BEYOND NEURAL SCALING LAWS FOR FAST PROVEN ROBUST CERTIFICATION OF NEAREST PROTOTYPE CLASSIFIERS

Nana Abeka Otoo
Authentic-Network
Chemnitz, Germany
nana.abekaotoo@authentic.network

Asirifi Boa
University of Applied Sciences
Mittweida, Germany
aboa@hs-mittweida.de

Muhammed Abubakar
Gesellschaft für wissenschaftliche
Datenverarbeitung mbH
Göttingen, Germany
muhammad.abubakar@gwdg.de

ABSTRACT

Methods beyond neural scaling laws for beating power scaling laws in machine learning have become topical for high-performance machine learning models. Nearest Prototype Classifiers (NPCs) introduce a category of machine learning models known for their interpretability. However, the performance of NPCs is frequently impacted by large datasets that scale to high dimensions. We surmount the performance hurdle by employing self-supervised prototype-based learning metrics to intelligently prune datasets of varying sizes, encompassing low and high dimensions. This process aims to enhance the robustification and certification of NPCs within the framework of the Learning Vector Quantization (LVQ) family of algorithms, utilizing Crammer normalization for arbitrary semi-norms (semi-metrics). The numerical evaluation of outcomes reveals that NPCs trained with pruned datasets demonstrate sustained or enhanced performance compared to instances where training is conducted with full datasets. The self-supervised prototype-based metric (SSL) and the Perceptual-SSL (P-SSL) utilized in this study remain unaffected by the intricacies of optimal hyperparameter selection. Consequently, data pruning metrics can be seamlessly integrated with triplet loss training to assess the empirical and guaranteed robustness of L^p-NPCs and Perceptual-NPCs (P-NPCs), facilitating the curation of datasets that contribute to research in applied machine learning.

Keywords Learning vector quantization · Prototype-based data pruning · Adversarial robustness · Guaranteed robustness certification

1 Introduction

Adversarial robustness has emerged as a focal point of concern for most applications in machine learning algorithms. [1, 2]. It is not a surprise that much research has been dedicated to discovering mathematically precise methods that can be used to harden learners, verify by complete methods their robustness [3, 4], and provide guarantees regarding their robustness by certification [4, 5]. Amongst the most interpretable machine learning models of interest are the Nearest Prototype Classifiers (NPCs) [6, 7]. Following the introduction of margin analysis by Crammer et al. [8], empirical evidence in favor of the adversarial robustness of Generalised Learning Vector Quantization (GLVQ) [9] was introduced [10] and later Saralajew et al.[11] presented a mathematical formulation that proved that GLVQ is robust against adversarial attacks using arbitrarily semi-norms. A similar conclusion derived by [11] regarding the robustness of NPCs is shown in [12] using the semi-metric. Hence, the interpretability and proven adversarial robustness guaranteed by the certification of NPCs are perfectly matched for their utilization and adoption in deep learning applications[11].

NPCs that optimize the hypothesis margin generalize well with guaranteed robustness certification[8, 10]. However, employing datasets with a substantial number of instances featuring predominantly higher dimensions doesn't necessarily guarantee enhanced performance of NPCs, as LVQs are specifically designed to be prototypically sparse models[13]. The application of a self-supervised prototype-based metric to beat power scaling laws has been investigated and introduced by [14]. Applied neural scaling laws that stipulate and confirm the fall of learner errors based on the power of the size of training data, size of the model, or a combination of both exist in practice for deep learning models [15, 16, 17].The case for error reduction relating to scaling the training dataset is not empirically true for NPCs, especially for high-dimensional datasets. However, generalization error reduction in NPCs regarding scaling the model size has been confirmed but fails for the same reason when dealing with robustness error against adversarial attacks[8, 10].

The practice of intelligent gathering of more limited quantities of thoughtfully chosen data, potentially resulting in the collection and distribution of foundational datasets[18] fit well for NPCs. Moreover, gathering large datasets for deep learning remains inefficient for most NPCs. Hence, the trade-off favoring adversarial robustness training and certification for NPCs will entail applying the best *fairly* sizeable dataset. This work seeks to look beyond neural scaling laws by investigating an optimal data pruning strategy inspired by prototype-based learning as a data preparation step [14] prior to the robustification and certification of NPCs with a focus on the LVQ family of advanced classifiers[9, 19].

2 Nearest Prototype Classifiers

Building Nearest Prototype Classifiers (NPCs) entail defining a dissimilarity measure along with a well-defined set of prototypes $\mathbf{W}$ chosen from the input space of $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, .., \mathbf{s}_n\} \subseteq \mathbb{R}^m$ where each prototype $\mathbf{w} \in \mathbf{W}$ is assigned a unique class $c(\mathbf{w}) \in \mathcal{C} = \{1, 2, \ldots, C\}$. Given an instance $\mathbf{s} \in \mathbf{S}$, the distance d between $\mathbf{s}$ and $\mathbf{w} \in \mathbf{W}$ is computed and the classification of $\mathbf{s}$ is based on equation (1).

$$f(\mathbf{s}) = \arg\min_{k} \; d(\mathbf{s}, \mathbf{w}_k), 1 \leq k \leq M \tag{1}$$

where the cardinality of the prototype set $\mathbf{W}$ is indicated by M. The prototypes used in NPCs can be learned alongside the dissimilarity measure, leading to good generalization ability and robustness[11]. Regarding learning vector quantization, a subset of NPCs [20], correct classification can only be attained when $d(\mathbf{s}, \mathbf{w}^*) < d(\mathbf{s}, \mathbf{w}_*)$ where $\mathbf{w}^*$ and $\mathbf{w}_*$ are the prototypes that best correctly assign and incorrectly assign a class c to $\mathbf{s}$ respectively[21, 22].

Definition 2.1 A mapping $\mathbf{S}\times\mathbf{S} \rightarrow \mathbb{R}$ is a semi-metric if the following properties are satisfied $\forall\ \mathbf{s}, \tilde{\mathbf{s}}, \bar{\mathbf{s}} \in \mathbf{S}$: $d(\mathbf{s}, \tilde{\mathbf{s}}) \geq 0$ (non-negativity); $d(\mathbf{s}, \tilde{\mathbf{s}}) = d(\tilde{\mathbf{s}}, \mathbf{s})$ (symmetry); $d(\mathbf{s}, \tilde{\mathbf{s}}) \leq d(\mathbf{s}, \bar{\mathbf{s}} + d(\bar{\mathbf{s}}, \tilde{\mathbf{s}})$ (triangle inequality). A semi-metric with additional property that $d(\mathbf{s}, \tilde{\mathbf{s}}) = 0 \implies \mathbf{s} = \tilde{\mathbf{s}}$ is a metric.

Definition 2.2 A mapping $\mathbf{S} \times \mathbf{S} \rightarrow \mathbb{R}$ is a semi-norm if the following properties are satisfied $\forall\ \mathbf{s}, \tilde{\mathbf{s}} \in \mathbf{S}: ||\mathbf{s}|| \geq 0$ (non-negativity); $||\alpha\mathbf{s}|| =| \alpha \mid ||\mathbf{s}||$ (absolute homogeneity); $||\mathbf{s} + \tilde{\mathbf{s}}|| \leq ||\mathbf{s}|| + ||\tilde{\mathbf{s}}||$ (triangle inequality). A semi-norm with additional property that $||\mathbf{s}|| = 0 \implies \mathbf{s} = 0$ is a norm.

3 Prototypical Metric for Data Pruning

Data pruning remains a relevant step for data preprocessing when dealing with large datasets, especially in areas where the only room for optimization of learner performance relies heavily on the size of the training set. A considerable number of data pruning metrics used in deep learning do not only come with a high computational burden but also necessitate a full input space along with its corresponding target space. A more efficient strategy is to employ an interpretable, comprehensible and scalable data pruning metric that significantly reduces computational demands, as proposed by Wong in 2018 [5]. Hence, the data pruning metric that satisfies the aforementioned conditions will be a self-supervised prototype-based metric (SSL) that can effectively prune the input space into *easy* and *hard* instances with or without any target space[14]. In this regard, we opt for a prototypical metric that avoids heuristics by utilizing a unity hyper-parameter for optimization. Given an input set $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, .., \mathbf{s}_n\} \subseteq \mathbb{R}^m$, let the objective function $J_{SSL} : M_c \times \mathbb{R}^{cm} \rightarrow \mathbb{R}^+$ for the self-supervised prototype-based metric be defined as

$$J_{SSL}(U, \mathbf{v}) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_i(\mathbf{s}_k)) \parallel \mathbf{s}_k - \mathbf{v}_i \parallel^2 \tag{2}$$

together with a Hard c-partition space for $\mathbf{S}$ given by (3) where H_{cn} is the set of real $c \times n$ matrices[23],

$$M_c = \left\{ U \in H_{cn} \;\middle|\; u_i\left(\mathbf{s}_k\right) \in \{0,1\} \,\forall\, i,k \,;\, \sum_{i=1}^{c} u_i\left(\mathbf{s}_k\right) = 1 \,\forall\, k \,;\, 0 < \sum_{k=1}^{n} u_i\left(\mathbf{s}_k\right) < n \,\forall\, i \right\} \tag{3}$$

We stipulate that $u_i\left(\mathbf{s}_k\right) = 1 \iff \mathbf{s}_k \in u_i$ and hence J_{SSL} in (2) reduces to the simplified form,

$$J_{SSL}\left(U, \mathbf{v}\right) = \sum_{i=1}^{c} \left(\sum_{\mathbf{s}_k \in u_{ik}} \parallel \mathbf{s}_k - \mathbf{v}_i \parallel^2 \right) \tag{4}$$

$$J_{SSL}\left(U, \mathbf{v}\right) = \sum_{i=1}^{c} \left[\sum_{\mathbf{s}_k \in u_{ik}} \left(\sum_{j=1}^{m} \left(\mathbf{s}_{kj} - \mathbf{v}_{ij}\right)^2 \right) \right] \tag{5}$$

we minimize globally J_{SSL} in (5) for the cluster prototypes $\mathbf{v}_i$ and construct the distance space D_{ik} from which we determine the *easy* (6) and *hard* (7) instances based on the magnitude of closeness to the cluster prototypes [5, 23] with regard to a specified prune ratio r.

$$easy = r \left[\min_{\mathbf{s}_k \to \mathbf{v}_i} D_{ik} \right],\ D_{ik} \in \mathbb{R}^{c \times n} \quad 1 \leq i \leq c, \quad 1 \leq k \leq n \tag{6}$$

$$hard = (1 - r) \left[\max_{\mathbf{s}_k \to \mathbf{v}_i} D_{ik} \right],\ D_{ik} \in \mathbb{R}^{c \times n} \quad 1 \leq i \leq c, \quad 1 \leq k \leq n \tag{7}$$

When $\mathbf{S}$ is chosen as an image dataset, best practices entail using a fine-tuned or pre-trained model to embed the input space features as an initial step before utilizing the SSL metric for the designated data pruning based on the embedded space as inputs [24]. Hence considering the squared Euclidean distance regarding any two arbitrary images $(\mathbf{x}, \mathbf{y}) \in \mathbf{S}$:

$$d^2\left(\mathbf{x}, \mathbf{y}\right) = \sum_{i=1}^{m} \left(\mathbf{x}_i - \mathbf{y}_i\right)^2 \tag{8}$$

a perceptual metric is obtained by computing the Euclidean distance between normalized feature activations in a neural network[12]. Given a fixed neural network ζ with I layers, let the i-th layer input specifications for the height, width and number of channels be indicated as H_i, B_i, C_i respectively. Consider $\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \;\forall\, i \in I$ as output features of ζ, the distance measure in (8) for the normalized by channel-level feature per hidden layer is given by:

$$d^2\left(\mathbf{x}, \mathbf{y}\right) = \sum_{i \in I} \sum_{h,b} \left(\frac{\mathbf{x}_{h,b}^{(i)}}{\sqrt{\sum_{j=1}^{C_i} (\mathbf{x}_{h,b}^{(i)})^2}} - \frac{\mathbf{y}_{h,b}^{(i)}}{\sqrt{\sum_{j=1}^{C_i} (\mathbf{y}_{h,b}^{(i)})^2}} \right)^2 \tag{9}$$

with $h \in H_i$ and $b \in B_i$. Considering the normalization by size of the i-th layer of (9) along with p_i indicating the learned weights from human perception data[12],

$$d^2\left(\mathbf{x}, \mathbf{y}\right) = \sum_{i \in I} \sum_{h,b} \frac{1}{H_i B_i} \left[p_i \odot \left(\frac{\mathbf{x}_{h,b}^{(i)}}{\sqrt{\sum_{j=1}^{C_i} (\mathbf{x}_{h,b}^{(i)})^2}} - \frac{\mathbf{y}_{h,b}^{(i)}}{\sqrt{\sum_{j=1}^{C_i} (\mathbf{y}_{h,b}^{(i)})^2}} \right) \right]^2 \tag{10}$$

Hence for the Learned Perceptual Image Patch Similarity (LPIPS) distance [25, 26] we have the form:

$$d^2\left(\mathbf{x}, \mathbf{y}\right) = \sum_{i \in I} \sum_{h,b} \left[p_i \odot \left(\frac{\mathbf{x}_{h,b}^{(i)}}{\sqrt{\sum_{j=1}^{C_i} (\mathbf{x}_{h,b}^{(i)})^2} \sqrt{H_i B_i}} - \frac{\mathbf{y}_{h,b}^{(i)}}{\sqrt{\sum_{j=1}^{C_i} (\mathbf{y}_{h,b}^{(i)})^2} \sqrt{H_i B_i}} \right) \right]^2 \tag{11}$$

and for a given input $\mathbf{x}$, the feature space embedding is normalized and flattened with regards to the mapping function ψ

$$\psi\left(\mathbf{x}\right) = \frac{\mathbf{x}_{h,b}^{(i)}}{\sqrt{\sum_j (\mathbf{x}_{h,b}^{(i)})^2} \sqrt{H_i B_i}} \quad i \in I, \;\; 1 \leq j \leq C_i \tag{12}$$

hence when p_i attains unity $\forall\, i \in I$, (11) becomes (13)

$$d^2\left(\mathbf{x},\mathbf{y}\right) = \|\psi\left(\mathbf{x}\right) - \psi\left(\mathbf{y}\right)\|_2^2\,, \quad \mathbf{x},\mathbf{y} \in \left[0,1\right]^d\;, \tag{13}$$

realizing a distance that is a semi-metric[25]. We can now replace the squared Euclidean distance in (5) with the LPIPS distance in (13) and obtain a Perceptual-SSL data pruning metric as:

$$J_{PSSL}\left(U,\mathbf{v}\right) = \sum_{i=1}^{c}\left[\sum_{\mathbf{s}_k \in u_{ik}}\left(\sum_{j=1}^{m}\left(\psi(\mathbf{s}_{kj}) - \psi(\mathbf{v}_{ij})\right)^2\right)\right] \tag{14}$$

The global minimization of (14) along with (6-7) leads to intelligently selected instances for NPCs regarding complex image datasets for which the LPIPS distance outperforms the distance measure induced by the L^2-norm[27].

4 Robust Nearest Prototype Classifiers

We consider the *robustification* of the LVQ family of classifiers, [21, 22] and narrow our scope to the Generalized LVQ [9] inspired learners. To this end, we define the training pair $T = \{\mathbf{s}_n, c\left(\mathbf{s}_n\right)\}_{n=1}^{N} \in \{\mathbb{R}^m, \mathcal{C}\}^N$ and a set of vectors $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\} \subseteq \mathbb{R}^m$ referred to as prototypes, to which each class $c \in \mathcal{C}$ is assigned one or more prototypes. The search for optimal learner performance involves minimizing (15) with respect to the prototypes $\mathbf{W}$.

$$E_{GLVQ}\left(\mathbf{s}, t_\beta\right) = \sum_{i=1}^{N} t_\beta\left(\mu\left(\mathbf{s}_i\right)\right) = \sum_{i=1}^{N} t_\beta\left(\frac{d\left(\mathbf{s}_i, \mathbf{w}^+\right) - d\left(\mathbf{s}_i, \mathbf{w}^-\right)}{d\left(\mathbf{s}_i, \mathbf{w}^+\right) + d\left(\mathbf{s}_i, \mathbf{w}^-\right)}\right) \tag{15}$$

The differentiable dissimilarity $d^+\left(\mathbf{s}\right) = d\left(\mathbf{s}, \mathbf{w}^+\right)$ and $d^-\left(\mathbf{s}\right) = d\left(\mathbf{s}, \mathbf{w}^-\right)$ used in (15) and defined by (2.2) indicates the correct and incorrect best matching distance respectively based on $\{\mathbf{w}^+, \mathbf{w}^-\} \in \mathbf{W}$ [9]. The activation function $t_\beta\left(k\right), k \in \mathbb{R}$ is conditioned to be monotonically increasing given an example as:

$$t_\beta\left(k\right) = \left(1 + e^{\left(-\beta k\right)}\right)^{-1} \tag{16}$$

The classifier function $\mu\left(\mathbf{s}\right)$ in (15) lies in the interval $\left[-1, 1\right]$ where correct classification indicates $d^+\left(\mathbf{s}\right) < d^-\left(\mathbf{s}\right)$ and vice-versa. Considering a minimal adversarial perturbation φ limited to a specified positive bound ϵ, an adversarial example is defined by:

$$\hat{\mathbf{s}} = \mathbf{s} + \varphi, \quad 0 < \varphi \leq \epsilon, \quad c\left(\mathbf{s}\right) \neq c\left(\mathbf{w}_{f(\hat{\mathbf{s}})}\right) \tag{17}$$

for which φ is the upper bound of the sample margin, which in turn upper bounds the hypothesis margin [8, 11, 10]. Therefore the robustification lies in the fact that GLVQ-loss function is a normalized form of the triplet loss [8, 9] and the minimization of (15) by the stochastic gradient descent optimization leads to the realization of a proven robust leaner [10, 11]. Hence regarding the *robust test error* of an L^p-GLVQ against a given ϵ-limited L^q adversarial attack, the upper bound is given by:

$$\mathit{URTE}_{q\leq p} = \frac{1}{|T|} \cdot |\{(\mathbf{s}, c(\mathbf{s})) \in T \mid d^-\left(\mathbf{s}\right) - d^+\left(\mathbf{s}\right) \leq 2\epsilon\}| \tag{18}$$

and for attack scenarios where $q > p$ regarding the test and train norm, the size of minimal adversarial perturbation φ has a bound determined by the Hölder's inequality[11] and hence the upper bound in (18) becomes (19).

$$\mathit{URTE}_{q>p} = \frac{1}{|T|} \cdot |\{(\mathbf{s}, c(\mathbf{s})) \in T \mid \; m^{\frac{1}{q}-\frac{1}{p}}\left(d^-\left(\mathbf{s}\right) - d^+\left(\mathbf{s}\right)\right) \leq 2\epsilon\}| \tag{19}$$

In some cases, examples denoted as $\mathbf{s} \in T$ may evade misclassification under a selected ϵ-limited adversarial attack. This is because the search for the minimal adversarial perturbation φ in these attacks does not guarantee the discovery of an adversary for every example under attack[11, 28]. Therefore, the absence of an adversary raises uncertainty about whether there is genuinely no adversary or if the method has simply failed[28]. Consequently, the corresponding lower bound on the *robust test error* is thus computed by:

$$\mathit{LRTE} = \frac{1}{|T|} \cdot |\{(\mathbf{s}, c(\mathbf{s})) \in T \,|\, \hat{\mathbf{s}} = \mathbf{s} + \varphi, \quad 0 < \varphi \leq \epsilon, \quad c\left(\mathbf{s}\right) \neq c\left(\mathbf{w}_{f(\hat{\mathbf{s}})}\right)\}| \tag{20}$$

The robustification and certification of GLVQ in (15) and (18-19) respectively involves the dissimilarity measure $d^*\left(\mathbf{s}\right)$ to be induced by a semi-norm[10, 11]. Consequently, when $d^*\left(\mathbf{s}\right)$ is replaced with the LPIPS distance [25] from (13) in Section (3), we attain a Perceptual-GLVQ (P-GLVQ) learner with the cost function given by:

$$E_{PGLVQ}\left(\psi(\mathbf{s}), t_{\beta}\right) = \sum_{i=1}^{N} t_{\beta}\left(\mu\left(\psi(\mathbf{s}_i)\right)\right) = \sum_{i=1}^{N} t_{\beta}\left(\frac{d\left(\psi(\mathbf{s}_i), \psi(\mathbf{w}^{+})\right) - d\left(\psi(\mathbf{s}_i), \psi(\mathbf{w}^{-})\right)}{d\left(\psi(\mathbf{s}_i), \psi(\mathbf{w}^{+})\right) + d\left(\psi(\mathbf{s}_i), \psi(\mathbf{w}^{-})\right)}\right) \tag{21}$$

Hence, regarding the minimization of (21), the robustification and certification of P-GLVQ involves using a semi-metric[12] as the dissimilarity measure. The same realization applies to the Generalized Tangent Learning Vector Quantization (GTLVQ), which is a GLVQ-loss-based classifier utilizing a tangent distance[10, 19]. Additional research is deemed necessary for Perceptual-GTLVQ (P-GTLVQ).

5 Experimentation

We utilized four datasets to examine the effect of optimal data pruning (utilizing prototype-based metrics), on the robustification and certification of GLVQ-loss-based learners. The datasets include two multi-class image datasets: MNIST handwritten data [29] and CIFAR-10 [30], along with two tabular binary class datasets: WDBC [31] and COD-RNA [32].

The results of GLVQ and GTLVQ models herein referred to as robust NPCs trained with L^p-Norms against L^q-threat models based on unpruned and pruned training datasets is presented in Tables (1,2,3). The ϵ-thresholds used by the LRTE and URTE evaluation metrics in (20) and (18-19), respectively, were chosen to correspond with practical threat models informed by the findings presented in ([11]) and ([12]). The L^1 and L^∞-threat models were derived from the Projected Gradient Descent attack [33], while the L^2-threat model was based on the Carlini & Wagner attack [34].

Table 1: Certified robustness of L^p-NPCs against L^q-threat models.

Dataset	Model	PPC	CTE	L^1		L^2		L^∞	
				LRTE	URTE	LRTE	URTE	LRTE	URTE
WDBC $\epsilon = 0.3$	L^2-GLVQ	2	5.26	6.14	8.77	21.93	30.70	92.11	100.00
	GTLVQ	2	8.77	14.91	19.30	53.51	89.47	99.12	100.00
COD-RNA $\epsilon = 0.025$	L^2-GLVQ	2	4.70	9.10	30.50	15.20	20.90	26.70	68.20
	L^∞-GLVQ	2	12.60	19.90	43.90	30.10	54.00	35.00	47.40
	GTLVQ	1	4.00	8.60	44.90	17.10	36.80	30.10	86.00
MNIST $\epsilon = 0.3$	L^2-GLVQ	2	6.88	7.12	17.13	11.50	19.00	99.87	100.00
	L^∞-GLVQ	1	20.13	31.25	63.88	75.25	97.38	83.00	100.00
	GTLVQ	1	5.00	5.00	28.38	9.13	44.37	100.00	100.00
CIFAR-10 $\epsilon = 8/225$	L^2-GLVQ	2	63.82	64.00	65.00	66.87	69.63	97.37	98.50
	GTLVQ	1	62.38	62.37	70.87	63.00	72.88	90.62	98.25

In Table (1), we present the results for L^p-GLVQ and GTLVQ based on the full-training dataset regarding the corresponding ϵ-limited L^q-adversarial attacks. The outcomes of L^2-GLVQ for the WDBC and CIFAR-10 datasets exhibit tight scores for both the LRTE and URTE, a pattern replicated in Table (3) when L^2-GLVQ is trained with both “easy” and “hard” instances of the training set, maintaining an 80% retention rate[1] for both pruning scenarios.

Observably, training with a significantly reduced number of intelligently selected instances using the SSL data pruning metric improves the evaluation results for clean test errors (CTE), with relatively minor discrepancies in the lower (LRTE) and upper bounds (URTE) on the robust test errors. The results of GTLVQ for the WDBC dataset in Tables (1) and (3) indicate enhanced evaluation performance when trained with pruned datasets. Similarly, the results of GTLVQ for CIFAR-10 demonstrate consistent behavior over pruned and fully trained datasets. Therefore, concerning the CTE, training with “easy” instances yields improved performance compared to training with “hard” instances. In contrast, the performance remains sustained for both the LRTE and URTE evaluation metrics. From Tables (1) and (3),

[1] The decision to retain a constant pruning ratio set at 0.8 of the training data aligns with the findings of Sorscher et al.[14], a correlation we verified through empirical experimentation involving various retention rates.

Table 2: Certified robustness of L^p-NPCs against L^q-threat models for the MNIST and CIFAR-10 datasets, each pruned utilizing the SSL metric, with the fixed pruning ratio set at 0.8.

Dataset	Prune Mode	Model	PPC	CTE	L^1		L^2		L^∞	
					LRTE	URTE	LRTE	URTE	LRTE	URTE
MNIST $\epsilon = 0.3$	*easy*	L^2-GLVQ	2	7.25	7.37	15.00	12.13	18.50	99.12	100.00
		L^∞-GLVQ	2	21.88	31.00	58.63	72.12	97.50	83.00	100.00
		GTLVQ	1	6.75	6.75	31.75	10.00	43.25	100.00	100.00
	hard	L^2-GLVQ	2	7.75	7.75	15.62	11.75	18.25	99.50	100.00
		L^∞-GLVQ	2	22.87	31.00	64.62	69.25	94.63	81.25	100.00
		GTLVQ	1	6.88	7.00	29.00	10.50	42.75	99.75	100.00
CIFAR-10 $\epsilon = 8/255$	*easy*	L^2-GLVQ	2	63.75	63.88	65.25	66.12	69.13	97.37	98.88
		GTLVQ	1	60.87	60.87	72.12	62.75	72.88	94.00	100.00
	hard	L^2-GLVQ	2	65.25	65.25	66.87	67.75	69.87	97.50	99.12
		GTLVQ	1	61.38	61.37	72.12	62.63	73.25	91.00	100.00

the results for the COD-RNA and MNIST datasets with regards to the L^2-GLVQ and L^∞-GLVQ models indicate sustained performance over the LRTE and URTE with fairly acceptable discrepancies for the CTE. Similarly, the GLVQ model for COD-RNA and MNIST datasets also exhibits behavior consistent with that observed for the L^2-GLVQ and L^∞-GLVQ models. Therefore, it is evident that the SSL metric employed for pruning the training set of COD-RNA and MNIST datasets does not compromise the performance of the models in terms of the CTE, as well as the LRTE and URTE. Thus, the employed SSL data pruning metric achieves a similar outcome as observed by Sorscher et al.[14].

Table 3: Certified robustness of L^p-NPCs against L^q-threat models for the WDBC and COD-RNA datasets, each pruned utilizing the SSL metric, with the fixed pruning ratio set at 0.8.

Dataset	Prune Mode	Model	PPC	CTE	L^1		L^2		L^∞	
					LRTE	URTE	LRTE	URTE	LRTE	URTE
WDBC $\epsilon = 0.3$	*easy*	L^2-GLVQ	2	4.39	6.14	8.77	21.05	32.46	92.11	100.00
		GTLVQ	2	7.02	14.04	16.67	55.26	84.21	100.00	100.00
	hard	L^2-GLVQ	2	5.26	6.14	8.77	20.18	29.82	92.11	100.00
		GTLVQ	2	10.53	13.16	16.67	66.67	71.05	100.00	100.00
COD-RNA $\epsilon = 0.3$	*easy*	L^2-GLVQ	2	5.10	8.70	32.20	16.20	20.70	26.00	67.20
		L^∞-GLVQ	2	13.40	20.80	33.30	31.70	51.60	38.00	48.60
		GTLVQ	1	5.50	8.90	39.10	16.40	38.70	31.70	85.10
	hard	L^2-GLVQ	2	5.30	9.30	31.60	18.20	23.10	27.40	67.70
		L^∞-GLVQ	2	13.90	21.30	35.40	33.00	48.50	38.60	52.30
		GTLVQ	1	5.00	9.20	39.60	15.70	37.30	28.40	69.70

From Table (4), the CIFAR-10 dataset results demonstrate enhanced performance scores for the CTE, LRTE and URTE when utilizing the Perceptual-SSL (P-SSL) data pruning metric and the Perceptual-GLVQ (P-GLVQ) learner. The observed improvement is evident when compared to the results obtained from pruning the CIFAR-10 dataset with the SSL data pruning metric for L^p-GLVQ, as depicted in Tables (1-2).

Notably, the CTE and LRTE evaluation metric scores remain remarkably tight compared to those derived from learning scenarios without perceptual-metric data pruning and perceptual-metric robustification. Additionally, the unpruned dataset results of P-GLVQ for CIFAR-10 surpass those of L^p-GLVQ for the same dataset. Hence, the application of optimal data pruning metrics (SSL and P-SSL) for the training and evaluation of robust NPCs has the potential to improve the performance of the learners across generalization, empirical robustness and guaranteed robustness in best-case scenarios and maintain the performance in the same regard in worst-case scenarios.

[2] It is crucial to note that, in the conducted experiments, all datasets were pruned based on the SSL data pruning metric. Additionally, for CIFAR-10, the P-SSL data pruning metric was specifically applied. The Python implementation for the SSL and P-SSL data pruning metrics, as well as the robustification and certification of NPCs discussed in this paper, can be found at `https://github.com/naotoo1/BNSFRNPC`.

Table 4: Certified robustness of P-GLVQ against L^q-threat models for CIFAR-10 dataset pruned utilizing the P-SSL metric with a fixed pruning ratio set at 0.8. Dash " $-$ " indicates the dataset was not pruned.

Dataset	Prune Mode	ϵ	PPC	CTE	L^1		L^2		L^∞	
					LRTE	URTE	LRTE	URTE	LRTE	URTE
CIFAR-10	$-$	$2/255$	2	41.50	41.62	46.00	42.75	47.25	84.00	100.00
		$8/255$	2	41.50	41.75	46.37	48.25	62.62	100.00	100.00
	easy	$2/255$	2	41.00	41.00	45.50	44.62	48.00	85.25	100.00
		$8/255$	2	41.00	41.62	47.50	47.62	62.50	100.00	100.00
	hard	$2/255$	2	40.62	40.62	47.75	42.12	46.37	85.50	100.00
		$8/255$	2	40.62	40.62	48.00	48.12	62.62	100.00	100.00

6 Discussion

The experimental outcomes in this paper indicate that training and evaluation of robust NPCs can be much improved when combined with optimal data pruning metrics spirited on self-supervised prototype-based models. The integration of the SSL and P-SSL data pruning metrics effectively solves the challenges posed by power laws in the context of NPCs. Furthermore, the optimal data pruning herein introduced is characterized by simplicity, comprehensibility, and interpretability. Thus, applying data pruning metrics for NPCs reduces the amount of data used for training even when the full set of training parameters are maintained. Consequently, reducing the amount of training time without compromising performance. Therefore, the practical application of an effective prototypical data pruning metric becomes valuable in training scenarios where practitioners face challenges in evaluating the performance of robust NPCs, particularly when dealing with large datasets.

The data pruning metrics introduced in this paper aid in intelligently gathering data subsets, ultimately creating foundational datasets for learning purposes. Additionally, the method described in this paper guarantees satisfactory performance across all evaluation metrics (CTE, LRTE and URTE), even when training with NPCs of reduced model complexity (few prototypes per class).

It is crucial to emphasize that the methodology detailed in this paper, relying on L^p-distances and L^q-threat models, has limitations when applied to the training and robustness evaluation of image datasets like CIFAR-10, especially in the presence of unforeseen threat models[35]. Interestingly, our demonstrated success with the Perceptual-GLVQ (P-GLVQ) model underscores its effectiveness for image datasets such as CIFAR-10, surpassing the performance of both L^p-GLVQ and GTLVQ models.

7 Conclusion

A novel data pruning metric, surpassing power-law approaches, has been introduced to enhance the robustification and certification of NPCs trained with L^p-norms and Perceptual-metrics. This work represents the first instance of employing pioneering data pruning metrics (SSL and P-SSL) for hardening L^p-NPCs and Perceptual-metric NPCs within the LVQ family of classifiers against L^q-threat models. Subsequent research will explore the use of class-level prototypical optimal pruning metrics as a seamless transition to initializing terminal prototypes to enhance the robustification and certification of NPCs. In pursuit of this goal, we aim to investigate more expansive and realistic non-L^q threat models within the domain of image datasets, incorporating insights from neural perceptual threat models [26]. As a result, our study will delve into strategies for perceptual robustification and certification tailored for robust Nearest Prototype Classifiers (NPCs), specifically focusing on LVQs and placing significant emphasis on Perceptual-GTLVQ (P-GTLVQ).

8 Acknowledgment and Disclosure

The researchers assert that there are no conflicts of interest associated with this work.

References

[1] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE symposium on security and privacy (SP)*, pages 582–597. IEEE, 2016.

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

[3] Vincent Tjeng, Kai Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. *arXiv preprint arXiv:1711.07356*, 2017.

[4] Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. *Advances in neural information processing systems*, 30, 2017.

[5] Eric Wong, Frank Schmidt, Jan Hendrik Metzen, and J Zico Kolter. Scaling provable adversarial defenses. *Advances in Neural Information Processing Systems*, 31, 2018.

[6] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

[7] Alfredo Vellido. The importance of interpretability and visualization in machine learning for applications in medicine and health care. *Neural computing and applications*, 32(24):18069–18083, 2020.

[8] Koby Crammer, Ran Gilad-Bachrach, Amir Navot, and Naftali Tishby. Margin analysis of the lvq algorithm. *Advances in neural information processing systems*, 15, 2002.

[9] Atsushi Sato and Keiji Yamada. Generalized learning vector quantization. *Advances in neural information processing systems*, 8, 1995.

[10] Sascha Saralajew, Lars Holdijk, Maike Rees, and Thomas Villmann. Robustness of generalized learning vector quantization models against adversarial attacks. In *Advances in Self-Organizing Maps, Learning Vector Quantization, Clustering and Data Visualization: Proceedings of the 13th International Workshop, WSOM+ 2019, Barcelona, Spain, June 26-28, 2019 13*, pages 189–199. Springer, 2020.

[11] Sascha Saralajew, Lars Holdijk, and Thomas Villmann. Fast adversarial robustness certification of nearest prototype classifiers for arbitrary seminorms. *Advances in Neural Information Processing Systems*, 33:13635–13650, 2020.

[12] Václav Voráček and Matthias Hein. Provably adversarially robust nearest prototype classifiers. In *International Conference on Machine Learning*, pages 22361–22383. PMLR, 2022.

[13] Marika Kästner, Marc Strickert, Thomas Villmann, and Saxonia-Germany Mittweida. A sparse kernelized matrix learning vector quantization model for human activity recognition. In *ESANN*, 2013.

[14] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

[15] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

[16] Jonathan S Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales. *arXiv preprint arXiv:1909.12673*, 2019.

[17] Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*, 2021.

[18] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[19] Sascha Saralajew and Thomas Villmann. Adaptive tangent distances in generalized learning vector quantization for transformation and distortion invariant classification learning. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 2672–2679. IEEE, 2016.

[20] Teuvo Kohonen and Teuvo Kohonen. Learning vector quantization. *Self-organizing maps*, pages 175–189, 1995.

[21] James C Bezdek and Ludmila I Kuncheva. Nearest prototype classifier designs: An experimental study. *International journal of Intelligent systems*, 16(12):1445–1473, 2001.

[22] Michael Biehl, Barbara Hammer, and Thomas Villmann. Prototype-based models in machine learning. *Wiley Interdisciplinary Reviews: Cognitive Science*, 7(2):92–111, 2016.

[23] James C Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media, 2013.

[24] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

[25] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[26] Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. *arXiv preprint arXiv:2006.12655*, 2020.

[27] Klim Kireev, Maksym Andriushchenko, and Nicolas Flammarion. On the effectiveness of adversarial training against common corruptions. In *Uncertainty in Artificial Intelligence*, pages 1012–1021. PMLR, 2022.

[28] Lu Wang, Xuanqing Liu, Jinfeng Yi, Zhi-Hua Zhou, and Cho-Jui Hsieh. Evaluating the robustness of nearest neighbor classifiers: A primal-dual perspective. *arXiv preprint arXiv:1906.03972*, 2019.

[29] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[30] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[31] Catherine L Blake. Uci repository of machine learning databases. *http://www. ics. uci. edu/˜ mlearn/MLRepository. html*, 1998.

[32] Andrew V Uzilov, Joshua M Keegan, and David H Mathews. Detection of non-coding rnas on the basis of predicted secondary structure formation free energy change. *BMC bioinformatics*, 7(1):1–30, 2006.

[33] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[34] Nicholas Carlini. Is ami (attacks meet interpretability) robust to adversarial examples? *arXiv preprint arXiv:1902.02322*, 2019.

[35] Max Kaufmann, Daniel Kang, Yi Sun, Steven Basart, Xuwang Yin, Mantas Mazeika, Akul Arora, Adam Dziedzic, Franziska Boenisch, Tom Brown, et al. Testing robustness against unforeseen adversaries. *arXiv preprint arXiv:1908.08016*, 2019.