KAN-Robust-Bench: A Benchmark for Evaluating the Robustness of Kolmogorov-Arnold Networks

Mohammad Meymani, Roozbeh Razavi-Far

Trustworthy and Secure AI (TSAI) Lab, Faculty of Computer Science, University of New Brunswick, Fredericton, Canada

Abstract

While machine learning models have demonstrated strong performance in many domains, these models have shown profound vulnerabilities when they are exposed to adversarial threats. While adversarial attacks fall into various categories, the most prominent category in research studies is evasion. In evasion attacks, the adversary generates perturbed versions of samples, which might not be observable by human eyes. These samples generally fool the machine learning models with high confidence. This phenomenon poses a significant security violation against machine learning models. In this paper, we investigate the certified and empirical robustness of various Kolmogorov-Arnold network architectures against strong evasion attacks. At first, we provide the mathematical foundations for randomized smoothing and interval bound propagation, and report the ℓ_2-certified robustness of the models under randomized smoothing. After that, we systematically evaluate the robustness of various defended and undefended KAN models under FGSM, PGD, and C&W attacks in order to find out the optimal defense strategies and architectures.

Keywords: Adversarial Machine Learning, Certified Robustness, Evasion Attacks, Adversarial Training, Randomized Smoothing, and Kolmogorov-Arnold Networks.

Email addresses: mohammad.meymani79@unb.ca (Mohammad Meymani), roozbeh.razavi-far@unb.ca (Roozbeh Razavi-Far)

1. Introduction

Machine learning (ML) has extended artificial intelligence (AI) by moving systems from hard-coded rules to data-driven learning. ML has facilitated numerous tasks in a huge number of fields including retail, finance, natural language processing (NLP), autonomous driving, computer vision (CV), healthcare, cybersecurity, and many other domains Shinde and Shah (2018); Sarker (2021). ML models have evolved during decades from classical networks and multi-layer perceptron (MLP) to deep neural networks. Each learning paradigm aims to solve the problems that are infeasible, hard, or inefficient for the previous ones Boutaba et al. (2018). Kolmogorov-Arnold networks (KANs) have been introduced to substitute MLP models in numerous areas Liu et al. (2025); Ji et al. (2024); Somvanshi et al. (2025).

Kolmogorov-Arnold representation theorem (KAT) states that any continuous multivariate function f can be written as a composition of finite number of continuous univariate functions. This can be formalized as follows:

$$f(x_1,\ x_2,\ ...,\ x_n) = \sum_{j=1}^{m} \psi_j(\sum_{i=1}^{n} \phi_{ij}(x_i)), \tag{1}$$

where n is the number of inputs, m is the number of compositions, x_i is the i^{th} input, ϕ_{ij} is the univariate function for x_i in j_{th} composition, and ψ_j is the j^{th} composition. KANs make use of KAT to provide powerful alternatives to MLP models Ji et al. (2024); Somvanshi et al. (2025); Lou et al. (2026).

KANs and MLPs differ in multiple aspects. First, KANs can outperform MLPs on specific tasks, including time-series analysis, with fewer trainable parameters. Second, in general, the training time for a KAN is significantly slower than that of an MLP on the same task Ji et al. (2024); Somvanshi et al. (2025). Third, KANs exhibit greater robustness to catastrophic forgetting in specific low-dimensional tasks than MLPs, making them suitable options for continual learning Cacciatore et al. (2024); Rahman et al. (2026). Figure 1 illustrates a simple architecture for a KAN, where traditional activation functions in MLPs are replaced by learnable ones. After each hidden layer, every connected Spline is summed together and sent to the next layer.

Adversarial machine learning (AML) is an interdisciplinary field, studying the vulnerabilities of ML models. Adversarial attacks exploit these vulnerabilities to harm utility, privacy, or explainability of ML models. Adversarial defenses, on the other hand, aim to secure ML models during different stages of ML lifecycle. This field faces an evolving arm-race, where new attacks

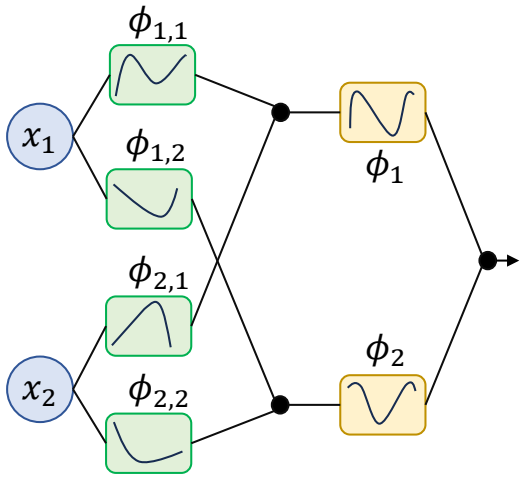

Figure 1: A simple KAN architecture consisting of four Splines at the first hidden layer and two Splines at the second hidden layer.

and defenses are introduced periodically. This necessitates the researchers to consider designing built-in defense systems when developing ML models Vashagh et al. (2026); Kurakin et al. (2017); Meymani and Razavi-Far (2026).

Adversarial attacks are divided into three major categories: evasion, poisoning, and exploratory attacks. In evasion attacks, the attacker generates adversarial samples by adding subtle and carefully crafted noise to the original data. These adversarial samples are imperceptible to human observers, but can fool the ML model with high confidence. Poisoning attacks aim to corrupt the training data by tampering with features and/or labels to harm the training process. Finally, exploratory attacks aim to violate the privacy of ML models by inferring their parameters and/or training data Vashagh et al. (2026); Meymani et al. (2026).

Moreover, an adversary can possess different amounts of knowledge about the target model's sensitive information including gradients, training data, and parameters. If an adversary has perfect knowledge about the model, it is considered as a white-box adversary. On the other hand, an adversary with zero information is considered as a black-box one. Finally, a gray-box adversary sits between white-box and black-box, where partial knowledge about the target model is available Vashagh et al. (2026); Meymani et al. (2026).

Certified robustness in AML refers to a formal and mathematically prov-

able guarantees that a model's prediction will remain unchanged within a specified perturbation bound and norm. There are various techniques that provide certified robustness by analyzing the robustness from a different perspective. These techniques include randomized smoothing (RS), interval bound propagation (IBP), convex relaxation, verification, and semidefinite programming (SDP) relaxation Cohen et al. (2019); Gowal et al. (2018); Meng et al. (2022); Zhang (2020); Salman et al. (2019); Raghunathan et al. (2018).

In this paper, we focus on white-box evasion attacks since they provide a direct means of robustness evaluation of the trained models during inference. This allows us to study the vulnerability of the learnt models when they are exposed to perturbed inputs. Moreover, evasion attacks are among the most widely studied adversarial threats, making them a suitable basis for systematically comparing the robustness of different KAN architectures and defense strategies. Our contributions are as follows:

- We evaluate the empirical robustness of state-of-the-art KAN-based computer vision architectures against strong white-box evasion attacks.
- We systematically evaluate the impact of adversarial training, randomized smoothing, and interval bound propagation on the robustness of KAN architectures across CIFAR-10 and SVHN.
- We analyze the certified robustness under randomized smoothing, and report certified accuracy and certified radius across different KAN architectures and perturbation bounds.

The remainder of the paper is as follows. In Section 2, we introduce the related research studies. In Section 3, we discuss the problem settings. In Section 4, we explain our methodology by providing defense algorithms and mathematical underpinning for RS and IBP. In Section 5, we discuss experimental setups, and demonstrate the certified and empirical results. Finally, in Section 6, we suggest future research directions and conclude our work.

2. Related Works

KANs are used for a variety of regression and classification tasks, including time series analysis, graph learning, signal processing healthcare, and CV. For CV, architectures such as KAN-Mixers Canuto et al. (2025), KANICE Ferdaus et al. (2024), and PoolKANNeXt Lv et al. (2025) have been proposed.

KAN-Mixers Canuto et al. (2025) extends MLP-mixer Tolstikhin et al. (2021) architecture by replacing the standard MLP-based layers with KAN layers. Similar to MLP-Mixer, KAN-Mixers performs token- and channel-mixing to learn both spatial and feature-level relationships. KAN-Mixers consists of a patch-embedding layer, mixer blocks, an adaptive average pooling layer, and a classification head. The results indicated that KAN-Mixers achieved the highest accuracy on Fashion MNIST and CIFAR-10 datasets compared to a standard KAN, MLP, and MLP-Mixer.

KANICE Ferdaus et al. (2024) is a hybrid architecture that combines KAN and CNN. KANICE consists of Interactive Convolution Block (ICB), standard convolutional layers, batch normalization, pooling layers, and KAN linear layers. ICBs are responsible for capturing spatial relationships, serving as initial feature extractor. Standard convolutional layers are used after each ICB to further process and extract the features. Batch normalization is used to stabilize the learning process, and pooling layers are used to reduce the dimensionality. Finally KAN linear layers are used to approximate complex functions more accurately. The results on MNIST, Fashion MNIST, EMNIST, and SVHN demonstrated the superior performance of KANICE compared to CNN, CNN_KAN, ICB_CNN, and ICB_KAN architectures.

PoolKANNeXt Lv et al. (2025) is another hybrid KAN-CNN architecture, which combines pooling-based feature-mixing, dual-activation KAN blocks, and ConvNeXt-style Liu et al. (2022) hierarchical architecture. PoolKAN-NeXt consists of four stages: stem, main, downsampling, and output. The stem phase is the initial feature extractor. The main phase consists of average pooling, convolution blocks, and GELU and Swish activation functions. After that, the inputs from previous phases are downsampled and sent to the output layer. While this model was originally designed for biomedical image classification, it demonstrated strong performance in general CV-based classification tasks including CIFAR-10.

In Ostanin et al. (2024), the authors evaluated the robustness of a KAN model and a MLP model under Gaussian noise and adversarial attacks using MNIST dataset. The authors used Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) to generate adversarial samples. While KAN achieved slightly higher accuracy than MLP on clean data, its performance decreased more sharply under noise and attacks. MLP model consistently outperformed the KAN model in terms of accuracy, precision, recall, and F1-score under noise and attacks. This trend indicated greater adversarial vulnerability of KAN compared to MLP.

In Ibrahum et al. (2024), the authors compared the performance of some KAN-based architectures (i.e., KAN-Mixer and convolutional KAN) and MLP-based architectures (i.e., MLP-Mixer) under basic iterative method (BIM), FGSM, and Carlini and Wagner (C&W). Experiments were conducted on BTSD, CTSD, and GTSRB datasets. Moreover, the authors investigated adversarial training (AT) and randomized smoothing as defense strategies to further study the robustness. Results indicated that robustness depended on the architecture. KAN-Mixer generally outperformed MLP-Mixer in terms of robustness, while standard KAN generally demonstrated less robustness than standard MLP model. Randomized smoothing and adversarial training improved robustness across models, with randomized smoothing often providing the stronger defense.

In Djosic et al. (2024), the authors evaluated the robustness of an MLP and four KAN variants (linear KAN, Fourier KAN, Chebyshev KAN, and Jacobi KAN) under Gaussian noise, FGSM, and PGD attacks with MNIST as the dataset. MLP and linear KAN demonstrated the highest accuracies. Under noise MLP showed the greatest robustness, with linear KAN being the second best, while other KAN variants showed severe degradation. After applying FGSM, MLP showed the greatest robustness followed by the linear KAN. In contrast, Chebyshev KAN and Fourier KAN demonstrated greatest robustness under PGD, while MLP and linear KAN had the lowest scores.

In Dong et al. (2024), the authors analyzed the robustness of KANs for time series classification under PGD attack. The authors used UCR2018 dataset. The results indicated that KAN demonstrated similar or sometimes slightly better than MLP in terms of clean accuracy. Moreover, the KAN model showed greater adversarial robustness than the MLP model. As a result, the authors concluded that the greater robustness of the KAN model stems from its lower Lipschitz constant.

In Alter et al. (2024), the authors evaluated the robustness of fully connected convolutional KANs across a wide range of datasets and adversarial attacks, and compared them with convolutional neural networks (CNNs) and fully connected neural networks (FCNNs). The datasets included MNIST, Fashion MNIST, KMNIST, CIFAR-10, SVHN, and ImageNet. The attacks included both white-box (i.e., FGSM and PGD) and black-box (i.e., Square Attack) attacks. The results suggested that convolutional KAN models generally exhibited greater robustness compared to CNN models with comparable sizes. Moreover, the impact of spline order and number of knots depends on model scale; higher spline orders often improve robustness in small and

medium models, whereas lower spline orders can be more robust in large models. The authors claimed that optimizing the hyperparameters of KAN-based architectures could be a promising research direction for robustness studies.

In Schumacher et al. (2025), the authors evaluated the robustness of KAN, MLP, and CNN models under FGSM and PGD attacks on Fashion MNIST and CIFAR 10 datasets. The authors proposed GloroKAN to provide certified robustness, however, adversarially trained KAN outperformed their proposed approach.

Although previous studies have investigated the adversarial robustness of KANs, most of them primarily focused on comparisons between KAN-based models and conventional architectures such as MLPs or CNNs. In contrast, our study focuses on robustness within the KAN family, aiming to determine how different KAN-based architectures behave under the same adversarial conditions. We evaluate KAN-Mixers, KANICE, and PoolKANNeXt across CIFAR-10 and SVHN, considering multiple defense strategies and a wider range of attack configurations. We further compare adversarial training, randomized smoothing, and interval bound propagation, while complementing the empirical evaluation with certified robustness analysis. This enables a focused evaluation of how architectural and defense choices influence robustness among KAN-based models.

3. Problem Settings

There are a variety of attacks that use white-box knowledge to exploit the vulnerabilities of machine learning models. These attacks use the model's parameters to generate adversarial samples, for instance, FGSM Goodfellow et al. (2015) is a one-step attack, generating adversarial samples by moving toward the gradient of the loss function:

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y)), \tag{2}$$

where x_{adv} is the generated adversarial sample, x is a benign sample, y is the corresponding label of that sample, ϵ is the perturbation size, $L(\theta, x, y)$ is the loss function, $\nabla_x L$ is the gradient of loss function with respect to x, and θ denotes the model's parameters.

Another famous attack is PGD Madry et al. (2018), which is a multi-step attack unlike FGSM. PGD refines a random perturbation over iterations to find a desired perturbation causing misclassification. Eq. (3) shows how an

adversarial sample is generated using PGD:

$$x_{adv}^{i+1} = \prod_{x+\delta} \left(x_{adv}^{i} + \alpha \cdot sign\left(\nabla_x L(\theta, x, y) \right) \right), \tag{3}$$

where δ is the perturbation, α is step size, i is iteration number, x_{adv}^{i} is the adversarial sample during i^{th} iteration, and $\prod_{x+\delta}$ is the projection function, ensuring the input stays within perturbation bounds ($[x-\delta,\ x+\delta]$).

Additionally, C&W Carlini and Wagner (2017) uses the following objective function:

$$\begin{cases} \text{minimize} & D(x, x+\delta) + \lambda . f(x+\delta) \\ \text{subject to} & x+\delta \in [0,1]^n \end{cases} \tag{4}$$

where $D(x, x+\delta)$ is the distance between the benign and adversarial samples, n is the number of features (pixels), λ controls the trade-off between perturbation size and attack success rate, $f(x+\delta)$ is an objective function, ensuring that $x+\delta$ is misclassified by the model.

Adversarial training is a defense strategy that aims to learn representations of both benign samples and adversarial samples during the training session. This can either be done by augmenting adversarial samples during the training session, or defining new objective function that learns both representations. Although this strategy has been widely used, most of the methods face robustness-accuracy trade-off. Robustness-accuracy trade-off occurs when clean accuracy is sacrificed to some extent to increase the robustness of the model Meymani and Razavi-Far (2026); Xu et al. (2021); Nath et al. (2023).

Randomized smoothing generates a smoothed model by adding Gaussian noise to the inputs and and aggregating predictions via Monte Carlo sampling. In Monte Carlo voting, multiple noisy versions of the input are generated and the predictions are aggregated via majority voting Cohen et al. (2019); Nandi et al. (2023).

Interval bound propagation provides a certified defense strategy by propagating the input perturbation bounds through the whole network. During IBP-based training, the training loss equals to the linear combination of clean loss and IBP loss to control the trade-off between accuracy and robustness. The final loss ℓ_{Final} can be formulated as follows:

$$\ell_{\text{Final}} = (1-\lambda) \times \ell_{\text{Clean}} + \lambda \times \ell_{\text{IBP}}, \tag{5}$$

where ℓ_{Clean} is the standard loss function, ℓ_{IBP} is the IBP-based loss functions, and λ controls the trade-off between robustness and accuracy Gowal et al. (2018); Mao et al. (2024); Mirman et al. (2018).

4. Methodology

In this section, we formulate certified robustness, explain defense mechanisms, and formalize the threat model. Figure 2 shows an step-by-step methodology process, where we begin by choosing a dataset, an architecture, and a training strategy to evaluate the performance of the models against a diverse set of adversarial attacks.

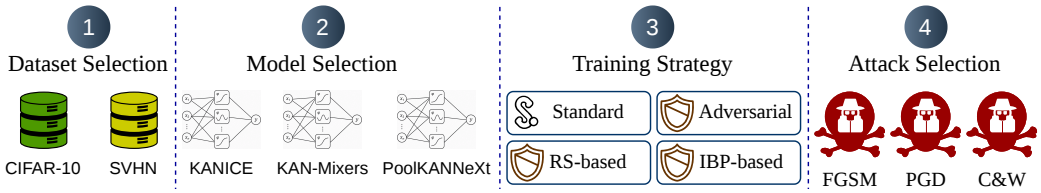

Figure 2: Step-by-step process of experimental methodology.

4.1. Certified Robustness Formulation

Unlike empirical defenses that rely on specific adversarial attacks and parameters, certified defenses provide mathematical guarantees on model behavior within predefined perturbation bounds Cohen et al. (2019); Nandi et al. (2023).

4.1.1. Randomized Smoothing

Let f denote a base KAN classifier. Randomized smoothing constructs a smoothed classifier by injecting Gaussian noise into the input space:

$$\eta \sim \mathcal{N}(0, \sigma^2 I), \tag{6}$$

where σ denotes the noise level and I is the identity matrix Cohen et al. (2019). The smoothed classifier is defined as:

$$g(x) = \arg \max_{c \in \{1,\ldots,C\}} P\left(f(x+\eta) = c\right), \tag{7}$$

where C is the number of classes. Let $p_A = P\left(f(x+\eta) = c_A\right)$ denote the probability of the most probable class c_A, and $p_B = \max_{c \neq c_A} P\left(f(x+\eta) = c\right)$ represent the probability of the second most probable class. According to the randomized smoothing certification theorem, the classifier prediction remains unchanged within an ℓ_2-ball of radius:

$$R = \frac{\sigma}{2}\left(\Phi^{-1}(p_A) - \Phi^{-1}(p_B)\right), \tag{8}$$

where $\Phi^{-1}(\cdot)$ is the inverse cumulative distribution function of the standard normal distribution Cohen et al. (2019). Therefore, for any perturbation δ satisfying $\|\delta\|_2 < R$, the prediction is guaranteed to remain unchanged:

$$g(x+\delta) = g(x). \tag{9}$$

During inference, class probabilities ($\hat{p}_c$) are estimated using Monte Carlo sampling:

$$\hat{p}_c = \frac{1}{M}\sum_{i=1}^{M} \mathbb{1}\big(f(x+\eta_i) = c\big), \tag{10}$$

where M is the number of noisy samples, $\mathbb{1}$ is an indicator function that outputs one for true conditions, and outputs zero for false ones, and $\eta_i \sim \mathcal{N}(0, \sigma^2 I)$. The certified accuracy of randomized smoothing is computed as:

$$CA_{RS}(r) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left(R_i \geq r \ \wedge \ y_i = \hat{y}_i\right), \tag{11}$$

where R_i is the certified ℓ_2 radius of sample i, N is the number of test samples, r is the threshold, y_i is the actual class label, and $\hat{y}_i$ is the predicted label.

4.1.2. Interval Bound Propagation

Interval Bound Propagation certifies robustness by propagating perturbation intervals through every layer of the network Gowal et al. (2018); Mao et al. (2024). Given an input sample x, the adversarial region is defined as:

$$\mathcal{X} = \{x + \delta : \|\delta\|_\infty \leq \epsilon\}, \tag{12}$$

where ϵ is the perturbation bound. The initial interval bounds are $l^{(0)} = x - \epsilon$ and $u^{(0)} = x + \epsilon$. For a linear transformation $z = Wh + b$, the interval bounds are propagated as:

$$l_z = W^+ l_h + W^- u_h + b, \tag{13}$$

$$u_z = W^+ u_h + W^- l_h + b, \tag{14}$$

where l_h represents the lower bound vector of the input layer, u_h represents the upper bound vector of the input layer, $W^+ = \max(W, 0)$, which keeps only

positive weights, and $W^- = \min(W, 0)$, which only keeps negative weights Gowal et al. (2018); Mirman et al. (2018).

Since KANs replace fixed activation functions with learnable spline functions, each KAN layer can be represented as:

$$y_j = \sum_{i=1}^{n} \phi_{ij}(x_i), \tag{15}$$

where ϕ_{ij} denotes a spline activation function. For an input interval $x_i \in [l_i, u_i]$ the lower and upper spline bounds are:

$$\underline{\phi}_{ij} = \min_{t \in [l_i, u_i]} \phi_{ij}(t), \tag{16}$$

$$\overline{\phi}_{ij} = \max_{t \in [l_i, u_i]} \phi_{ij}(t). \tag{17}$$

The output interval of a KAN neuron is therefore:

$$l_j = \sum_{i=1}^{n} \underline{\phi}_{ij}, \tag{18}$$

$$u_j = \sum_{i=1}^{n} \overline{\phi}_{ij}. \tag{19}$$

These bounds are recursively propagated through all KAN blocks until the final logit layer. Let l_k and u_k denote the lower and upper bounds of output k. For a sample with true class y, robustness is certified whenever:

$$l_y > \max_{k \neq y} u_k. \tag{20}$$

Under this condition, no perturbation satisfying $\|\delta\|_\infty \leq \epsilon$ can alter the classifier prediction Gowal et al. (2018); Mirman et al. (2018). The certified accuracy under IBP is then calculated as:

$$CA_{IBP}(\epsilon) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}\left(l_{y_i} > \max_{k \neq y_i} u_k \right). \tag{21}$$

Algorithm 1: Adversarial Training.

Input: Training data: X_{train}, training labels: Y_{train}, perturbation bound: ϵ, step size: α, iteration number: I_t, and number of epochs: E.

model $\leftarrow$ Instantiate(KAN);

for $e \in \{0,\ 1,\ \ldots,\ E-1\}$ **do**

  for $(x,\ y) \in (X_{train},\ Y_{train})$ **do**

    $x_{\text{adv}} \leftarrow \text{PGD}(x,\ \epsilon,\ \alpha,\ I_t)$;

    $x_{\text{mixed}} \leftarrow \text{Concatenate}(x,\ x_{\text{adv}})$;

    $y_{\text{mixed}} \leftarrow \text{Concatenate}(y, y)$;

    $y_p \leftarrow \text{model.forward}(x_{\text{mixed}})$;

    $\text{loss} \leftarrow \ell(y_p,\ y_{\text{mixed}})$;

    model.backward(loss);

return model.

4.2. Defense Mechanisms

In this section, we explain the defense mechanisms that are used during the experiments. These defenses include adversarial training, randomized smoothing, and interval bound propagation.

Algorithm 1 shows the adversarial training strategy that we employ during the training session. We use PGD to generate adversarial images within each batch with 8/255 as perturbation bound, 2/255 as step size, and 10 as the number of iterations. Then, we restrict each pixel between 0 and 1. After that, we train the model on the mixed batch of benign and adversarial data in order to make the model learn both benign and adversarial distributions.

Algorithm 2 demonstrates how we employ randomized smoothing during the training session and the test phase. During the training session, we sample from a Gaussian noise distribution with noise level of 0.25. After that, we obtain noisy images by adding the noise to the original images and train the base classifier on the noisy images. During the inference phase, we generate a smoothed classifier with 0.25 as noise level and 64 as the number of Monte Carlo samples. Adversarial data (FGSM, PGD, and C&W) are generated based on the gradients of the base model; however, the final evaluation is conducted on the smoothed model.

Algorithm 3 demonstrates the IBP-based training. Epochs are divided into three intervals: warm-up, ramp-up, and robust training. In warm-up,

Algorithm 2: Randomized Smoothing (Training and Inference)

Input: Training data: X_{train}, training labels: Y_{train}, test data: X_{test}, test labels: Y_{test}, Gaussian noise level σ, identity matrix: I, number of epochs: E, and Monte Carlo sample number: $\mathcal{M}$.

model $\leftarrow$ Instantiate(KAN);

for $e \in \{0,\ 1,\ \ldots,\ E-1\}$ **do**

  for $(x,\ y) \in (X_{train},\ Y_{train})$ **do**

    $\xi \leftarrow \mathcal{N}(0, \sigma^2 I)$;

    $x_{\text{noisy}} \leftarrow x + \xi$;

    $y_p \leftarrow$ model.forward(x_{noisy});

    loss $\leftarrow \ell(y_p,\ y)$;

    model.backward(loss);

Inference phase:

smoothed = SmoothedClassifier(model, σ, $\mathcal{M}$);

standard_accuracy $\leftarrow$ Evaluate(smoothed, X_{test}, Y_{test});

$X_{\text{Adversarial}} \leftarrow$ Attack(model, X_{test}, Y_{test});

robust_accuracy $\leftarrow$ Evaluate(smoothed, $X_{\text{Adversarial}}$, Y_{test});

perturbation size and trade-off parameter λ are equal to 0 in order to normally train the model. After that, in ramp-up, λ gradually increases to 0.5 and ϵ gradually increases to $4/255$. Finally, in robust training interval the model is trained on the maximum pre-defined values of λ and ϵ. The formula for linear ramp-up function $\mathcal{S}_t$ is as follows:

$$\mathcal{S}_t = \begin{cases} 0 & t \leq W_E \\ \frac{t - W_E}{R_E - W_E} & W_E < t \leq R_E \\ 1 & R_E < t \end{cases} \tag{22}$$

where W_E is warm-up epochs, R_E is ramp-up epochs, and t is the epoch number. Moreover, $\ell_{\text{IBP}}(y_p,\ y,\ \epsilon_t)$ constructs the initial interval $[x - \epsilon_t,\ x + \epsilon_t]$ and propagates these bounds through the network. The final loss during IBP-based training is calculated based on Eq. (5).

4.3. Threat Model

We assume strong white-box evasion attacks with full knowledge about the models' architecture, parameters, gradients, and defense mechanism. The

Algorithm 3: Interval Bound Propagation Training.

Input: Training data: X_{train}, training labels: Y_{train}, maximum trade-off coefficient: $\lambda_{\max}$, number of epochs: E, warm-up epochs: W_E, ramp-up epochs: R_E, maximum perturbation size: $\epsilon_{\max}$, ramp-up function: $\mathcal{S}_t$.

for $e \in \{0,\ 1,\ \ldots,\ E-1\}$ **do**

  $t \leftarrow e + 1$;

  for $(x,\ y) \in (X_{train},\ Y_{train})$ **do**

    $\lambda_t = \mathcal{S}_t \times \lambda_{\max}$;

    $\epsilon_t = \mathcal{S}_t \times \epsilon_{\max}$;

    $y_p \leftarrow$ model.forward(x);

    loss $\leftarrow (1-\lambda_t) \times \ell_{\text{Clean}}(y_p,\ y) + \lambda_t \times \ell_{\text{IBP}}(y_p,\ y,\ \epsilon_t)$;

    model.backward(loss);

attackers cannot modify training data or model parameters and are restricted to test-time perturbations within predefined norm-bounded regions.

Attacks include FGSM, PGD, and C&W. We use ℓ_∞-norm FGSM with perturbation sizes ranging from 0.01 to 0.09. For PGD, we use three different variants: standard, multi-step, and adaptive PGD. All the PGD attacks use perturbation bound of 8/255 and step size of 2/255.

In standard PGD, the iteration ranges from 10 to 50, while other types generate adversarial samples during 100 iterations. In multi-step PGD, we use three restarts, and for adaptive PGD, we use three expectation-over-transformation (EoT) samples. Moreover, we use ℓ_2-norm C&W to further evaluate the robustness of these KAN-based models. Table 1 summarizes the specifications and parameters of each attack.

Table 1: The parameters of each adversarial attack.

Attack	Parameters	Norm
FGSM	$\epsilon \in \{0.01,\ 0.02, ..., 0.09\}$	ℓ_∞
Standard PGD	$\epsilon = 8/255$, $\alpha = 2/255$, $i \in \{10,\ 20,\ 30,\ 40,\ 50\}$	ℓ_∞
Multi-Step PGD	$\epsilon = 8/255$, $\alpha = 2/255$, $i = 100$, Restart $= 3$	ℓ_∞
Adaptive PGD	$\epsilon = 8/255$, $\alpha = 2/255$, $i = 100$, EoT Samples $= 3$	ℓ_∞
C&W	binary-search steps=9, steps $= 1000$, step size $= 1e-2$, confidence $= 0$	ℓ_2

5. Experimental Setups and Results

In this section, we explain the experimental setups, certified results, empirical results, and discuss our findings. The empirical results are averaged over five runs for all the models on both datasets.

5.1. Datasets and Implementation Details

We use CIFAR-10 and SVHN datasets. Both datasets include three input channels: red, green, and blue. As a result, we use vectors of size three to normalize the data, where each element correspond to one of the channels. We normalize CIFAR-10 using mean vector of (0.4914, 0.4822, 0.4465) and standard deviation vector of (0.247, 0.243, 0.261).

Moreover, we normalize SVHN with mean vector of (0.4377, 0.4438, 0.4728) and standard deviation vector of (0.1980, 0.2010, 0.1970). We keep the same normalization values during both training and test sessions.

We used a server equipped with two NVIDIA H100 GPUs. For the main libraries, we used numpy `2.2.4`, torch `2.7.0`, torchvision `0.22.0`, and scipy `1.15.3`.

5.2. Evaluated Architectures and Metrics

We use KAN-Mixers Canuto et al. (2025), KANICE Ferdaus et al. (2024), and PoolKANNeXt Lv et al. (2025) architectures to conduct our experiments. We use these architectures since they have demonstrated state-of-the-art performances in CV-based tasks; this enables us to evaluate the accuracy and robustness of these models more fairly across the datasets.

In order to assess the performance of the selected models, we use standard accuracy (SA) and robust accuracy (RA). SA shows the accuracy of the model under benign dataset, while RA measures the accuracy under adversarial samples. By using these two metrics, we can analyze the performance of the models more deeply, and gain valuable insights of the behavior of these models.

5.3. Hyperparameters

KANICE is trained for 200 epochs, with batch size of 128, and learning rate of $1e-3$. KAN-Mixers is trained for 50 epochs, with batch size of 64, and learning rate of 0.0001282. PoolKANNeXt is trained for 30 epochs, with batch size of 128, and learning rate of 0.0037. These settings stay the same during standard, adversarial, RS-based, and IBP-based training sessions. For

adversarial training, we use perturbation bound 8/255, step size 2/255, and 10 iterations across all the architectures.

For RS, we set Gaussian noise level (σ) to 0.25. Moreover, the number of Monte Carlo samples ($\mathcal{M}$) is a test-time parameter, which equals to 64 for all models across all datasets. During IBP-based training sessions, maximum perturbation size is 4/255 and maximum λ equals to 0.5 across all the models. The number of warm-up and ramp-up epochs are 10 and 80 for KANICE, 5 and 40 for KAN-Mixers, and 5 and 20 for PoolKANNeXt. Table 2 summarizes the hyperparameters that are used during the training session.

Table 2: Training hyperparameters.

Model	Common			AT			RS		IBP			
	epochs	bs	lr	ϵ	α	Iterations	σ	$\mathcal{M}$	$\epsilon_{\max}$	W_E	R_E	$\lambda_{\max}$
KANICE	200	128	$1e-3$	8/255	2/255	10	0.25	64	4/255	10	80	0.5
KAN-Mixers	50	64	0.0001282	8/255	2/255	10	0.25	64	4/255	5	40	0.5
PoolKANNeXt	30	128	0.0037	8/255	2/255	10	0.25	64	4/255	5	20	0.5

5.4. Certified Robustness Results

In this section, we explain the certified robustness results of randomized smoothing.

Table 3: ℓ_2-certified robustness of different KAN architectures under randomized smoothing at certification thresholds of 2/255, 4/255, and 8/255 - certified accuracy (CA) and mean radius.

Model	Dataset	CA@$\frac{2}{255}$	CA@$\frac{4}{255}$	CA@$\frac{8}{255}$	MR
KANICE	CIFAR-10	60.85	60.19	59.01	0.46
KAN-Mixers	CIFAR-10	66.84	66.27	65.46	0.75
PoolKANNeXt	CIFAR-10	80.33	80.10	79.78	1.18
KANICE	SVHN	92.70	92.45	91.93	0.91
KAN-Mixers	SVHN	91.38	90.92	90.03	0.71
PoolKANNeXt	SVHN	92.36	92.24	92.05	1.10

Table 3 shows the ℓ_2-certified robustness of different KAN architectures under randomized smoothing. Certified accuracy (CA) measures the percentage of test samples satisfying $R_i \geq r$, where R_i denotes the certified ℓ_2 radius and r is the certification threshold. The mean radius (MR) represents the average certified radius across all test samples, where larger values indicate

stronger certified robustness guarantees. In CIFAR-10, PoolKANNeXt demonstrates the strongest certified robustness, while in SVHN, PoolKANNeXt and KANICE demonstrate competitive certified robustness.

5.5. Empirical Robustness Results

Table 4 shows the SA and RA under FGSM attacks for all evaluated architectures and defense strategies on CIFAR-10 and SVHN. Across both datasets, the results reveal a robustness hierarchy, in which adversarial training consistently provides the strongest protection against gradient-based attacks.

Table 4: Standard accuracy and robust accuracy of different KAN models under FGSM attacks.

Model \ Metric	SA	RA - FGSM								
		0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
CIFAR-10										
KAN-Mixers-Plain	76.17	32.03	28.62	25.64	23.12	21.16	19.34	17.58	16.28	14.92
KANICE-Plain	85.09	40.74	38.35	36.42	34.87	33.37	32.05	30.60	29.36	28.18
PoolKANNeXt-Plain	85.82	36.41	25.73	18.96	14.58	12.12	10.31	9.08	8.03	7.54
KAN-Mixers-AT	74.42	60.09	57.53	54.85	52.25	49.46	47.04	44.54	42.05	39.90
KANICE-AT	84.55	65.74	64.45	63.28	**61.89**	**60.48**	**59.09**	**58.00**	**56.96**	**56.22**
PoolKANNeXt-AT	85.27	**70.21**	**67.30**	**64.04**	60.79	57.75	54.51	51.04	48.23	45.56
KAN-Mixers-RS	73.77	34.21	32.81	31.06	29.50	28.27	26.66	25.23	24.31	22.93
KANICE-RS	83.11	46.38	44.92	43.53	42.08	40.65	39.45	37.97	36.66	35.41
PoolKANNeXt-RS	**86.37**	41.90	35.62	30.18	25.54	21.74	18.50	16.40	14.70	13.09
KAN-Mixers-IBP	75.88	32.31	29.37	26.96	24.94	23.11	21.76	20.20	19.05	18.05
KANICE-IBP	**86.65**	44.35	42.19	40.24	38.44	36.63	35.36	34.06	33.01	31.85
PoolKANNeXt-IBP	77.05	35.28	31.79	28.54	25.14	22.70	20.50	17.99	16.20	14.75
SVHN										
KAN-Mixers-Plain	94.51	62.18	54.54	48.19	42.74	38.31	34.55	31.63	29.03	26.83
KANICE-Plain	93.39	65.34	63.88	62.35	61.06	59.78	58.74	57.73	56.78	55.94
PoolKANNeXt-Plain	92.62	49.71	37.06	28.78	23.26	19.34	16.55	14.50	13.04	11.84
KAN-Mixers-AT	**95.74**	**78.75**	**77.69**	**76.49**	**75.09**	**73.69**	**72.26**	**70.82**	**69.35**	**67.76**
KANICE-AT	93.90	73.76	72.36	71.07	69.63	68.29	66.93	65.65	64.44	63.26
PoolKANNeXt-AT	92.75	73.24	71.48	69.45	67.39	65.17	62.85	60.48	58.09	55.76
KAN-Mixers-RS	93.78	68.02	65.87	63.29	60.53	57.78	54.58	52.03	49.10	46.64
KANICE-RS	91.66	63.51	62.43	61.15	59.59	58.10	56.54	55.02	53.57	52.03
PoolKANNeXt-RS	92.57	63.54	59.35	54.36	49.90	45.62	41.82	38.52	35.58	33.03
KAN-Mixers-IBP	89.07	60.42	55.51	50.79	46.33	42.58	39.21	36.24	33.77	31.45
KANICE-IBP	92.34	62.07	60.11	58.05	56.14	54.34	52.76	51.39	50.34	49.19
PoolKANNeXt-IBP	90.58	56.65	49.06	42.39	36.76	32.19	28.34	25.05	22.38	20.19

On CIFAR-10, clean accuracy varies substantially across architectures. The highest SA is achieved by KANICE-IBP (86.65%) and PoolKANNeXt-RS (86.37%), demonstrating that certified defenses do not necessarily degrade benign performance. However, robustness under FGSM tells a different

story. As the perturbation size increases from 0.01 to 0.09, all models experience performance degradation. Moreover, adversarially trained models exhibit significantly slower degradation rates. For example, PoolKANNeXt-AT maintains 45.56% RA at $\epsilon = 0.09$, whereas its plain counterpart drops significantly to 7.54%. Similarly, KANICE-AT retains 56.22% RA at the strongest perturbation level, substantially exceeding the robustness achieved by KANICE-RS (35.41%) and KANICE-IBP (31.85%). These observations suggest that exposure to adversarial examples during the training session enables the models to learn decision boundaries that are substantially more resistant to first-order gradient attacks.

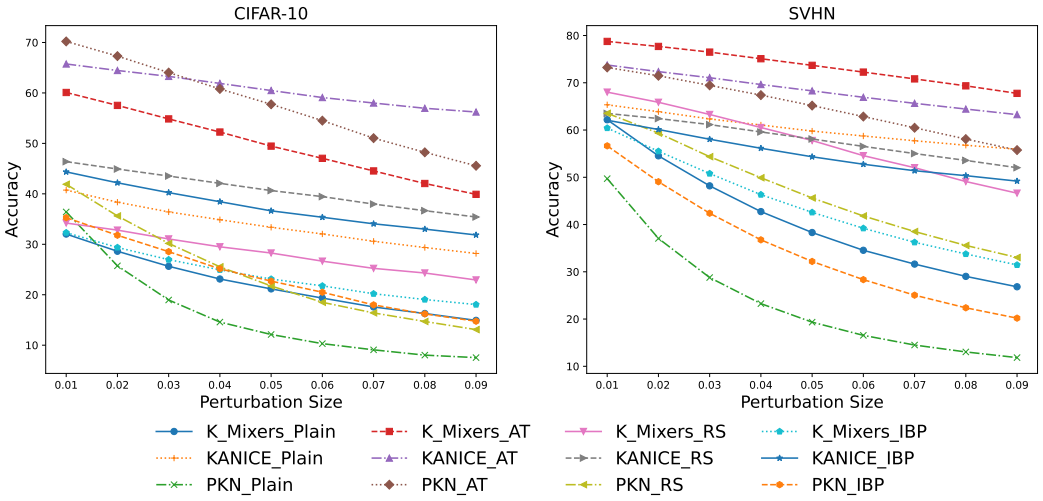

Figure 3: Robust accuracy of KAN models against FGSM across different perturbation sizes on the CIFAR-10 and SVHN datasets.

A similar trend is observed on SVHN, although the overall robustness levels are noticeably higher. Nearly all models achieve more than 90% SA, indicating that SVHN is intrinsically easier for the evaluated architectures than CIFAR-10. Furthermore, adversarially trained variants not only preserve robustness but also achieve some of the highest clean accuracies. KAN-Mixers-AT attains 95.74% SA while maintaining 67.76% RA at $\epsilon = 0.09$, outperforming all other defense mechanisms. Figure 3 demonstrate the RA of the models under FGSM with different perturbation sizes. RA consistently decreases as the perturbation size increases, suggesting the more profound effects of the stronger attack configurations. Compared with CIFAR-10, SVHN generally exhibits a steeper decline in RA as perturbation size increases, although its

absolute robustness remains consistently higher across perturbation sizes.

Another notable observation is the limited effectiveness of randomized smoothing and interval bound propagation against FGSM attacks. While both defenses generally improve robustness compared to the undefended models, their gains remain modest relative to adversarial training. This outcome is expected because RS and IBP are designed primarily to provide certified guarantees within predefined perturbation regions rather than explicitly optimizing robustness against the strongest empirical adversarial examples. As a result, their empirical robustness remains lower than that of adversarially trained models despite occasionally achieving higher clean accuracies.

Table 5 reports the robust accuracy under multiple PGD variants and the ℓ_2-based C&W attack. Compared with FGSM, PGD represents a substantially stronger adversary because it performs iterative optimization to find more effective perturbations. As a result, the robustness gap between defense mechanisms becomes more obvious.

For CIFAR-10, adversarial training again emerges as the most effective defense. PoolKANNeXt-AT achieves the highest robustness under all PGD configurations, maintaining approximately 54.6% RA regardless of attack iterations or attack variants. KAN-Mixers-AT and KANICE-AT follow with robust accuracies of approximately 41% and 38%, respectively. In contrast, the corresponding undefended models achieve less than 8% RA under PGD, highlighting the severe vulnerability of standard KAN architectures to iterative gradient-based attacks. Interestingly, robustness remains relatively stable as the number of PGD iterations increases from 10 to 50 and even under multi-step and adaptive PGD variants, indicating that the attacks have largely converged and that the reported robustness values represent meaningful lower-bound estimates.

The performance of randomized smoothing under PGD is mixed. Although RS substantially improves robustness compared with the plain models, it remains consistently inferior to adversarial training. For example, KANICE-RS achieves approximately 35% RA under PGD, while KANICE-AT maintains approximately 38%. Similarly, IBP provides only moderate improvements over the baseline models. These findings suggest that certified defenses offer limited empirical protection against strong optimization-based attacks despite providing formal robustness guarantees within specific perturbation bounds.

The SVHN results exhibit the same overall ranking but with considerably higher robustness values. KAN-Mixers-AT achieves the strongest PGD robustness at approximately 69%, followed by PoolKANNeXt-AT. Randomized

Table 5: Robust accuracy of different KAN models under PGD and C&W attacks including multi-step (MS) and adaptive (A) PGD.

Model \ Metric	PGD							C&W
	10	20	30	40	50	MS	A	
CIFAR-10								
KAN-Mixers-Plain	7.71	7.61	7.60	7.60	7.60	7.59	7.60	24.51
KANICE-Plain	7.10	6.42	6.39	6.36	6.31	6.08	6.26	27.82
PoolKANNeXt-Plain	2.56	2.39	2.37	2.37	2.36	2.29	2.34	26.91
KAN-Mixers-AT	41.52	41.42	41.39	41.38	41.38	41.37	41.38	25.48
KANICE-AT	38.66	38.10	38.04	38.01	37.99	37.32	38.00	20.71
PoolKANNeXt-AT	**54.77**	**54.65**	**54.62**	**54.61**	**54.61**	**54.59**	**54.60**	**31.17**
KAN-Mixers-RS	25.20	25.18	25.38	25.37	25.41	25.42	25.47	25.52
KANICE-RS	34.59	34.46	34.65	34.88	34.83	34.89	34.65	20.58
PoolKANNeXt-RS	11.80	11.55	11.50	11.51	11.52	11.43	11.38	23.51
KAN-Mixers-IBP	9.19	9.08	9.06	9.05	9.05	8.99	9.05	22.42
KANICE-IBP	12.44	11.88	11.82	11.81	11.81	11.62	11.80	26.85
PoolKANNeXt-IBP	16.57	16.34	16.29	16.29	16.29	16.29	16.30	22.13
SVHN								
KAN-Mixers-Plain	25.81	25.04	24.95	24.90	24.89	24.75	24.87	72.09
KANICE-Plain	39.34	37.98	37.90	37.89	37.88	37.59	37.86	66.15
PoolKANNeXt-Plain	12.38	11.99	11.90	11.88	11.86	11.85	11.83	60.71
KAN-Mixers-AT	**69.10**	**68.92**	**68.90**	**68.90**	**68.90**	**68.85**	**68.90**	61.75
KANICE-AT	51.66	51.08	51.03	51.01	51.00	50.81	50.98	61.42
PoolKANNeXt-AT	63.15	63.04	63.03	63.03	63.03	63.02	63.03	50.42
KAN-Mixers-RS	56.28	56.25	56.21	56.10	56.22	56.09	56.18	**72.58**
KANICE-RS	51.41	51.40	51.37	51.29	51.46	51.32	51.46	51.30
PoolKANNeXt-RS	43.63	43.44	43.39	43.32	43.30	43.31	43.33	67.47
KAN-Mixers-IBP	33.50	33.05	32.99	32.96	32.96	32.90	32.95	62.93
KANICE-IBP	32.57	31.28	31.23	31.21	31.20	31.02	31.18	60.61
PoolKANNeXt-IBP	29.77	29.29	29.21	29.17	29.18	29.11	29.17	44.03

smoothing also performs notably better on SVHN than on CIFAR-10, with KAN-Mixers-RS maintaining approximately 56% robustness across all PGD variants. This behavior further supports the observation that SVHN is inherently more resilient to adversarial perturbations and allows certified defenses to operate more effectively.

The C&W results reveal a different trend. Unlike FGSM and PGD, where adversarial training dominates, randomized smoothing frequently provides competitive robustness under the optimization-based ℓ_2 attack. On CIFAR-10, PoolKANNeXt-AT achieves the highest robustness (31.17%), but several RS and plain-model configurations achieve comparable performance. On SVHN, KAN-Mixers-RS attains the highest robustness (72.58%), slightly exceeding both the plain and adversarially trained variants. This indicates that the

smoothing mechanism can effectively mitigate perturbations generated in the ℓ_2 space, making RS particularly suitable against C&W-style attacks.

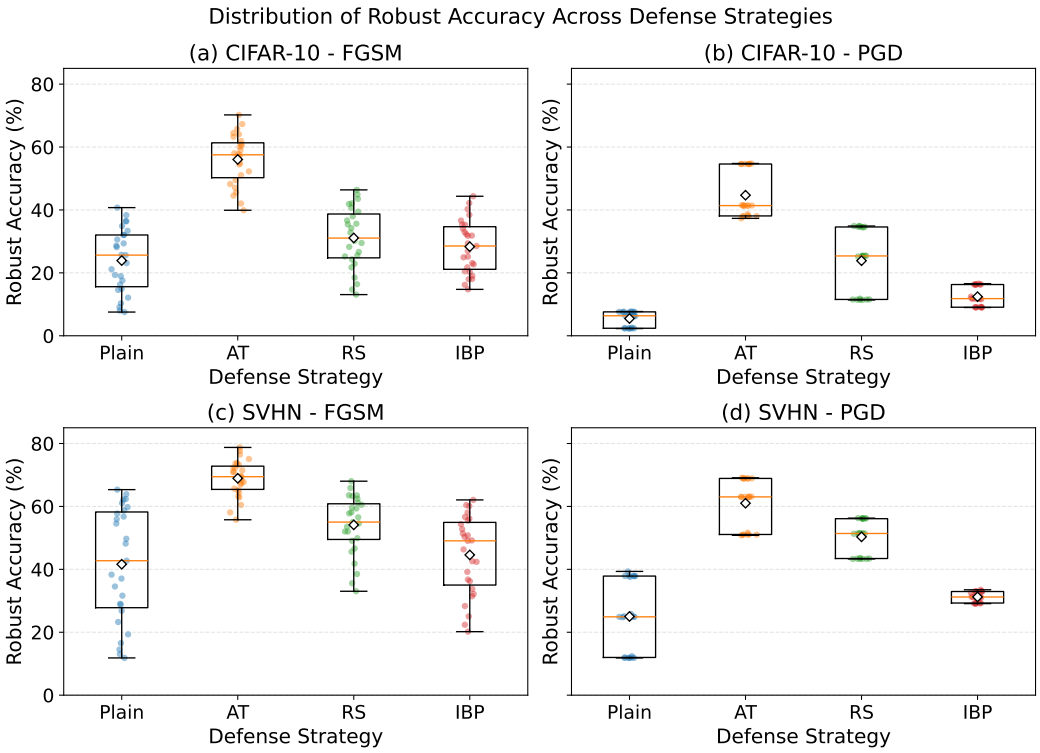

Figure 4: Box plots of robust accuracy across defense strategies under FGSM and PGD attacks on CIFAR-10 and SVHN.

Figure 4 provides a distributional comparison of RA across the evaluated defense strategies by aggregating the results of all selected KAN architectures over different attack configurations. For FGSM, each box represents the RA values obtained across nine perturbation sizes, while for PGD, each box summarizes the results across the seven evaluated PGD configurations. The figure reveals a clear separation between AT and the other defense strategies, with AT demonstrating the RA distributions upward on both CIFAR-10 and SVHN. This confirms that AT provides the strongest overall empirical protection against FGSM and PGD attacks. The advantage of AT is particularly evident under PGD, where the plain models exhibit very low robust accuracy on CIFAR-10, while the adversarially trained variants maintain substantially higher values. RS generally provides the second strongest protection, partic-

ularly on SVHN, while IBP produces more moderate empirical robustness despite its certified-training objective. The distributions also indicate that SVHN models generally maintain higher robust accuracy than their CIFAR-10 counterparts across defense strategies.

Furthermore, the spread of the box plots suggests that robustness remains architecture-dependent. Although a defense may improve the overall robustness distribution, its effectiveness varies among KAN-Mixers, KAN-ICE, and PoolKANNeXt. These findings suggest that architectural choices, including interactive convolutional blocks, pooling-based feature aggregation, and mixer-style token processing, influence both empirical and certified robustness in different ways, despite all models being built upon the same Kolmogorov-Arnold learning framework.

6. Conclusion

In this paper, we evaluated the empirical robustness of different defended and undefended KAN architectures against FGSM, PGD, and C&W attacks. While adversarial training proved to be the most effective defense strategy against gradient-based attacks, randomized smoothing demonstrated a competitive performance against ℓ_2-norm C&W attack. Moreover, the choices of architecture can directly affect both certified and empirical robustness.

Our future research will focus more deeply on certified robustness of KAN models by employing more certification techniques including verification and convex relaxation. Moreover, investigating more diverse architectures of KAN models in terms of certified and empirical robustness is another future direction. By investigating these areas and other diverse set of defenses, the limitations and strengths of these models could be further investigated.

References

Alter, T., Lapid, R., Sipper, M., 2024. On the robustness of kolmogorov-arnold networks: An adversarial perspective. arXiv preprint arXiv:2408.13809 .

Boutaba, R., Salahuddin, M.A., Limam, N., Ayoubi, S., Shahriar, N., Estrada-Solano, F., Caicedo, O.M., 2018. A comprehensive survey on machine learning for networking: evolution, applications and research opportunities. Journal of Internet Services and Applications 9, 1–99.

Cacciatore, A., Morelli, V., Paganica, F., Frontoni, E., Migliorelli, L., Berardini, D., 2024. A preliminary study on continual learning in computer vision using kolmogorov-arnold networks. arXiv preprint arXiv:2409.13550 .

Canuto, J.L.d.S., Aylon, L.B.R., de Souza, R.C.T., 2025. Kan-mixers: a new deep learning architecture for image classification. arXiv preprint arXiv:2503.08939 .

Carlini, N., Wagner, D., 2017. Towards evaluating the robustness of neural networks, in: 2017 ieee symposium on SP, Ieee. pp. 39–57.

Cohen, J., Rosenfeld, E., Kolter, Z., 2019. Certified adversarial robustness via randomized smoothing, in: international conference on machine learning, PMLR. pp. 1310–1320.

Djosic, N., Ostanin, E., Hussain, F., Sharieh, S., Ferworn, A., 2024. Kan vs kan: Examining kolmogorov-arnold networks (kan) performance under adversarial attacks. Proceedings of the SECURWARE , 17–22.

Dong, C., Zheng, L., Chen, W., 2024. Kolmogorov-arnold networks (kan) for time series classification and robust analysis, in: International Conference on Advanced Data Mining and Applications, Springer. pp. 342–355.

Ferdaus, M.M., Abdelguerfi, M., Ioup, E., Dobson, D., Niles, K.N., Pathak, K., Sloan, S., 2024. Kanice: Kolmogorov-arnold networks with interactive convolutional elements, in: Proceedings of the 4th International Conference on AI-ML Systems, pp. 1–10.

Goodfellow, I.J., Shlens, J., Szegedy, C., 2015. Explaining and harnessing adversarial examples. stat 1050, 20.

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

Ibrahum, A.D.M., Shang, Z., Hong, J.E., 2024. How resilient are kolmogorov–arnold networks in classification tasks? a robustness investigation. Applied Sciences 14, 10173.

Ji, T., Hou, Y., Zhang, D., 2024. A comprehensive survey on kolmogorov arnold networks (kan). arXiv e-prints , arXiv–2407.

Kurakin, A., Goodfellow, I.J., Bengio, S., 2017. Adversarial machine learning at scale, in: International Conference on Learning Representations.

Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S., 2022. A convnet for the 2020s, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11976–11986.

Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljacic, M., Hou, T., Tegmark, M., 2025. Kan: Kolmogorov–arnold networks, in: International conference on learning representations, pp. 70367–70413.

Lou, S., Shao, Y., Du, Q., 2026. Kolmogorov-arnold optimized unet: An enhanced image segmentation model based on kolmogorov-arnold network and convolutional kolmogorov-arnold network. Engineering Applications of Artificial Intelligence 173, 114405.

Lv, F., Wei, Q., Huang, Y., Tuncer, T., Dogan, S., Özyurt, F., 2025. Poolkan-next: A new pooling-based kolmogorov arnold convolutional neural network. Alexandria Engineering Journal 128, 144–152.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A., 2018. Towards deep learning models resistant to adversarial attacks, in: International Conference on Learning Representations.

Mao, Y., Müller, M.N., Fischer, M., Vechev, M., 2024. Understanding certified training with interval bound propagation, in: International Conference on Learning Representations, pp. 13470–13492.

Meng, M.H., Bai, G., Teo, S.G., Hou, Z., Xiao, Y., Lin, Y., Dong, J.S., 2022. Adversarial robustness of deep neural networks: A survey from a formal verification perspective. IEEE Transactions on Dependable and Secure Computing .

Meymani, M., Razavi-Far, R., 2026. Divided we fall: Defending against adversarial attacks via soft-gated fractional mixture-of-experts with randomized adversarial training. Information Sciences , 123427.

Meymani, M., Razavi-Far, R., Vashagh, A., Biggio, B., 2026. Defense against adversarial attacks: Foundations, strategies, and future directions .

Mirman, M., Gehr, T., Vechev, M., 2018. Differentiable abstract interpretation for provably robust neural networks, in: International Conference on Machine Learning, PMLR. pp. 3578–3586.

Nandi, S., Addepalli, S., Rangwani, H., Babu, R.V., 2023. Certified adversarial robustness within multiple perturbation bounds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2298–2305.

Nath, V., Chattopadhyay, C., Desai, K., 2023. On enhancing prediction abilities of vision-based metallic surface defect classification through adversarial training. Engineering Applications of Artificial Intelligence 117, 105553.

Ostanin, E., Djosic, N., Hussain, F., Sharieh, S., Ferworn, A., 2024. Evaluating the robustness of kolmogorov-arnold networks against noise and adversarial attacks. Proceedings of the SECURWARE , 11–16.

Raghunathan, A., Steinhardt, J., Liang, P.S., 2018. Semidefinite relaxations for certifying robustness to adversarial examples. Advances in neural information processing systems 31.

Rahman, M.M., Wang, G., Zhou, K., Chen, M., Yang, F., 2026. Catastrophic forgetting in kolmogorov-arnold networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 25082–25089.

Salman, H., Yang, G., Zhang, H., Hsieh, C.J., Zhang, P., 2019. A convex relaxation barrier to tight robustness verification of neural networks. Advances in Neural Information Processing Systems 32.

Sarker, I.H., 2021. Machine learning: Algorithms, real-world applications and research directions. SN computer science 2, 1–21.

Schumacher, M.L., Heiderich, B., Huber, M.F., 2025. Training and verifying robust kolmogorov-arnold networks, in: 2025 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), IEEE. pp. 1–3.

Shinde, P.P., Shah, S., 2018. A review of machine learning and deep learning applications, in: 2018 Fourth international conference on computing communication control and automation (ICCUBEA), IEEE. pp. 1–6.

Somvanshi, S., Javed, S.A., Islam, M.M., Pandit, D., Das, S., 2025. A survey on kolmogorov-arnold network. ACM Computing Surveys 58, 1–35.

Tolstikhin, I.O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al., 2021. Mlp-mixer: An all-mlp architecture for vision. Advances in neural information processing systems 34, 24261–24272.

Vashagh, A., Razavi-Far, R., Meymani, M., Biggio, B., 2026. Recent advances in adversarial attacks on model utility, privacy, and explainability: A comprehensive survey .

Xu, H., Liu, X., Li, Y., Jain, A., Tang, J., 2021. To be robust or to be fair: Towards fairness in adversarial training, in: International conference on machine learning, PMLR. pp. 11492–11501.

Zhang, R., 2020. On the tightness of semidefinite relaxations for certifying robustness to adversarial examples. Advances in Neural Information Processing Systems 33, 3808–3820.