve the certificates, we show in Figure 2 the ball which can be certified in $\mathbb{R}^d$ resp. $[0, 1]^d$.

2.3. How to do the certification efficiently

Table 1 shows that $\rho_p^q(z)_{i,j}$ can be computed efficiently or even given in closed form except for the two cases when $(p, q) \in \{(1, 1), (1, 2)\}$. However, that would still mean that the lower bound of Lemma 2.9

$$\epsilon^q_p(z) \geq \min_{i\in I_y} \max_{j\in I^c_y} \rho^q_p(z)_{i,j},$$

would require us to solve naively $|I_y||I^c_y|$ such problems. Seemingly, the bound in Theorem 2.4 is much cheaper as it requires only $(|I_y| + |I^c_y|)$ operations even though one has to note that the bound only exists for the case when $p = q$.

i) A lower bound: Theorem 2.10 shows that when fixing $i^* = \operatorname*{arg\,min}_{i\in I_y} \|z - w_i\|$ and then computing

$$\min\nolimits_{j\in I^y_c} \rho^q_p(z)_{i^*,j},$$

yields by Lemma 2.9 a lower bound on $\epsilon^q_p(z)$. By Theorem 2.10 this lower bound is for the case $p = q$ still better than the one of Theorem 2.4 while having the same complexity of $|I_y| + |I^c_y|$ operations. Obviously, when integrating box constraints, that is $\mathcal{X} = [0, 1]^d$, the gap can only become larger between the two bounds.

ii) Using simpler lower bounds: When certifying bounds for $\mathcal{X} = [0, 1]^d$ we first compute the lower bounds for $\mathcal{X} = \mathbb{R}^d$ as they are often available in closed form and are definitely lower bounds for the more restricted case $\mathcal{X} = [0, 1]^d$. By fixing again i^* we can then use $s_j := \rho^q_p(z)_{i^*,j}$ and define the minimum and minimizer as $(\lambda, j^*) = \min_{j\in I^c_y} \rho^q_p(z)_{i^*,j}$. Now, let us denote by $\kappa^q_p(z)_{i^*,j}$ the corresponding quantity when using $\mathcal{X} = [0, 1]^d$ instead of $\mathcal{X} = \mathbb{R}^d$. Then we only need to compute $\kappa^q_p(z)_{i^*,j}$ if $s_j < \kappa^q_p(z)_{i^*,j^*}$, which is typically satisfied for very few instances, so most computations are pruned.

iii) Dual problems: as in (Wang et al., 2019) we use the dual problems when computing $r^q_2(z)_j$. This has three advantages. First, we always get a lower bound using weak duality, second, we stop solving $r^q_p(z)_j$ when the dual value is higher than our currently smallest upper bound and third; empirically only few constraints of the problems become active; thus, the solutions are dual-sparse.

Final Certification: in Algorithm 1 we sketch the certification process. It does not include all details (see above) which we use for speeding up the computation of lower bounds as well as the exact minimal adversarial perturbation.

Algorithm 1 Sketch of certification algorithm for correctly classified point z

```
**// Computation of $\lambda$ as lower bound on $\epsilon^q_p(z)$**
$i^* = \operatorname*{arg\,min}_{i\in I_y} \|z - w_i\|_p$
$s_j = \rho^q_p(z)_{i^*,j}, j \in I^c_y$ ($s_j$ lower bounds $r^q_p(z)_j$)
$(\lambda, j^*) = \min_{j\in I^c_y} \rho^q_p(z)_{i^*,j}$
**if** minimizer $x^*$ of $\rho^q_p(z)_{i^*,j^*}$ is feasible for $r^q_p(z)_{j^*}$ **then**
  $\epsilon^q_p(z) = \lambda$ and return
**else**
  $\lambda$ is lower bound on $\epsilon^p_q(z)$
**end if**
**// Computation of $\epsilon^q_p(z)$ ( $p = 2$ or $(p, q) = (1, \infty)$)**
$\mu = r^q_p(z)_{j^*}$ // (it holds $\mu \geq \epsilon^q_p(z)$)
**for** $j = 1$ **to** $|I^c_y|$ **do**
  **if** $s_j < \mu$ **then**
    compute $r^q_p(z)_j$
    **if** $r^q_p(z)_j < \mu$ **then**
      $\mu = r^q_p(z)_j$
    **end if**
  **end if**
  $\epsilon^q_p(z) = \mu$
**end for**
```

3. Perceptual Metric

The hypothesis underlying the goal of adversarial robustness is that images which have the same semantic content, should be classified the same (with the exception at the true decision boundary). However, this would require a human oracle which judges if the semantic content is similar. A proxy is the typical ℓ_p-threat model, where for suitable chosen radius ϵ_p one expects that for a given image x also $B_p(x, \epsilon_p)$ should be classified the same as for humans the resulting images are (semantically) indistinguishable from the original image. However, it is well known that pixel-based ℓ_p-distances are not a good measure of image similarity. A huge literature in computer vision discusses the construction of metrics which better correspond to human perception of similarity of images e.g. the SSIM metric of (Wang et al., 2004). More recently, neural perceptual metrics, such as the LPIPS distance, have been proposed in (Zhang et al., 2018). The LPIPS distance is based on a feature mapping of a fixed neural network and has been shown to correlate better with human perception (Zhang et al., 2018; Laidlaw et al., 2021). In (Laidlaw et al., 2021) it has been used as threat model in adversarial training. Moreover, (Kireev et al., 2021) have shown that the LPIPS distance better correlates with the severity level of common corruptions than the ℓ_2-distance. Moreover, ℓ_p-distance based NPC are not competitive for CIFAR10. These two aspects motivate us to investigate the perceptual metric-based PNPC as well as novel techniques for the certification in the LPIPS-threat model.

The perceptual metric: Given the output $g^{(l)}(x) \in \mathbb{R}^{H_l \times W_l \times C_l}$ of the l-th layer of a fixed neural network (we use Alexnet as suggested by (Zhang et al., 2018)) of height H_l and width W_l and channels C_l, we define the normalized output of a layer as $\hat{g}^{(l)}_{h,w}(x) = \frac{g^{(l)}_{h,w}(x)}{\left\|g^{(l)}_{h,w}(x)\right\|_2}$. The LPIPS distance d is then defined in (Zhang et al., 2018) as

$$d^2(x, y) = \sum_{l\in I_L} \frac{1}{H_l W_l} \sum_{h,w} \left\| w_l \odot \left(\hat{g}^{(l)}_{h,w}(x) - \hat{g}^{(l)}_{h,w}(y)\right)\right\|^2_2,$$

where the weights w_l are learned using human perception data and I_L is the index set of layers used for the metric. We follow (Laidlaw et al., 2021) and use the unweighted (i.e., weights perform an identity mapping) version in order to be able to directly compare to them. However, it would be easy to adapt our approach for the weighted version. We define the embedding, $\phi : [0,1]^d \to \mathbb{R}^D$

$$x \mapsto \phi(x) = \left(\frac{\hat{g}^{(l)}}{\sqrt{H_l W_l}}\right)_{l \in I_L}, \qquad (3)$$

so that the unweighted LPIPS distance can simply be written as a standard Euclidean distance $d(x,y) = \|\phi(x) - \phi(y)\|_2$ in the embedding space.

The mapped image space $\phi(I)$ of all natural images I is a subset of $\phi([0,1]^d)$, which can be seen as an at most d-dimensional continuous "submanifold" of the embedding space $\mathbb{R}^D$. Thus for all points $z \in \mathbb{R}^D \backslash \phi([0,1]^d)$ there exists no pre-image in $[0,1]^d$. However, the Euclidean distance between every mapped images $x, y \in I$ corresponds to the perceptual distance between them. Thus we train our PNPC in the embedding space $\mathbb{R}^D$ and certify it with respect to the Euclidean distance which in turn yields guarantees with respect to the LPIPS distance.

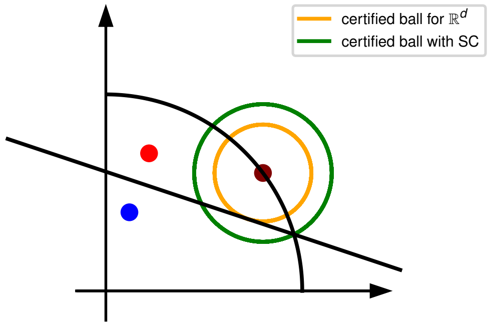


Figure 3: The embedded data $\phi(x)$ lies on the intersection of the positive orthant and the sphere (shown in black). In the embedding space the ℓ_2-metric corresponds to the perceptual metric. Taking these non-negative spherical constraints (SC) into account we can certify a much larger ball than using only the standard certification in $\mathbb{R}^d$.

3.1. Certification in the Perceptual threat model

Up to our knowledge this is the first paper showing results for certification with respect to this threat model aligned with human vision. We can use all techniques we have discussed in Section 2 as we are working with a Euclidean distance in $\mathbb{R}^D$. However, we have more knowledge about $\phi([0,1]^d)$ as the output of each layer is normalized so that $\phi(x)$ lies on a product of spheres with radius $r_l = \frac{1}{\sqrt{H_l W_l}}$ as

$$\left\|\phi^{(l)}_{h,w}(x)\right\|_2 = \left\|\frac{\hat{g}^{(l)}_{h,w}}{\sqrt{H_l W_l}}\right\|_2 = \frac{1}{\sqrt{H_l W_l}} := r_l, \qquad (4)$$

for any $l \in I_L, h \in I_H, w \in I_W$. Additionally, we know due to the structure of Alexnet that $\phi_l(x)$ is non-negative for all layers, see Figure 3 for an illustration. While we can integrate some of the properties of the mapping ϕ into the certification, it is computationally intractable to use as constraint $x \in \phi([0,1]^d)$. Thus our certification works on an overapproximation of $\phi([0,1]^d)$ and thus yields lower bounds on the minimal adversarial perceptual distance.

Basically, we can write our constraints in $\mathbb{R}^D$ as

$$\begin{aligned} \mathcal{X} &= \mathcal{X}_1 \times \cdots \times \mathcal{X}_L \qquad (5) \\ \mathcal{X}_l &= \left(\frac{1}{\sqrt{H_l W_l}} S^{c_l} \cap [0,\infty)^{c_l}\right)^{H_l W_l}, \quad l = 1, \ldots, L, \end{aligned}$$

where c_l is the number of channels in layer l of the output of the layer l and $D = \sum_{l=1}^{L} H_l W_l c_l$. We use upper indices (e.g., $x^{(h,w,l)}$) to denote slice of vector x which corresponds to vector of channels at position h, w in layer l. The constants r_l for $1 \leq l \leq L$ were defined in (4).

As we use ℓ_2-NPC we have to compute:

$$\begin{aligned} \rho(z)_{i,j} = \min_{x \in \mathbb{R}^D} \quad & \|x - z\|_2 \qquad (6) \\ \text{sbj. to:} \quad & \langle x, w_j - w_i \rangle + \frac{\|w_i\|_2^2 - \|w_j\|_2^2}{2} \geq 0 \\ & \left\|x^{(h,w,l)}\right\|_2^2 = r_l^2, \; l = 1, \ldots, L \\ & \qquad\qquad h = 1, \ldots, H_l \\ & \qquad\qquad w = 1, \ldots, W_l \\ & x_d \geq 0, \qquad d = 1, \ldots, D. \end{aligned}$$

Despite this problem is non-convex due to the quadratic equality constraints we can derive a convex dual problem (we derive it for an equivalent problem) which is sufficient to provide us with lower bounds using weak duality.

Proposition 3.1. *Define $v = w_j - w_i$ and $b = \frac{\|w_i\|_2^2 - \|w_j\|_2^2}{2}$. A lower bound on the optimal value of the optimization problem (6) is given by*

$$\sqrt{2L + 2\left(\max_{\lambda \geq 0} - \sum_{h,w,l} \left\|\left(z^{(h,w,l)} - \lambda v^{(h,w,l)}\right)^+\right\|_2 r_l + \lambda b\right)}$$

which can be efficiently computed using bisection. The summation $\sum_{h,w,l}$ is a shortcut for $\sum_{1 \leq l \leq L} \sum_{1 \leq h \leq H_l} \sum_{1 \leq w \leq W_l}$.

In the experimental results in Figure 4 one can clearly see that using this lower bound improves significantly over the standard lower bound of Lemma 2.9.

4. Efficient Training of PNPC

In this section we describe the training procedure for our PNPC. The key advantage compared to the work of (Saralajew et al., 2020) is that despite our lower bounds, see Theorem 2.10, are better and often tight, they can be computed with the same time complexity as theirs if $p \in \{2, \infty\}$. Thus we can do efficient certified training. As objective we use the capped sum of the lower bounds:

$$\max_{(w_i)_{i \in I}} \frac{1}{n} \sum_{r=1}^{n} \min \Big\{ \min_{j \in I_y^c} \max_{i \in I_y} \rho_p^q(z_r)_{i,j}, R \Big\},$$

where we recall the definition of ρ_p^q from 2:

$$\begin{aligned} \rho_p^q(z)_{i,j} = \min_{x \in \mathbb{R}^d} \quad & \left\|x - z\right\|_q \\ \text{sbj. to:} \quad & \left\|x - w_i\right\|_p - \left\|x - w_j\right\|_p \geq 0 \\ & x \in \mathcal{X} \end{aligned} \tag{7}$$

and R is an upper bound on the margin we want to enforce. The cap is introduced in order to avoid that single training points have excessive margin at the price of many others having small margin; in turn, it is equivalent to minimizing hinge-loss. The loss is minimized via stochastic gradient descent resp. ADAM with large batch sizes. Note further that, for misclassified points we use a signed version of $\rho_p^q(z_r)_{i,j}$ by flipping the constraint in (2) and using $-\rho_p^q(z_r)_{j,i}$ instead, which can be interpreted as signed distance to the decision boundary. Doing this has the advantage that we get gradient information from all points. Maximizing our objective has a direct interpretation in terms of maximizing robust accuracy or more precisely the area under the robustness curve capped at radius R. This is in contrast to (Saralajew et al., 2020) who use as loss their lower bound divided by the sum of the distances where this interpretation is due to the rescaling not applicable.

5. Experiments

The code for experiments is available in our repository[1] where we also provide the training details. We first evaluate the improvements in the certification of better lower bounds resp. exact computation compared to the ones of (Saralajew et al., 2020) as well as (Wang et al., 2019). In a second set of experiments we compare our ℓ_p-PNPC to the ℓ_p-NPC of (Saralajew et al., 2020) resp. to nearest neighbor classification as well as deterministic and probabilistic certification techniques for neural networks on MNIST and CIFAR10 (see App. H). Finally, we discuss our NPC using the perceptual metric and its certification where there is no competitor as up to our knowledge this is the first paper providing robustness certificates. The training time is about a few hours on a laptop.

Table 3: **Lower bounds on $\epsilon_p^q(z)$.** Mean of the lower bounds of (Saralajew et al., 2020) (Theorem 2.4), the lower bounds of (Wang et al., 2019)) in (13) ($\mathcal{X} = \mathbb{R}^d$), our lower bounds integrating $\mathcal{X} = [0, 1]^d$ and the exact radius on the test set for ℓ_2-NPC for ℓ_1-,ℓ_2- and ℓ_∞-threat model.

Model	Num. Proto.	Threat model	Lower bounds Th. 2.2 $\mathbb{R}^d$	Th. 2.6 $\mathbb{R}^d$	Th. 2.6 $[0,1]^d$	Exact radius $[0,1]^d$
ℓ_2-PNPC	4000	ℓ_1	-	9.71	11.77	12.11
		ℓ_2	0.39	1.86	1.96	1.99
MNIST		ℓ_∞	-	0.14	0.16	0.17

Comparison of our lower bounds: One of the major contributions of this paper are our efficient lower bounds on the minimal adversarial perturbation $\epsilon_p^q(z)$. They can be computed so fast that it is feasible to use them during training. We show in Table 3 that our ℓ_q-bounds improve significantly over the ones of (Saralajew et al., 2020) (Th. 2.4, $\mathcal{X} = \mathbb{R}^d$), which only work if $p = q$ and (Wang et al., 2019) (Lemma 2.9, $\mathcal{X} = \mathbb{R}^d$, see (13) for $p = 2$) as we are the only ones who integrate box constraints (Lemma 2.9, $\mathcal{X} = [0, 1]^d$). In Table 3, we show that for the ℓ_1-, ℓ_2- and ℓ_∞ threat models our lower bounds are very close to the exact values. The computation of these lower bounds takes for the **full** test set of MNIST: ℓ_1: 188s, ℓ_2: 33s, ℓ_∞: 131s. This is two orders of magnitude faster than the computation of the exact bounds in Table 4. For our ℓ_∞-NPC and ℓ_∞-threat model we get mean lower bounds of 0.3545 for (Saralajew et al., 2020), 0.3560 for the ones from (15) with $\mathcal{X} = \mathbb{R}^d$, and 0.3616 for ours from Lemma 2.9 with $\mathcal{X} = [0, 1]^d$ in (20). Here the differences are smaller than for the ℓ_2-NPC.

Time for certification: The computation of the exact minimal adversarial perturbation is only feasible for relatively small neural networks (Tjeng & Tedrake, 2017) and for ensemble of decision trees (Kantchelian et al., 2016). Both use mixed-integer formulations which do not scale well. For boosted decision stumps one can compute the exact robust accuracy (Andriushchenko & Hein, 2019). However, the computation of the exact robust accuracy is already considerably easier than the minimal adversarial perturbation. For ℓ_2-NPC we can compute the exact adversarial perturbation for the ℓ_1-, ℓ_2-, and ℓ_∞-threat model. In Table 4 we report the certification time per point and other statistics for our ℓ_2-PNPC prototypes on MNIST with 400 prototypes per class (ppc) and the ℓ_2-GLVQ -model of (Saralajew et al., 2020) on CIFAR10 with 128 ppc. We can also produce weaker

[1] https://github.com/vvoracek/Provably-Adversarially-Robust-Nearest-Prototype-Classifiers.

Table 4: **Time/Statistics for exact minimal adversarial perturbation** for ℓ_2-NPC

Model	Num. Proto.	Threat model	Direct solved	Total QP/LP	QP/LP per pt	Cert. Time per pt
ℓ_2-PNPC MNIST	4000	ℓ_1	4261 (43.8%)	10195	1.86	0.54s
		ℓ_2	3170 (32.6%)	11630	1.77	0.49s
		ℓ_∞	2073 (21.3%)	21081	2.75	1.3s
ℓ_2-GLVQ CIFAR10	1280	ℓ_1	3683 (75.8%)	1817	1.54	0.76s
		ℓ_2	3546 (73.0%)	1777	1.35	0.25s
		ℓ_∞	3511 (72.2%)	1933	1.43	0.9s

Table 5: **MNIST:** lower (CRA) and upper bounds (URA) on ℓ_2-robust accuracy for ℓ_2-NPC

MNIST	std.	$\epsilon_2 = 1.5$		$\epsilon_2 = 1.58$		$\epsilon_2 = 2$	
	acc.	CRA	URA	CRA	URA	CRA	URA
ℓ_2-PNPC	97.3	**75.5**	75.5	**73.0**	73.0	**56.1**	56.1
ℓ_2-GLVQ	95.8	69.7	69.7	67.1	67.1	53.5	53.5
1-NN	96.9	52.1	52.1	47.3	47.3	23.7	23.7
GloRob	97.0	-	-	62.8	81.9	-	-
OrthConv	**98.1**	-	-	61.0	75.5	-	-
LocLip	96.3	-	-	55.8	78.2	-	-
BCP	92.4	-	-	47.9	64.7	-	-
CAP	88.1	-	-	44.5	67.9	-	-
SmoothLip$_{\sigma=0.5}$	**98.7**	**81.8***	-	-	-	0*	-
SmoothLip$_{\sigma=1}$	93.7	62.7*	-	-	-	44.9*	-

Table 6: **MNIST:** lower (CRA) and upper bounds (URA) on robust accuracy for multiple threat models for our ℓ_2-PNPC, the ℓ_2-NPC of (Saralajew et al., 2020), a 1-NN classifier. As comparison we show MMR-Univ of (Croce & Hein, 2020a) which is a neural network specifically trained for certifiable multiple-norm robustness.

MNIST	std.	$\epsilon_1 = 1$		$\epsilon_2 = 0.3$		$\epsilon_\infty = 0.1$		union	
	acc.	CRA	URA	CRA	URA	CRA	URA	CRA	URA
ℓ_2-PNPC	**97.3**	**96.2**	96.2	**95.6**	95.6	85.8	85.8	**85.8**	85.8
ℓ_2-GLVQ	95.8	94.2	94.2	93.2	93.2	80.9	80.9	80.9	80.9
1-NN	96.9	95.0	-	93.6	93.6	78.3	-	78.3	-
MMR-U	97.0	79.2	93.6	89.6	93.8	**87.6**	87.6	79.2	87.6

certificates faster. For instance, using Lemma 2.9, we can certify MNIST robust accuracy 67% in under $2s$ instead of the exact 73% reported in Table 5.

Regarding the model of ℓ_2-GLVQ on CIFAR10, we have an accuracy of 48.6% (which corresponds to 4859 correctly classified test points). Of these ones we can solve between 72.2% for ℓ_∞ and 75.8% for ℓ_1 directly using Lemma 2.9 by checking the condition after the computation of the lower bounds. This shows the usefulness of Lemma 2.9 as it avoids a lot of QPs (ℓ_2) rsp. LPs (ℓ_1, ℓ_∞) to be solved. Next we see that the number of LPs/QPs needed to be solved per point is less than 1.43 which has to be compared to the worst case of $|I_y^c| = 1152$. This shows that our prior reduction using our tight lower bounds integrating box constraints helps to significantly reduce the number of problems $r_p^q(z)_j$ which need to be solved. In total we get certification times between $0.25s$ (ℓ_2) and $0.9s$ (ℓ_∞) per point which allows us to do the exact certification for all three threat models.

Evaluation of our NPC: We report certified robust accuracy (CRA) and upper bounds on robust accuracy (URA), e.g. computed via an adversarial attack, on MNIST and CIFAR10 (in App. H) for PNPC and the GLVQ of (Saralajew et al., 2020). For ℓ_2-NPC CRA and URA are equal as we compute exact adversarial perturbations. As an interesting baseline, we report results for the one nearest neighbor classifier (1NN). Additionally, we compare to deterministic and probabilistic certification techniques of neural networks.

MNIST - ℓ_2-NPC: In Table 5 we show the results for the **ℓ_2-threat model** on MNIST. Our ℓ_2-PNPC outperforms the ℓ_2-GLVQ for all ϵ_2. The values for ϵ_2 were chosen according to the neural network literature. Note that our ℓ_2-PNPC outperforms all deterministic methods: GlobRob (Leino et al., 2021), OrthConv (Singla et al., 2022), LoClip (Huang et al., 2021), BCP (Lee et al., 2020) and CAP (Wong et al., 2018) in terms of certified robust accuracy and often in the terms of clean accuracy. For the details on comparison with orthogonal convolutions, see Appendix I. The randomized smoothing approach SmoothLip of (Jeong et al., 2021) outperforms us for $\sigma = 0.5$ in terms of clean accuracy and robust accuracy at $\epsilon_2 = 1.5$ but their robust accuracy at $\epsilon_2 = 2$ is zero, whereas we have 56.1% exact robust accuracy. Their second model with $\sigma = 1$ which is able to certify also larger radii is in all aspects worse than our ℓ_2-PNPC. This shows that our certified prototype classifiers can challenge neural networks in terms of certified robust accuracy. Moreover, (Saralajew et al., 2020) report for their ℓ_2-GLVQ a certified robust accuracy of 34.4% at $\epsilon = 1.58$ whereas with our exact computation we get that their exact robust accuracy is 67.1%. This shows the quality of our exact certification techniques. With our certified training PNPC has 6% better robust accuracy and 1.5% better standard accuracy (97.3% vs. 95.8%) than ℓ_2-GLVQ.

The advantage of our ℓ_2-NPC is that we can certify any ℓ_q-threat model, especially ℓ_1 and ℓ_∞. This allows us to compute the **exact robust accuracy in the union of the ℓ_1-, ℓ_2- and ℓ_∞-balls.** The only other approach which has provided certified lower bounds (CRA) on multiple-norm robustness is MMR-U from (Croce & Hein, 2020a) who certify a neural network. In Table 6 we compare our multiple-norm robust accuracy for the ϵ_q which were chosen in (Croce & Hein, 2020a). Our ℓ_2-PNPC outperforms MMR-U significantly in terms of certified ℓ_1-and ℓ_2-robustness as well as in the union.

Table 7: **MNIST:** lower (CRA) and upper bounds (URA) on ℓ_∞-robust accuracy for ℓ_∞-NPC obtained using Lemma 2.9.

MNIST	std.	$\epsilon_\infty = 0.1$		$\epsilon_\infty = 0.3$		$\epsilon_\infty = 0.4$	
	acc.	CRA	URA	CRA	URA	CRA	URA
ℓ_∞-PNPC	94.69	91.19	91.19	78.68	78.86	65.58	65.96
ℓ_∞-GLVQ	96.34	93.52	93.52	80.76	81.04	61.29	62.94
ℓ_∞-neuron	**98.6**	-	-	**93.1**	95.3	-	-
CROWN-IBP	98.2	-	-	93.0	94.0	**87.4**	90.4
ReLU-S	97.3	-	-	80.7	92.1	-	-
CAP	87.4	-	-	56.9	-	-	-

MNIST - ℓ_∞-NPC We compare our ℓ_∞-PNPC to the ℓ_∞-GLVQ of (Saralajew et al., 2020). For reference we provide the best results for the ℓ_∞-certfied neural networks: ℓ_∞-neurons (Zhang et al., 2021), CROWN-IBP (Zhang et al., 2020), as well as slightly older results; ReLU-stability (Xiao et al., 2019) and CAP (Wong et al., 2018) to put our results into context. We perform slightly worse than (Saralajew et al., 2020) for small radii, but significantly better for the bigger one. Due to our better lower bounds but also by using AutoAttack (Croce & Hein, 2020b) for computing the upper bounds we close the gap between upper and lower bounds from 4.2% in (Saralajew et al., 2020) to 0.3%. To attack the classifier with AutoAttack, we interpret the negative distance to the closest prototype from a particular class as the logit value.

Perceptual metric NPC As discussed in Section 3 it is unlikely that ℓ_p-NPC will work for image classifcation tasks like CIFAR10. However, with the perceptual metric LPIPS (based on Alexnet) which corresponds to an ℓ_2-metric in the embedding space, we get much better results with our Perceptual-PNPC (P-PNPC). In Figure 4 we show the certified robust accuracy (lower bound of Lemma 2.9) as a function of the LPIPS-radius for the standard ℓ_2-lower bounds and for the improved lower bounds taking into account the constraints of the embedding. We have three important observations. We achieve a clean accuracy of 80.3% which is quite remarkable for a classifier with certified robust accuracy. Second, this is up to our knowledge the first result on certified robustness with respect to the LPIPS-threat model. Third, (Laidlaw et al., 2021) who do empirical perceptual adversarial training with a a ResNet 50 get only 71.6% clean accuracy and only a URA of 9.8% which is more than 30% **worse** than our CRA of 40.5%. Moreover, our URA computed using the LPA-attack of (Laidlaw et al., 2021) is with 70.3% remarkably high. These are very promising results justifying more research in PNPC for perceptual metrics.

On the other hand, in (Laidlaw et al., 2021) it is noted that models trained to be robust w.r.t. LPIPS-threat model are empirically robust also to other threat models such as ℓ_2 or ℓ_∞ - even though one has to state that their model has only a robust accuracy of 9.8%. This generalization does not hold for P-PNPC. For ℓ_∞ threat model, we observed (empirical) robust accuracies 49%, 23%, 2%, 0% for radii $1/255, 2/255, 4/255, 8/255$. For ℓ_2 we have robust accuracy $51\%, 29\%, 5\%, 0\%$ for radii $0.14, 0.25, 0.5, 1$. While the robust accuracies are non-trivial, they are not comparable to the ones achieved in (Laidlaw et al., 2021). As our P-PNPC is much more robust with respect to the LPIPS-threat model than the neural network of (Laidlaw et al., 2021), it is thus an open question if this threat model leads indeed to a generalization to other threat models.

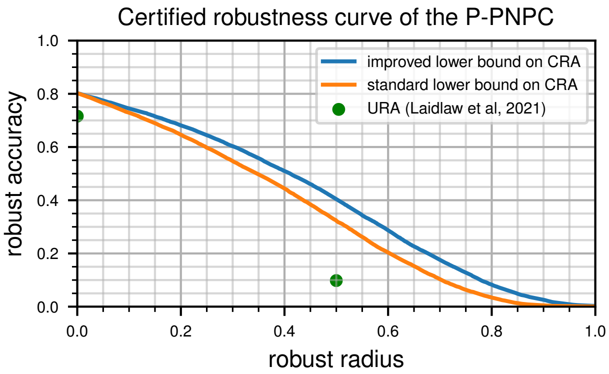


Figure 4: The certified robust accuracy as a function of the radius of the LPIPS-threat model. Integrating the spherical plus non-negativity constraints leads to huge improvements. The standard accuracy as well as the **empirical** robust accuracy of (Laidlaw et al., 2021) are *worse* than **certified** robust accuracy of P-PNPC by a large margin.

6. Conclusion

We have provided theoretical foundations as well as efficient algorithmic tools for the computation of the exact minimal adversarial perturbation, as well as lower bounds, for nearest prototype classifiers for several threat models, including the perceptual metric LPIPS. We have shown SOTA performance for deterministic ℓ_2-certification on MNIST and remarkably strong certified robustness results with respect to the LPIPS metric. Thus we think that NPC deserve more attention in our research community.

Acknowledgements

The authors acknowledge support from the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039A) and the DFG Cluster of Excellence “Machine Learning – New Perspectives for Science”, EXC 2064/1, project number 390727645.

References

Andriushchenko, M. and Hein, M. Provably robust boosted decision stumps and trees against adversarial attacks. In *NeurIPS*, 2019.

Athalye, A., Carlini, N., and Wagner, D. A. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

Bertsimas, D., Dunn, J., Pawlowski, C., and Zhuo, Y. D. Robust classification. *INFORMS Journal on Optimization*, 1:2–34, 2018.

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

Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., Madry, A., and Kurakin, A. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019.

Chen, H., Zhang, H., Boning, D., and Hsieh, C.-J. Robust decision trees against adversarial examples. In *ICML*, 2019.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing. In *NeurIPS*, 2019.

Crammer, K., Gilad-bachrach, R., Navot, A., and Tishby, N. Margin analysis of the lvq algorithm. In *NeurIPS*, 2003.

Croce, F. and Hein, M. Provable robustness against all adversarial l_p-perturbations for $p \geq 1$. In *ICLR*, 2020a.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks, 2020b.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Gowal, S., Dvijotham, K., Stanforth, R., Bunel, R., Qin, C., Uesato, J., Arandjelovic, R., Mann, T. A., and Kohli, P. On the effectiveness of interval bound propagation for training verifiably robust models. preprint, arXiv:1810.12715v3, 2018.

Hein, M. and Andriushchenko, M. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *NeurIPS*, 2017.

Huang, Y., Zhang, H., Shi, Y., Kolter, J. Z., and Anandkumar, A. Training certifiably robust neural networks with efficient local lipschitz bounds. In *NeurIPS*, 2021.

Jeong, J., Park, S., Kim, M., Lee, H.-C., Kim, D., and Shin, J. Smoothmix: Training confidence-calibrated smoothed classifiers for certified robustness. In *NeurIPS*, 2021.

Kantchelian, A., Tygar, J., and Joseph, A. Evasion and hardening of tree ensemble classifiers. In *ICML*, 2016.

Kireev, K., Andriushchenko, M., and Flammarion, N. On the effectiveness of adversarial training against common corruptions. *arXiv preprint, arXiv:2103.02325*, 2021.

Kohonen, T. *Learning Vector Quantization*, pp. 175–189. Springer Berlin Heidelberg, 1995.

Laidlaw, C., Singla, S., and Feizi, S. Perceptual adversarial robustness: Defense against unseen threat models. In *ICLR*, 2021.

Lee, S., Lee, J., and Park, S. Lipschitz-certifiable training with a tight outer bound. In *NeurIPS*, 2020.

Leino, K., Wang, Z., and Fredrikson, M. Globally-robust neural networks. In *ICML*, 2021.

Li, L., Qi, X., Xie, T., and Li, B. Sok: Certified robustness for deep neural networks. *arXiv preprint arXiv:2009.04131*, 2020.

Li, Q., Haque, S., Anil, C., Lucas, J., Grosse, R. B., and Jacobsen, J.-H. Preventing gradient attenuation in lipschitz constrained convolutional networks. In *NeurIPS*, 2019.

Mirman, M., Gehr, T., and Vechev, M. Differentiable abstract interpretation for provably robust neural networks. In *ICML*, 2018.

Papernot, N., McDaniel, P., and Goodfellow, I. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016.

Russu, P., Demontis, A., Biggio, B., Fumera, G., and Roli, F. Secure kernel machines against evasion attacks. In *ACM workshop on AI and security*. ACM, 2016.

Saralajew, S., Holdijk, L., and Villmann, T. Fast adversarial robustness certification of nearest prototype classifiers for arbitrary seminorms. In *NeurIPS*, 2020.

Singla, S., Singla, S., and Feizi, S. Improved deterministic l2 robustness on CIFAR-10 and CIFAR-100. In *ICLR*, 2022.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, pp. 2503–2511, 2014.

Tjeng, V. and Tedrake, R. Verifying neural networks with mixed integer programming. preprint, arXiv:1711.07356v1, 2017.

Tramer, F., Carlini, N., Brendel, W., and Madry, A. On adaptive attacks to adversarial example defenses. In *NeurIPS*, 2020.

Trockman, A. and Kolter, J. Z. Orthogonalizing convolutional layers with the cayley transform. In *ICLR*, 2021.

Wang, L., Liu, X., Yi, J., Zhou, Z.-H., and Hsieh, C.-J. Evaluating the robustness of nearest neighbor classifiers: A primal-dual perspective. *arXiv preprint, arXiv:1906.03972*, 2019.

Wang, Y., Jha, S., and Chaudhuri, K. Analyzing the robustness of nearest neighbors to adversarial examples. In *ICML*, 2018.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Wong, E. and Kolter, J. Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *ICML*, 2018.

Wong, E., Schmidt, F., Metzen, J. H., and Kolter, J. Z. Scaling provable adversarial defenses. In *NeurIPS*, 2018.

Xiao, K. Y., Tjeng, V., Shafiullah, N. M., and Madry, A. Training for faster adversarial robustness verification via inducing relu stability. In *ICLR*, 2019.

Xu, H., Caramanis, C., and Mannor, S. Robustness and regularization of support vector machines. *Journal of Machine Learning Research*, 10:1485–1510, 2009.

Zhang, B., Cai, T., Lu, Z., He, D., and Wang, L. Towards certifying l-infinity robustness using neural networks with l-inf-dist neurons. In *ICML*, 2021.

Zhang, H., Chen, H., Xiao, C., Gowal, S., Stanforth, R., Li, B., Boning, D., and Hsieh, C.-J. Towards stable and efficient training of verifiably robust neural networks. In *ICLR*, 2020.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

The appendix includes the missing proofs from the paper (App. A to App. G), results for ℓ_2-NPC for CIFAR10 in App. H and comparison to orthogonal convolutions in I.

A. Proof of Theorem 2.4

Proof. We note that for any x it holds by the triangle inequality

$$d(x, w_i) \leq d(z, x) + d(w_i, z).$$

Thus it holds

$$d(x, w_i) - d(x, w_j) \leq d(z, w_i) + d(x, z) - d(z, w_j) + d(x, z),$$

and we get that all points in $B_d(z, r)$ are classified the same as z if

$$\max_{x \in B_d(z,r)} \left(\min_{i \in I_y} d(x, w_i) - \min_{j \in I_y^c} d(x, w_j) \right) \leq \min_{i \in I_y} d(z, w_i) - \min_{j \in I_y^c} d(z, w_j) + 2r \leq 0$$

This yields that

$$r \leq \frac{\min_{j \in I_y^c} d(z, w_j) - \min_{i \in I_c} d(z, w_i)}{2}.$$

□

B. Proof of Theorem 2.6

Proof. We define the set

$$U_j^{(p)} = \{x \in \mathbb{R}^n \mid \|x - w_i\|_p - \|x - w_j\|_p \geq 0 \quad \forall\, i \in I_y\}.$$

as the set of points which are not classified as y when only the single prototype with index $j \in I_y^c$ would be considered. We get the full set of points not classified as y as the union $\bigcup_{j \in I_y^c} U_j^{(p)}$. We define $r_p^q(z)_j = \min_{x \in U_j^{(p)}} \|z - x\|_q$ as the radius of the largest ℓ_q-ball which still fits into $\mathbb{R}^d \backslash U_j^{(p)}$ and thus is fully classified as class y when only considering $j \in I_y^c$. Thus the radius $\epsilon_p^q(z)$ of the largest ℓ_q-ball fitting into $\mathbb{R}^d \backslash \bigcup_{j \in I_y^c} U_j^{(p)} = \bigcap_{j \in I_y^c} \left(\mathbb{R}^d \backslash U_j^{(p)} \right)$ is given by

$$\epsilon_p^q(z) = \min_{j \in I_y^c} r_p^q(z)_j,$$

which can be seen using the fact that $r_p^q(z)_j$ is the minimal ℓ_q-distance to $U_j^{(p)}$. □

C. Proof of Lemma 2.9

Proof. As for each $i \in I_y$ the problem for $\rho_p^q(z)_{i,j}$ is a relaxation of the problem for $r_p^q(z)_j$ (as we are omitting constraints), it holds for each $i \in I_y$:

$$r_p^q(z)_j \geq \rho_p^q(z)_{i,j} \implies r_p^q(z)_j \geq \max_{i \in I_y} \rho_p^q(z)_{i,j}.$$

Thus

$$\epsilon_q^p(z) = \min\nolimits_{j \in I_y^c} r_p^q(z)_j \;\geq\; \min_{j \in I_c^y} \max_{i \in I_y} \rho_p^q(z)_{i,j}.$$

For the second part if x^* satisfies

$$\|x^* - w_i\|_p - \|x^* - w_{j^*}\|_p \geq 0 \quad \forall i \in I_y,$$

then it is a feasible point for the optimization problem of $r_p^q(z)_{j^*}$ in (1) and thus $\rho_p^q(z)_{i^*,j^*} = r_p^q(z)_{j^*}$. By definition and by the just derived result it holds

$$\rho_p^q(z)_{i^*,j^*} = r_p^q(z)_{j^*} \geq \epsilon_p^q(z) \geq \rho_p^q(z)_{i^*,j^*},$$

and thus equality has to hold. □

D. Proof of Theorem 2.10

Lemma D.1. $\rho_p^p(z)_{i,j} \geq \frac{\|w_j - z\|_p - \|w_i - z\|_p}{2}$

Proof. First we restate the definition of ρ:

$$\begin{aligned} \rho_p^q(z)_{i,j} = \min_{x \in \mathbb{R}^d} \quad & \|x - z\|_q \\ \text{sbj. to:} \quad & \|x - w_i\|_p - \|x - w_j\|_p \geq 0 \\ & x \in \mathcal{X} \end{aligned} \tag{8}$$

We consider $p = q$. By the triangle inequality the following holds for any $x \in \mathcal{X}$, thus also for any adversarial perturbation x for which $\|x - w_i\|_p - \|x - w_j\|_p \geq 0$:

$$\begin{aligned} &\|x - w_i\|_p \leq \|x - z\|_p + \|z - w_i\|_p \\ &\|z - w_j\|_p \leq \|x - z\|_p + \|x - w_j\|_p \quad \implies \quad \|x - w_j\|_p \geq \|z - w_j\|_p - \|x - z\|_p \end{aligned} \tag{9}$$

Summing the inequalities up we get for any feasible $x \in \mathcal{X}$ satisfying the inequality constraint,

$$\|z - w_i\|_p - \|z - w_j\| + 2\|x - z\|_p \geq \|x - w_i\|_p - \|x - w_j\|_p \geq 0. \tag{10}$$

which yields finally

$$\|x - z\|_p \geq \frac{\|w_j - z\|_p - \|w_i - z\|_p}{2}. \tag{11}$$

Therefore, $\rho_p^p(z)_{i,j} \geq \frac{\|w_j - z\|_p - \|w_i - z\|_p}{2}$. □

Proof of Theorem 2.10. If z is misclassified, then it reduces to $0 \geq 0$ which holds. Otherwise, by Lemma D.1, it holds $\rho_p^p(z)_{i,j} \geq \frac{\|z - w_j\|_p - \|z - w_i\|_p}{2}$. Then

$$\begin{aligned} \min_{j \in I_c^y} \max_{i \in I_y} \rho_p^p(z)_{i,j} &\geq \min_{j \in I_c^y} \rho_p^p(z)_{i^*,j} \\ &\geq \min_{j \in I_c^y} \frac{\|z - w_j\|_p - \|z - w_{i^*}\|_p}{2} \\ &= \frac{\min_{j \in I_y^c} \|z - w_j\|_p - \min_{i \in I_y} \|z - w_i\|_p}{2} \end{aligned}$$

We further show that there are cases where the inequality is strict. Consider a d-dimensional example where $z = (0, \ldots, 0)$, $\{w_j \mid j \in I_c^y\} = \{(2, 0, \ldots, 0)\}$, $\{w_i \mid i \in I^y\} = \{(1, 0, \ldots, 0)\}$. It clearly holds that $\min_{j \in I_c^y} \max_{i \in I_y} \rho_p^p(z)_{i,j} = 1.5$, while $\max\left\{0, \frac{\min_{j \in I_y^c} \|z - w_j\|_p - \min_{i \in I_y} \|z - w_i\|_p}{2}\right\} = 1$ for any p.

□

E. Proof of Theorem 2.7

Theorem 2.7 *The computational complexities of optimization problems $\rho_p^q(z)_{i,j}$ for $p, q \in \{1, 2, \infty\}$ for $\mathcal{X} = \mathbb{R}^d$ are summarised in Table 8.*

Throughout the proof, we assume z is correctly classified, otherwise the solution is 0. We prove the theorem gradually for cases $p = 2$ and any q, then $q = \infty$ and any p, then $p = 1$ and any $q \neq \infty$ and finally $p = \infty$, $q = 1, 2$. For most of the cases, we discuss the possibility of incorporating box constraints, which usually increases complexity from $O(d)$ to $O(d \log(d))$. We also remark that using the median of medians algorithm, one could avoid sorting coordinates, and could achieve $\Theta(d)$ complexities. We, for the sake of simplicity, will be sorting point for the price of $log(d)$ factor in complexity.

		ℓ_q-threat model		
		ℓ_1	ℓ_2	ℓ_∞
ℓ_p-distance	ℓ_1	NP-hard	NP-hard	$O(d\log(d))$
	ℓ_2	$\Theta(d)$	$\Theta(d)$	$\Theta(d)$
	ℓ_∞	$\Theta(d)$	$O(d\log(d))$	$\Theta(d)$

Table 8: Computational complexity of $\rho_p^q(z)_{i,j}$.

Proof for case $p = 2$ ***and any*** q.

$$\begin{aligned} \rho_2^q(z)_{i,j} = \min_{x\in\mathbb{R}^d} \quad & \|x-z\|_q \\ \text{sbj. to:} \quad & \|x-w_i\|_2 - \|x-w_j\|_2 \geq 0 \\ & x \in \mathcal{X} \end{aligned} \tag{12}$$

We equivalently rewrite the constraint in the following way:

$$\begin{aligned} \|x-w_i\|_2 - \|x-w_j\|_2 &\geq 0, \\ \|x-z+z-w_i\|_2^2 - \|x-z+z-w_j\|_2^2 &\geq 0, \\ \|x-z\|_2^2 + 2\langle x-z, z-w_i\rangle + \|z-w_i\|_2^2 - \left(\|x-z\|_2^2 + 2\langle x-z, z-w_j\rangle + \|z-w_j\|_2^2\right) &\geq 0, \\ 2\langle x-z, w_j-w_i\rangle &\geq \|z-w_j\|_2^2 - \|z-w_i\|_2^2, \\ 2\|x-z\|_q \|w_j-w_i\|_{\frac{q}{q-1}} \geq 2\langle x-z, w_j-w_i\rangle &\geq \|z-w_j\|_2^2 - \|z-w_i\|_2^2, \\ \|x-z\|_q &\geq \frac{\|z-w_j\|_2^2 - \|z-w_i\|_2^2}{2\|w_j-w_i\|_{\frac{q}{q-1}}}. \end{aligned}$$

Since Hölder's inequality is tight, we conclude

$$\rho_2^q(z)_{i,j} = \frac{\|z-w_j\|_2^2 - \|z-w_i\|_2^2}{2\|w_j-w_i\|_{\frac{q}{q-1}}}. \tag{13}$$

We note that analogical derivation holds for minimising $\|x-z\|$ in any norm, not just for the q-norm. In that case, $\|\cdot\|_{\frac{q}{q-1}}$ is replaced with the dual norm of the considered norm. The box-constrained version of this problem can be solved in $O(d\log d)$, see e.g., Section 4 of (Hein & Andriushchenko, 2017). □

Proof for case $p = q = \infty$.

$$\begin{aligned} \rho_p^\infty(z)_{i,j} = \min_{x\in\mathbb{R}^d} \quad & \|x-z\|_\infty \\ \text{sbj. to:} \quad & \|x-w_i\|_\infty - \|x-w_j\|_\infty \geq 0 \\ & x \in \mathcal{X} \end{aligned} \tag{14}$$

We note that whenever $\|x-w_i\|_\infty - \|x-w_j\|_\infty \geq 0$, then also $\|x'-w_i\|_\infty - \|x'-w_j\|_\infty \geq 0$, where $x'^{(l)} = x^{(l)} + \alpha\,\mathrm{sign}(w_j^{(l)} - w_i^{(l)})$ for any positive α and some $l = 1 \dots d$, and $x'^{(l)} = x^{(l)}$ for the coordinates. That is, we can move $x^{(l)}$ in the direction from $w_i^{(l)}$ to $w_j^{(l)}$, since if $|x'^{(l)} - w_j^{(l)}| > |x^{(l)} - w_j(l)|$, then also $|x'^{(l)} - w_i^{l}| > |x'^{(l)} - w_j(l)|$. On the other hand, if $|x^{(l)} - w_i^{(l)}| > |x'^{(l)} - w_i^{(l)}|$, then also $|x^{(l)} - w_i^{(l)}| < |x^{(l)} - w_j^{(l)}|$, thus l was not the maximising index of $\|x-w_i\|_\infty$, and consequently $\|x-w_i\|_\infty = \|x'-w_i\|_\infty$. The remaining case is trivial; thus, $\|x'-w_i\|_p - \|x'-w_j\|_p \geq 0$. This argument may be repeated d times to conclude that when $\rho_\infty^\infty(z)_{i,j} = \epsilon$, then a minimizer of Problem 14 is $x^* = z + \epsilon\,\mathrm{sign}\,(w_j - w_i)$.

Therefore, the problem is to find the smallest ϵ for which $\|z + \epsilon \operatorname{sign}(w_j - w_i) - w_i\|_\infty \geq \|z + \epsilon \operatorname{sign}(w_j - w_i) - w_j\|_\infty$. Note that

$$\|z+\epsilon \operatorname{sign}(w_j - w_i) - w_j\|_\infty = \max_{l=1,\dots,d} \max \left\{ z^{(l)} + \epsilon \operatorname{sign}\left(w_j^{(l)} - w_i^{(l)}\right) - w_j^{(l)}, -\left(z^{(l)} + \epsilon \operatorname{sign}\left(w_j^{(l)} - w_i^{(l)}\right) - w_j^{(l)}\right)\right\};$$

thus, it is a maximum of $2d$ linear functions, each of which has slope either 1, or -1. Let $\alpha_i = \arg\min \operatorname{sign}(w_j - w_i)(z - w_i)$ and $\beta_i = \arg\max \operatorname{sign}(w_j - w_i)(z - w_i)$, analogously for α_j, β_j. Then

$$\begin{aligned}
&\|z + \epsilon \operatorname{sign}(w_j - w_i) - w_i\|_\infty - \|z + \epsilon \operatorname{sign}(w_j - w_i) - w_j\|_\infty = \\
&\max\left\{-\epsilon - \operatorname{sign}\left(w_j^{(\alpha_i)} - w_i^{(\alpha_i)}\right)\left(z^{(\alpha_i)} - w_i^{(\alpha_i)}\right), \epsilon + \operatorname{sign}\left(w_j^{(\beta_i)} - w_i^{(\beta_i)}\right)\left(z^{(\beta_i)} - w_i^{(\beta_i)}\right)\right\} - \\
&\max\left\{-\epsilon - \operatorname{sign}\left(w_j^{(\alpha_j)} - w_i^{(\alpha_j)}\right)\left(z^{(\alpha_j)} - w_j^{(\alpha_j)}\right), \epsilon + \operatorname{sign}\left(w_j^{(\beta_j)} - w_i^{(\beta_j)}\right)\left(z^{(\beta_j)} - w_j^{(\beta_j)}\right)\right\}.
\end{aligned}$$

Moreover, we can analyse to slope of $\|z + \epsilon \operatorname{sign}(w_j - w_i) - w_i\|_\infty - \|z + \epsilon \operatorname{sign}(w_j - w_i) - w_j\|_\infty$ and see that it is non-zero only in the interval between points

$$\epsilon_i = \frac{-\operatorname{sign}\left(w_j^{(\alpha_i)} - w_i^{(\alpha_i)}\right)\left(z^{(\alpha_i)} - w_i^{(\alpha_i)}\right) - \operatorname{sign}\left(w_j^{(\beta_i)} - w_i^{(\beta_i)}\right)\left(z^{(\beta_i)} - w_i^{(\beta_i)}\right))}{2},$$

and

$$\epsilon_j = \frac{-\operatorname{sign}\left(w_j^{(\alpha_j)} - w_i^{(\alpha_j)}\right)\left(z^{(\alpha_j)} - w_j^{(\alpha_j)}\right) - \operatorname{sign}\left(w_j^{(\beta_j)} - w_i^{(\beta_j)}\right)\left(z^{(\beta_j)} - w_j^{(\beta_j)}\right))}{2},$$

where the slope is 2. Now it is easy to compute the value of $\|z + \epsilon \operatorname{sign}(w_j - w_i) - w_i\|_\infty - \|z + \epsilon \operatorname{sign}(w_j - w_i) - w_j\|_\infty$ for very big (V_+) and very small (V_-) values of ϵ, where the active linear function is the one with negative slope. Concretely

$$\begin{aligned}
V_- &= \operatorname{sign}\left(w_j^{(\alpha_j)} - w_i^{(\alpha_j)}\right)\left(z^{(\alpha_j)} - w_j^{(\alpha_j)}\right) - \operatorname{sign}\left(w_j^{(\alpha_i)} - w_i^{(\alpha_i)}\right)\left(z^{(\alpha_i)} - w_i^{(\alpha_i)}\right), \\
V_+ &= \operatorname{sign}\left(w_j^{(\beta_i)} - w_i^{(\beta_i)}\right)\left(z^{(\beta_i)} - w_i^{(\beta_i)}\right) - \operatorname{sign}\left(w_j^{(\beta_j)} - w_i^{(\beta_j)}\right)\left(z^{(\beta_j)} - w_j^{(\beta_j)}\right).
\end{aligned}$$

Now we use the fact that the slope is 2 between ϵ_i and ϵ_j to find the point where the norms are equal; Thus, we can express $\rho_\infty^\infty(z)_{i,j}$ as

$$\rho_\infty^\infty(z)_{i,j} = \max\{\epsilon_i, \epsilon_j\} - \frac{V_+}{2},$$

or as

$$\rho_\infty^\infty(z)_{i,j} = \min\{\epsilon_i, \epsilon_j\} - \frac{V_-}{2}.$$

We can take the mean of both expression, then we arrive at

$$\rho_\infty^\infty(z)_{i,j} = \epsilon_i + \epsilon_j - \frac{V_- + V_+}{2},$$

which simplifies to

$$\rho_\infty^\infty(z)_{i,j} = -\frac{\operatorname{sign}\left(w_j^{(\alpha_j)} - w_i^{(\alpha_j)}\right)\left(z^{(\alpha_j)} - w_j^{(\alpha_j)}\right) + \operatorname{sign}\left(w_j^{(\beta_i)} - w_i^{(\beta_i)}\right)\left(z^{(\beta_i)} - w_i^{(\beta_i)}\right)}{2},$$

and by substituting back the definitions of α_j, β_i:

$$\rho^{\infty}_{\infty}(z)_{i,j} = \frac{\max\limits_{l=1,\ldots,d} -\operatorname{sign}\left(w_j^{(l)} - w_i^{(l)}\right)\left(z^{(l)} - w_j^{(l)}\right) - \max\limits_{l=1,\ldots,d} \operatorname{sign}\left(w_j^{(l)} - w_i^{(l)}\right)\left(z^{(l)} - w_i^{(l)}\right)}{2}. \tag{15}$$

□

***Proof for case** $q = \infty$**,** $p \neq \infty$ **.** The value of $\rho^{\infty}_{p}(z)_{i,j}$ is the minimal non-negative ϵ for which the following maximization problem has non-negative value.

$$\begin{aligned} \max_{x \in \mathbb{R}^d} \quad & \|x - w_i\|_p^p - \|x - w_j\|_p^p \\ \text{sbj. to:} \quad & \|x - z\|_\infty \leq \epsilon \\ & x \in \mathcal{X} \end{aligned} \tag{16}$$

It can be decomposed into d independent problems indexed by l.

$$\begin{aligned} \max_{x^{(l)} \in \mathbb{R}} \quad & |x^{(l)} - w_i^{(l)}|^p - |x^{(l)} - w_j^{(l)}|^p \\ \text{sbj. to:} \quad & |x^{(l)} - z^{(l)}| \leq \epsilon \\ & x^{(l)} \in \mathcal{X}^{(l)} \end{aligned} \tag{17}$$

Derivative of the objective function w.r.t. $x^{(l)}$ is $p|x^{(l)} - w_i^{(l)}|^{p-1} \operatorname{sign}(x^{(l)} - w_i^{(l)}) - p|x^{(l)} - w_j^{(l)}|^{p-1} \operatorname{sign}(x^{(l)} - w_j^{(l)})$, which is non-zero whenever $w_i^{(l)} \neq w_j^{(l)}$. Thus, the maximum is attained at a point where a constraint is active, and the value of the problem is $|z^{(l)} + \epsilon \operatorname{sign}(w_j^{(l)} - w_i^{(l)}) - w_i^{(l)}|^p - |z^{(l)} + \epsilon \operatorname{sign}(w_j^{(l)} - w_i^{(l)}) - w_j^{(l)}|^p$. When $p = 2$, the value of the objective is a quadratic function in ϵ; thus, the value of the original objective is also a quadratic function in ϵ and we can easily obtain a solution to the original problem. For the sake of completeness, we show that this approach results in the same $\rho^{\infty}_{2}(z)_{i,j}$ as we derived before:

$$\begin{aligned} \sum_{l=1}^{d} \left(\left(z^{(l)} + \epsilon \operatorname{sign}(w_j^{(l)} - w_i^{(l)}) - w_i^{(l)}\right)^2 - \left(z^{(l)} + \epsilon \operatorname{sign}(w_j^{(l)} - w_i^{(l)}) - w_j^{(l)}\right)^2 \right) &\geq 0, \\ \sum_{l=1}^{d} \left((z^{(l)} - w_i^{(l)})^2 - (z^{(l)} - w_j^{(l)})^2 + 2\epsilon \operatorname{sign}(w_j^{(l)} - w_i^{(l)})(w_j^{(l)} - w_i^{(l)}) \right) &\geq 0, \\ \|z - w_i\|_2^2 - \|z - w_j\|_2^2 + 2\epsilon \|w_j - w_i\|_1 &\geq 0, \\ \epsilon \geq \frac{\|z - w_j\|_2^2 - \|z - w_i\|_2^2}{2\|w_j - w_i\|_1}&. \end{aligned} \tag{18}$$

If $p = 1$, the value of the objective is piecewise linear and non-decreasing; thus, the original objective is again, piecewise linear and non-decreasing. Then we can order the breaking points and find the smallest admissible ϵ for the original problem using binary search. Note that the objective is maximised not just in the aforementioned case, but also when

$$x^{(l)} = \begin{cases} w_j^{(l)}, & \text{if } |z^{(l)} - w_j^{(l)}| \leq \epsilon. \\ z_j^{(l)} + \epsilon \operatorname{sign}\left(w_j^l - z^l\right), & \text{otherwise.} \end{cases} \tag{19}$$

For other values of p, it may be difficult to solve the problem exactly. However, as we have already shown, it is easy ($\Theta(d)$) to determine if $\rho^{\infty}_{p}(z)_{i,j} > \epsilon$ given an ϵ, thus the problem can be solved approximately using binary search for any p with logarithmic complexity in accuracy.

To conclude the cases $\rho^{p}_{\infty}(z)_{i,j}$, we discuss the addition of box constraints. As we have shown, a minimizer of the problems is always $x^* = z + \epsilon \operatorname{sign}(w_j - w_i)$, and identical arguments would suggest that with box constraints, it would hold that $x^* = \max(0, \min(1, z + \epsilon \operatorname{sign}(w_j - w_i)))$. Therefore, given a radius, certification is done in $O(d)$ even with box constraints. Otherwise, we would need to either order coordinates according to value of ϵ when a box constraint for $x^* = \max(0, \min(1, z + \epsilon \operatorname{sign}(w_j - w_i)))$ becomes active, and then perform a binary search over the constrained problems. This adds a $log(d)$ factor to the complexity. Note that for the case $p = 1$ we are already performing a binary search, so we do them at once. Or we can do a binary search over ϵ to find a minimal one which causes

$$x = \max(0, \min(1, z + \epsilon \operatorname{sign}(w_j - w_i))) \tag{20}$$

to be misclassified.

□

Proof for case $p = 1$***,*** $q \neq \infty$. We ave already discussed the case of $\rho_1^\infty(z)_{i,j}$, so it is omitted here. For all other values of q, we show its NP-hardness by reducing the knapsack problem to the decision version of problem if given $\epsilon > 0$, $\rho_1^\infty(z)_{i,j} \leq \epsilon$.

Theorem E.1 (Knapsack)**.** *The following problem is NP-complete.*
Given vectors $w, p \in \mathbb{N}^n$ *and constants* W, P*. Decide if there is a vector* $x \in \{0,1\}^n$ *such that* $\langle p, x\rangle \geq P$ *and* $\langle w, x\rangle \leq W$*.*

For the sake of clarity, we use u, v instead of w_i, w_j to get rid of unnecessary subscript. Let w, p, W, P describe an instance of the knapsack problem. Let a pair of prototypes $u_t, v_t \in \mathbb{R}^{n+2}$ be defined in the following way for some real t and $l = 1, \dots, n$

$$\begin{aligned} u_t^{(l)} &= \sqrt[q]{w^{(l)}}, \\ v_t^{(l)} &= \sqrt[q]{w^{(l)}} - \frac{p^{(l)}}{t}, \end{aligned} \tag{21}$$

let also

$$\begin{aligned} u_t^{(n+1)} &= \sqrt[q]{W} + \frac{\max\left(0, \left(2P - \sum_{i=1}^n p^{(i)}\right)\right)}{t}, \\ v_t^{(n+1)} &= \sqrt[q]{W}, \\ u_t^{(n+2)} &= \sqrt[q]{W}, \\ v_t^{(n+2)} &= \sqrt[q]{W} + \frac{\max\left(0, \left(\sum_{i=1}^n p^{(i)} - 2P\right)\right)}{t}, \end{aligned} \tag{22}$$

and $\epsilon = \sqrt[q]{W}$. Now we show that whenever there is an allocation $x \in \{0,1\}^n$ such that $\langle p, x\rangle \geq P$ and $\langle w, x\rangle \leq W$, then $\rho_1^q(0) \leq \epsilon$ for any sufficiently large t such that the first n components of $v^{(t)}$ are positive. It holds that:

$$\|v_t\|_1 \leq \sum_{i=1}^n \sqrt[q]{w^{(i)}} - \sum_{i=1}^n \frac{p^{(i)}}{t} + 2\sqrt[q]{W} + \frac{\max\left(0, \left(\sum_{i=1}^n p^{(i)} - 2P\right)\right)}{t} \leq \sum_{i=1}^n \sqrt[q]{w^{(i)}} + 2\sqrt[q]{W} \leq \|u_t\|_1 . \tag{23}$$

Consider the following point

$$\delta^{(k)} = \begin{cases} \sqrt[q]{w^{(k)}}, & \text{if } x^{(k)} = 1. \\ 0, & \text{otherwise.} \end{cases} \tag{24}$$

It has q-norm of at most ϵ:

$$\|\delta\|_q = \left(\sum_{i=1}^{n+2} \delta^{(i)q}\right)^{\frac{1}{q}} = \left(\sum_{i=1}^n x^{(i)} \cdot w^{(i)}\right)^{\frac{1}{q}} \leq \sqrt[q]{W} = \epsilon. \tag{25}$$

Also it holds that

$$\begin{aligned}
\|v_t-\delta\|_1 &= \sum_{i=1}^{n}\left(x^{(i)}\cdot\frac{p^{(i)}}{t}+(1-x^{(i)})\sqrt[q]{w^{(i)}}\right)+2\sqrt[q]{W}+\frac{\max\left(0,\left(\sum_{i=1}^{n}p^{(i)}-2P\right)\right)}{t},\\
&\geq \sum_{i=1}^{n}(1-x^{(i)})\sqrt[q]{w^{(i)}}+2\sqrt[q]{W}+\frac{P+\max\left(0,\left(\sum_{i=1}^{n}p^{(i)}-2P\right)\right)}{t},\\
&\geq \sum_{i=1}^{n}(1-x^{(i)})\sqrt[q]{w^{(i)}}+2\sqrt[q]{W}+\frac{\sum_{i=1}^{n}p^{(i)}-P+\max\left(0,\left(2P-\sum_{i=1}^{n}p^{(i)}\right)\right)}{t}\geq\|u_t-\delta\|_1 .
\end{aligned} \tag{26}$$

Therefore, $\rho_1^q(0)\leq\epsilon$.

Now we move on to the second direction; we show that whenever the constructed problem is feasible, then also the knapsack problem is feasible.

Let there be a δ such that $\|\delta\|_q\leq\epsilon$ and $\|v_t-\delta\|\geq\|u_t-\delta\|$. Then we can WLoG assume $\delta^{(n+1)}=0$, and $\sqrt[q]{w^{(l)}}-p^{(l)}/t\leq\delta^{(l)}\leq\sqrt[q]{w^{(l)}}$ for $l=1,\dots,n$. Now consider the following allocation for $k=1,\dots,n$.

$$x^{(k)}=\begin{cases}0, & \text{if } \delta^{(k)}=0.\\ 1, & \text{otherwise.}\end{cases} \tag{27}$$

We show that if t is sufficiently large, then x is a valid allocation. First, let us look at the $\langle w,x\rangle\leq W$ constraint;

$$\sum_{i=1}^{n}\delta^{(i)q}=\sum_{i=1}^{n}w^{i}\cdot x^{i}-o(1)=\langle w,x\rangle-o(1)\leq W;$$

thus, $\langle w,x\rangle\leq W$ for sufficiently large t.

For the other constraint, first note for $l=1,\dots,n$:

$$\left(|v_t-\delta|^{(l)}-|u_t-\delta|^{(i)}\right)=\begin{cases}p^{(i)}/t, & \text{if } x^{(i)}=0.\\ \geq -p^{(i)}/t, & \text{otherwise.}\end{cases} \tag{28}$$

Then

$$\sum_{i=1}^{n}\left(|v_t-\delta|^{(i)}-|u_t-\delta|^{(i)}\right)\geq\frac{\sum_{i=1}^{n}p^{(i)}-2\langle x,p\rangle}{t}, \tag{29}$$

and finally

$$\begin{gathered}
\frac{\sum_{i=1}^{n}p^{(i)}-2P}{t}\geq\frac{\sum_{i=1}^{n}p^{(i)}-2\langle x,p\rangle}{t},\\
\langle x,p\rangle\geq P.
\end{gathered} \tag{30}$$

□

Proof for case $p=\infty$***,*** $q=1$.

$$\begin{aligned}
\rho_\infty^1(z)_{i,j}=\min_{x\in\mathbb{R}^d}\quad & \|x-z\|_1\\
\text{sbj. to:}\quad & \|x-w_i\|_\infty-\|x-w_j\|_\infty\geq 0\\
& x\in\mathcal{X}
\end{aligned} \tag{31}$$

Let $\delta_x = x - z$ where $x \in \arg\min \rho^1_\infty(z)_{i,j}$, We note that there exists x such that δ_x contains only a single non-zero element. To see why, let there be some δ_x with multiple non-zero elements from which we construct $\delta_{x'}$ with more zeros such that $x' \in \arg\min \rho^1_\infty(z)_{i,j}$. Let $l^* = \arg\max_l |x^{(l)} - w_i^{(l)}|$. Take any index $k \neq l^*$ such that $\delta_x^{(k)} \neq 0$. Then consider a perturbation $\delta_x{}'$

$$\delta_{x'}^{(l)} = \begin{cases} \delta_x^{(l)} + |\delta_x^{(k)}| \operatorname{sign}(x^{(l)} - w_i^{(l)}), & \text{if } l = l^*. \\ 0, & \text{if } l = k. \\ \delta_x^{(l)}, & \text{otherwise.} \end{cases} \tag{32}$$

Now, $\|x' - w_i\|_\infty = \|x - w_i\|_\infty + |\delta_x^k| \geq \|x - w_j\|_\infty + |\delta^{(l)}| \geq \|\delta' - w_j\|_\infty$ which concludes the argument. Now it is sufficient to solve the problem for every coordinate separately and take the maximal value; thus, the original problem is solved in linear time.

□

Proof for case $p = \infty$***,*** $q = 2$.

$$\begin{aligned} \rho^2_\infty(z)_{i,j} = \min_{x \in \mathbb{R}^d} \quad & \|x - z\|_2 \\ \text{sbj. to:} \quad & \|x - w_i\|_\infty - \|x - w_j\|_\infty \geq 0 \\ & x \in \mathcal{X} \end{aligned} \tag{33}$$

Let x be the minimizer. Then we split the proof into two cases. Either there is an index l such that $\|x - w_i\|_\infty = |x^{(l)} - w_i^{(l)}| = |x^{(l)} - w_j^{(l)} = \|x - w_j\|_\infty$. In that case, $|z^{(l)} - w_j^{(l)}| > |z^{(l)} - w_i^{(l)}|$ and $|w_i^{(l)} - w_j^{(l)}|$ is maximal. Then we can compute x in one pass and verify that indeed $\|x - w_i\|_\infty = \|x - w_j\|_\infty$.

Otherwise, let us Assume that we know $\|x - w_i\|_\infty = \|x - w_j\|_\infty = d$ for the optimal x. That is, for every coordinate l we have to ensure that $|x^{(l)} - w_j^{(l)}| \leq d$, and also that there is a coordinate k where $|x^{(l)} - w_i^{(l)}| = d$; thus, we can construct x minimizing $\|x - z\|_2$ as

$$x^{(l)} = \begin{cases} w_j^{(l)} + d \operatorname{sign}(z^{(l)} - w_i^{(l)}), & \text{if } |w_j^{(l)} - x^{(l)}| > d. \\ w_i^{(l)} + d \operatorname{sign}(w_j^{(l)} - w_i^{(l)}), & \text{if } l = k. \\ z^{(l)}, & \text{otherwise,} \end{cases} \tag{34}$$

where $k = \min \arg\max_l \quad \operatorname{sign}\left(w_j^{(l)} - w_i^{(l)}\right)\left(z^{(l)} - w_i^{(l)}\right)$.

Now we sort (so further we assume the array is sorted) the coordinates according to values of $|w_j^{(l)} - x^{(l)}|$.

Then minimum of $\|x - z\|_2^2$ is attained for some d which lies in some interval $[|w_j^{(m)} - x^{(m)}|, |w_j^{(m+1)} - x^{(m+1)}|]$. Inside every such interval, $\|x - z\|_2^2$ is a quadratic expression in d. For the m-th interval, the equation is

$$\|x - z\|_2^2 = \sum_{l=1}^{m} \left(z^{(l)} - w_j^{(l)} + d \operatorname{sign}(z^{(l)} - w_i^{(l)})\right)^2 + \left(z^{(l)} - \operatorname{sign}\left(w_j^{(k)} - w_i^{(k)}\right)\left(z^{(k)} - w_i^{(k)}\right)\right)^2.$$

So we can minimize a quadratic function $\|x - z\|_2^2$ over an interval $[|w_j^{(m)} - x^{(m)}|, |w_j^{(m+1)} - x^{(m+1)}|]$. We can also see that for the $m+1$-th equation, we only add one term to the m-th equation; thus, we can solve every interval in $O(1)$ and take the minimal ϵ. Consequently, the time complexity is dominated by $O(d\log(d))$ needed for sorting which concludes the proof.

□

		ℓ_q-threat model		
		ℓ_1	ℓ_2	ℓ_∞
ℓ_p-distance	ℓ_1	NP-hard	NP-hard	Poly
	ℓ_2	Poly	Poly	Poly
	ℓ_∞	NP-hard	NP-hard	NP-hard

Table 9: Computational Complexity of $r^q(z)$.

F. Proof of Theorem 2.8

Theorem 2.8 *The computational complexities of optimization problems $r_p^q(z)_{i,j}$ in (1) for $p, q \in \{1, 2, \infty\}$ and $\mathcal{X} = [0, 1]^d$ are summarized in Table 9.*

Proof. The Problem $r_1^q(z)_j$ for $q \neq \infty$ cannot be easier than the problem $\rho_1^q(z)_{i,j}$, thus since the latter is NP-hard, the first also has to be NP-hard. For the case $r_1^\infty(z)_j$, we recall that the optimal argument of $\rho_1^\infty(z)_{i,j}$ was in the form

$$x^{(l)} = \begin{cases} w_j^{(l)}, & \text{if } |z^{(l)} - w_j^{(l)}| \leq \epsilon, \\ z_j^{(l)} + \epsilon \operatorname{sign}\left(w_j^l - z^l\right), & \text{otherwise}, \end{cases} \tag{35}$$

where ϵ is the value of $\rho_1^\infty(z)_{i,j}$. Therefore, $r_1^\infty(z)_j = \max_i \rho_1^\infty(z)_{i,j}$ and the overall complexity is $O(d \log(d)|I_c^y|)$. When $p = 2$, then the problem reads as

$$\begin{aligned} r_2^q(z)_j = \min_{x \in \mathbb{R}^d} \quad & \|x - z\|_q \\ \text{sbj. to:} \quad & \|x - w_i\|_2 - \|x - w_j\|_2 \geq 0 \quad \forall\, i \in I_y \\ & x \in [0, 1]^d \end{aligned}$$

which is a convex optimization problem for any q and can be solved in polynomial time.

Finally, for the case $p = \infty$ we show that it is $NP - complete$ to solve the feasibility problem of $r_p^q(z)_j$, thus the problem is NP-hard for any q. To shorten the notation, we consider $I_y = 1, \ldots, n$ and whenever we say that some proposition holds for w_i, then we mean it holds for any w_i, $i \in 1, \ldots, n$.

We show this by reducing 3-SAT to it. Let there be a formula in CNF $\bigwedge_{i=1}^n \left(\alpha^{(i)} \vee \beta^{(i)} \vee \gamma^{(i)}\right)$, where all all the literals are from a set of v variables. For the sake of brevity, we make a correspondence between the literals and indices $1, \ldots, v$. Also when literal corresponding to i is negative, we will write it as $-i$. We will consider the following set of prototypes from $\mathbb{R}^{(v+1)}$.

$$w_j = (0, \ldots, 0, 3)$$

$$w_i^{(l)} = \begin{cases} -1, & \text{if } l \in \{\alpha^{(i)}, \beta^{(i)}, \gamma^{(i)}\}, \\ 2, & \text{if } -l \in \{\alpha^{(i)}, \beta^{(i)}, \gamma^{(i)}\}, \\ 0, & \text{otherwise}. \end{cases}$$

Clearly, for any $x \in [0, 1]^d$ it holds that $\|w_j - x\|_\infty \geq 2$, and also $\|x - w_i\|_\infty \leq 2$. Therefore, if $r_\infty^p(z)$ is feasible, then $\|x - w_i\|_\infty = 2$, which is equivalent to proposition $\left(x^{(|\alpha_i|)} = \frac{1+\operatorname{sign}\alpha_i}{2}\right) \vee \left(x^{(|\beta_i|)} = \frac{1+\operatorname{sign}\beta_i}{2}\right) \vee \left(x^{(|\gamma_i|)} = \frac{1+\operatorname{sign}\gamma_i}{2}\right)$. Such proposition have to be satisfied for every i, therefore it is equivalent to a formula in CNF

$$\bigwedge_{i=1}^n \left(\left(x^{(|\alpha_i|)} = \frac{1 + \operatorname{sign}\alpha_i}{2}\right) \vee \left(x^{(|\beta_i|)} = \frac{1 + \operatorname{sign}\beta_i}{2}\right) \vee \left(x^{(|\gamma_i|)} = \frac{1 + \operatorname{sign}\gamma_i}{2}\right) \right),$$

which is clearly equisatisfiable with the original CNF formula; thus, the feasibility problem is NP-complete. □

G. Proofs from the Perceptual Metric NPC

Here we slightly deviate from the main text, that we consider the squared objective which clearly is an equivalent problem

$$\begin{aligned}
\rho^2(z)_{i,j} = \min_{x\in\mathbb{R}^d} \quad & \|x-z\|_2^2 \\
\text{sbj. to:} \quad & \langle x, w_j - w_i\rangle + \frac{\|w_i\|_2^2 - \|w_j\|_2^2}{2} \geq 0 \\
& \left\|x^{(l)}\right\|_2^2 = r_l^2 \\
& x \geq 0,
\end{aligned}$$

where we use a shortcut $x^{(l)}$, instead of $x^{(h,w,l)}$, to simplify notation.

Proof of Proposition 3.1. Note that

$$\|x-z\|_2^2 = \sum_{l\in I_l} \left\|x^{(l)} - z^{(l)}\right\|_2^2,$$

and as $\left\|z^{(l)}\right\|_2 = r_l$ and we have $\left\|x^{(l)}\right\|_2 = r_l$ as constraint, we can equivalently minimize $-\sum_{l\in I_L}\left\langle x^{(l)}, z^{(l)}\right\rangle$ as objective. Let $v = w_j - w_i$ and $b = \frac{\|w_i\|_2^2 - \|w_j\|_2^2}{2}$. The Lagrangian of the non-convex problem (due to the quadratic *equality* constraints) is

$$\underset{\mu\geq 0}{L(x,\lambda,\alpha,\mu)} = -\sum_{l\in I_L}\left\langle x^{(l)}, z^{(l)}\right\rangle + \lambda\Big(\sum_{l\in I_l}\left\langle v^{(l)}, x^{(l)}\right\rangle + b\Big) + \sum_{l\in I_L}\frac{\alpha_l}{2}\Big(\left\|x^{(l)}\right\|_2^2 - r_l^2\Big) - \sum_{l\in I_L}\left\langle \mu^{(l)}, x^{(l)}\right\rangle$$

We get as critical point condition:

$$\nabla_{x^{(l)}} L = -z^{(l)} + \lambda v^{(l)} + \alpha_l x^{(l)} - \mu^{(l)} = 0,$$

which yields

$$x^{(l)} = \frac{1}{\alpha_l}\left(z^{(l)} - \lambda v^{(l)} + \mu^{(l)}\right).$$

The dual function $q(\lambda,\alpha,\mu)$ becomes

$$\begin{aligned}
q(\lambda,\alpha,\mu) = & -\sum_{l\in I_L}\frac{1}{\alpha_l}\left(\left\|z^{(l)}\right\|_2^2 - \lambda\left\langle v^{(l)}, z^{(l)}\right\rangle + \left\langle \mu^{(l)}, z^{(l)}\right\rangle\right) \\
& + \lambda\left(\sum_{l\in I_L}\frac{1}{\alpha_l}\left(\left\langle v^{(l)}, z^{(l)}\right\rangle - \lambda\left\|v^{(l)}\right\|_2^2 + \left\langle v^{(l)}, \mu^{(l)}\right\rangle\right) + b\right) \\
& + \sum_{l\in I_L}\frac{1}{2\alpha_l}\left(\left\|z^{(l)}\right\|_2^2 + \lambda^2\left\|v^{(l)}\right\|_2^2 + \left\|\mu^{(l)}\right\|_2^2 - 2\lambda\left\langle z^{(l)}, v^{(l)}\right\rangle + 2\left\langle z^{(l)}, \mu^{(l)}\right\rangle - 2\lambda\left\langle v^{(l)}, \mu^{(l)}\right\rangle\right) \\
& - \sum_{l\in I_L}\frac{\alpha_l r_l^2}{2} - \sum_{l\in I_L}\frac{1}{\alpha}\left(\left\langle \mu^{(l)}, z^{(l)}\right\rangle - \lambda\left\langle \mu^{(l)}, v^{(l)}\right\rangle + \left\|\mu^{(l)}\right\|_2^2\right),
\end{aligned}$$

which simplifies to

$$q(\lambda,\alpha,\mu) = -\sum_{l\in I_L}\frac{1}{2\alpha_l}\left\|z^{(l)} - \lambda v^{(l)} + \mu^{(l)}\right\|_2^2 + \lambda b - \sum_{l\in I_L}\frac{\alpha_l r_l^2}{2}.$$

We solve explicitly for α and get

$$\alpha_l = \frac{1}{r_l}\left\|z^{(l)} - \lambda v^{(l)} + \mu^{(l)}\right\|_2.$$

Then we get

$$q(\lambda,\mu) = -\sum_{l\in I_L}\left\|z^{(l)} - \lambda v^{(l)} + \mu^{(l)}\right\|_2 r_l + \lambda b.$$

Table 10: **CIFAR10:** lower (CRA) and upper bounds (URA) on ℓ_2-robust accuracy

CIFAR10	std.	$\epsilon_2 = 0.1$		$\epsilon_2 = 36/255$		$\epsilon_2 = 0.25$	
	acc.	CRA	URA	CRA	URA	CRA	URA
PNPC	49.2	43.9	43.9	41.9	41.9	36.4	36.4
GLVQ	48.6	43.3	43.3	41.5	41.5	37.9	37.9
1-NN	35.7	31.2	-	29.7	29.7	25.7	-
GloRob	77.0	-	-	58.4	69.2	-	-
LocLip	**77.4**	-	-	**60.7**	70.4	-	-
BCP	65.7	-	-	51.3	60.8	-	-
SmoothLip$_{\sigma=0.25}$	77.1	-	-	-	-	**67.9***	67.9*

Solving explicitly for μ, we get

$$q(\lambda) = -\sum_{l \in I_L} \left\| \left(z^{(l)} - \lambda v^{(l)} \right)^+ \right\|_2 r_l + \lambda b.$$

So this is a lower bound on $-\sum_{l \in I_L} \langle x^{*(l)}, z^{(l)} \rangle$, where x^* is the optimal primal variable by weak duality and thus going back to our actual objective we get using $\left\| x^{*(l)} \right\|_2 = \left\| z^{(l)} \right\| = r_l$ that

$$\|x^* - z\| = \sqrt{\|x^* - z\|_2^2} = \sqrt{2\sum_{l \in I_L} r_l^2 - 2\sum_{l \in I_L} \langle x^{*(l)}, z^{(l)} \rangle} \geq \sqrt{2\sum_{l \in I_L} r_l^2 + 2\left(\max_{\lambda \geq 0} - \sum_{l \in I_L} \left\| \left(z^{(l)} - \lambda v^{(l)} \right)^+ \right\|_2 r_l + \lambda b \right)},$$

where we have used weak duality. Now we go back to indexing using h, w, l instead of just l. Since $r_l = \frac{1}{\sqrt{H_l W_l}}$, it holds that

$$\sum_{h=1,\dots,H_l} \sum_{w=1,\dots,W_l} r_l^2 = 1;$$

thus, we can simplify the final expression as

$$\sqrt{2L + 2\left(\max_{\lambda \geq 0} - \sum_{h,w,l} \left\| \left(z^{(h,w,l)} - \lambda v^{(h,w,l)} \right)^+ \right\|_2 r_l + \lambda b \right)}.$$

Thus we have a one-dimensional convex optimization problem to solve in order to get a lower bound on the original objective, which is all we need for the certification. □

H. Results for CIFAR10 with ℓ_2-NPC

CIFAR10 - ℓ_2-NPC In Table 10 we compare certified robust accuray (CRA) and an upper bound on the robust accuray (URA) of several models on CIFAR10 for ℓ_2-threat model. Our ℓ_2-PNPC (800ppc) is slightly better than ℓ_2-GLVQ (128ppc) in terms of clean accuracy, and robust accuracy for $\epsilon_2 \in \{0.1, 36/255\}$, but ℓ_2-GLVQ is better for $\epsilon_2 = 0.25$. Note that the 1-NN is significantly worse showing that learning the prototypes helps improving the performance. Nevertheless, all NPC models are not competitive with neural networks which is to be expected as the ℓ_2-distance is not a good measure for image similarity. This is why we study PNPC with the perceptual metric which achieves to clean accuracies which are higher than the one of neural networks with provable robustness guarantees.

Table 11 shows the performance of ℓ_2-NPC for multiple threat models. ℓ_2-PNPC outperforms ℓ_2-GLVQ in terms of clean accuracy, ℓ_1- and ℓ_2-robust accuracy but is worse for ℓ_∞-robust accuracy and as this is the most difficult threat model it is also worse in the union. MMR-U outperforms the ℓ_2-NPC but the margin is relatively small.

Table 11: **CIFAR10:** lower (CRA) and upper bounds (URA) on robust accuracy for multiple threat models for our ℓ_2-PNPC, the ℓ_2-NPC of (Saralajew et al., 2020), a 1-NN classifier. As comparison we show MMR-Univ of (Croce & Hein, 2020a) which is a neural network specifically trained for certifiable multiple-norm robustness.

CIFAR10	std.	$\epsilon_1 = 2$		$\epsilon_2 = 0.1$		$\epsilon_\infty = 2/255$		union	
	acc.	CRA	URA	CRA	URA	CRA	URA	CRA	URA
ℓ_2-PNPC	49.2	**42.5**	42.5	41.9	41.9	32.7	32.7	32.7	32.7
ℓ_2-GLVQ	48.6	42.3	42.3	41.5	41.5	35.2	35.2	35.2	35.2
1-NN	35.7	30.0	-	29.7	29.7	22.5	-	22.5	-
MMR-U	**53.0**	36.6	43.6	**46.4**	48.1	**36.2**	36.2	**35.4**	36.2

Table 12: **MNIST:** Certified robust accuracy of networks with orthogonal convolutions. We computed robust accuracy after every epoch and the reported numbers are the maximal ones. The radius is 1.58

blocks \ γ	0	0.1	0.2	0.5	1
1	57.17	58.23	58.75	58.82	58.57
2	58.31	58.85	59.63	59.21	58.99
4	59.50	60.75	**61.02**	60.33	58.82
6	59.78	60.47	59.05	59.99	57.53

I. Comparison with orthogonal convolution networks

We evaluated the robustness orthogonal convolution networks on MNIST at radius 1.58. According to the evaluation in (Singla et al., 2022), the currently best method for orthogonal convolution networks is to combine skew orthogonal convolutions with Householder activations. According to the official repository, they suggest to choose to set the following parameters

- `--conv-layer` - We chose `soc` because it consistently outperformed baselines in the paper.
- `--activation` - We chose `hh1` activation, which is used in the experiments in the original paper.
- `--num-blocks` - We tried $1, 2, 4, 6$ blocks, possible values are $1 \dots 8$. In the original paper, it did not seem that more blocks boost performance.
- `--gamma` - We tried $0, 0.1, 0.2, 0.5, 1$. The original experiments used 0.1.
- `--lln` - The authors suggest to use last layer normalization when the number of classes is large, e.g., for CIFAR100, and do not use it for CIFAR10. We also did not use it.

We padded the MNIST images by 2 black pixels, so that we can directly use the original architecture which relied on the fact that the input images are 32×32. We also turned off the normalization by mean and variance as it is not commonly use for MNIST. We removed random horizontal flip from the set of possible augmentations, otherwise the setup is exactly as recommended. We note that the padding of MNIST image by 2 pixels is likely not the optimal way how to adapt the network to work with MNIST dataset.

The orthogonal convolutions from (Li et al., 2019) reports 56.4% certified robust accuracy. The method of (Trockman & Kolter, 2021) yielded 54% robust accuracy with the suggested setup.

I.1. Empirical robustness

We evaluated the empirical robustness of (Singla et al., 2022) using AutoAttack which is a stronger attack than what the competing methods used in Table 5. Thus, we don't conclude that orthogonal convolutions are (significantly) less empirically robust than the other evaluated methods.