ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240745

Classifier Guidance Enhances Diffusion-Based Adversarial Purification by Preserving Predictive Information

Mingkun Zhang [a, c], **Jianing Li** [a], **Wei Chen** [a, c, *], **Jiafeng Guo**[b, c] **and Xueqi Cheng**[a, c]

[a]CAS Key Laboratory of AI Safety
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b]Key Laboratory of Network Data Science and Technology
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[c]University of Chinese Academy of Sciences, Beijing, China

Abstract. Adversarial purification is one of the promising approaches to defend neural networks against adversarial attacks. Recently, methods utilizing diffusion probabilistic models have achieved great success for adversarial purification in image classification tasks. However, such methods fall into the dilemma of balancing the needs for noise removal and information preservation. This paper points out that existing adversarial purification methods based on diffusion models gradually lose sample information during the core denoising process, causing occasional label shift in subsequent classification tasks. As a remedy, we suggest to suppress such information loss by introducing guidance from the classifier confidence. Specifically, we propose Classifier-cOnfidence gUided Purification (COUP) algorithm, which purifies adversarial examples while keeping away from the classifier decision boundary. Experimental results show that COUP can achieve better adversarial robustness under strong attack methods.

1 Introduction

[1]

Extensive research has shown that neural networks are vulnerable to well-designed adversarial examples, which are created by adding imperceptible perturbations on benign samples [12, 27, 3, 6]. Various approaches have been explored to improve model robustness, including model training enhancement [27, 47, 14] and input data preprocessing [32, 25]. While these works have significantly improved adversarial robustness, there is still a clear gap in the classification accuracy between clean and adversarial data.

In recent years, adversarial purification with diffusion probabilistic models [28, 44, 4, 43, 39] has become an effective approach to defend against adversarial attacks in image classification tasks. The key idea is to preprocess the input image using an auxiliary diffusion model before feeding it into the downstream classifier. Leveraging the strong ability of generative models to fit data distributions, adversarial purification methods are able to purify the adversarial examples by pushing them toward the manifold of benign data. Such a process is essentially a denoising process, which gradually removes possible noise from input data.

Though achieved advanced performance on robust image classification tasks, adversarial purification methods rely solely on the denoising function during purification, thus inevitably fall into the dilemma of balancing the need for noise removal and information preservation [28]. While stronger purification may destroy the image details that are necessary for classification, weaker purification may not be sufficient to remove the adversarial perturbations completely. The passive strategy to balance this, i.e. controlling the global purification steps [28], is limited in its effect, in the sense that data information monotonically loses as the purification steps grow. The existing method to mitigate the loss of information is to constrain the distance between the input adversarial example and the purified image [39, 43]. However, such constraint may inhibit the purified example from escaping the adversarial region effectively.

In this paper, we aim to propose a method that directly takes into consideration the need for information preservation. We borrow the idea of classifier guidance for diffusion models [37, 17, 8, 20], using the classifier confidence on the current class label y given data x as an indicator of the degree of preservation and try to maintain high confidence during the purification process. Staying away from low-confidence areas is beneficial to successful purification since such areas are close to the decision boundary and are more sensitive to small perturbations. Approaching a low confidence area can result in a potential label shift problem, i.e. a sample that initially has the correct label is misclassified after purification, especially when combined with stochastic defense strategies.

Specifically, we propose a Classifier-cOnfidence gUided Purification algorithm (COUP) with diffusion models to match the requirement of information preservation. The key idea is to gradually push input data towards high probability density regions while keeping relatively high confidence for classification. This process is realized by applying the denoising process together with a regularization term which improves the confidence score of the downstream classifier. This guidance discourages the purification process from moving toward decision boundaries, where the classifier becomes confused, and the confidence decreases.

We empirically evaluate our algorithm using strong adversarial at-

* Corresponding Author. Email: chenwei2022@ict.ac.cn
[1] The appendix is available at https://arxiv.org/pdf/2408.05900

tack methods, including AutoAttack [6], which contains both white-box and black-box attacks, Backward Pass Differentiable Approximation (BPDA) [1], as well as EOT [1] to tackle the randomness in defense strategy. Results show that COUP outperforms purification method without classifier guidance, e.g., DiffPure [28] in terms of robustness on CIFAR-10 and CIFAR-100 datasets.

Our work has the following main contributions:

- We propose a new adversarial purification algorithm COUP. By leveraging the confidence score from the downstream classifier, COUP is able to preserve the predictive information while removing the malicious perturbation.
- We provide both theoretical and empirical analysis for the effect of confidence guidance, showing that keeping away from the decision boundary can preserve predictive information and alleviate the label shift problem which are beneficial for classification. Though classifier guidance has been proven to be useful for better generation quality in previous works [17], we are the first to demonstrate its necessity for adversarial purification to the best of our knowledge.
- Experiments demonstrate that COUP can achieve significantly higher adversarial robustness against strong attack methods, reaching a robustness score of 73.05% for l_∞ and 83.13% for l_2 under AutoAttack method on CIFAR-10 dataset.

2 Related Work

Adversarial Training Adversarial training consolidates the discriminative model by enriching trained data [27]. Such methods include generating adversarial examples during model training [27, 47, 29, 19], or using an auxiliary generation model for data augmentation [14, 30, 40]. Though effective, such methods still face adversarial vulnerability for unseen threats [23] and suffer from computational complexity [41] during training. These works are orthogonal to ours and can be combined with our purification method.

Adversarial Purification Adversarial purification is another effective approach to defense. The idea is to use a generative model to purify the adversarial examples before feeding them into the discriminative model for classification. Based on different generative models [11, 21, 24, 18, 37], corresponding purification methods are proposed [32, 25, 10, 15, 16, 4] to convert the perturbed sample into a benign one. Recently, adversarial purification methods based on diffusion models have been proposed and achieved better performance [46, 44, 28, 43, 39]. Among these works, DiffPure [28] achieves the most remarkable result, which is the focus of our comparison.

Classifier Guided Diffusion Models Diffusion models [34, 18, 35, 37] are recently proposed generation models, achieving high generation quality on images. Some works further leverage the guidance of the classifier to achieve controllable generation and improve the image synthesis ability [17, 37, 8, 20]. Although the idea of classifier guidance has been proven to be beneficial for better image generation quality, whether the guided diffusion is helpful for adversarial purification is not yet verified. In our work, we utilize the classifier guidance to mitigate the loss of predictive information, so as to strike a balance between information preservation and purification.

3 Objective of Adversarial Purification

In this section, we present an objective of adversarial purification from the perspective of classification tasks and discuss how to achieve such an objective. The analysis results indicate the importance of considering the need for information preservation directly, which can be achieved by introducing guidance from classifier confidence during the purification process.

The concept of adversarial examples is first proposed by Szegedy et al. [38], showing that neural networks are vulnerable to imperceptible perturbations. Data $\boldsymbol{x}_{\text{adv}}$ is called an adversarial example w.r.t. $\boldsymbol{x} \in \mathbb{R}^d$ if it is close enough and belongs to the same class y_{true} under ground truth classifier $p(y|\cdot)$, but has a different label under model $\hat{p}(y|\cdot)$, such that

$$\arg\max_y \hat{p}(y|\boldsymbol{x}_{\text{adv}}) \neq \arg\max_y \hat{p}(y|\boldsymbol{x}) = y_{\text{true}}, \tag{1}$$

with the constraint that $\|\boldsymbol{x}_{\text{adv}} - \boldsymbol{x}\| \leq \epsilon$.

The idea of adversarial purification is to introduce a purification process before feeding data into the classifier. Though the optimal purification result would be converting $\boldsymbol{x}_{\text{adv}}$ back to $\boldsymbol{x}$, it is almost impossible and not necessary. For the task of classification, it is sufficient as long as $\boldsymbol{x}_{\text{adv}}$ shares the same label with $\boldsymbol{x}$. Therefore, the objective of adversarial purification from the classification perspective can be formulated as

$$\max_r \ \hat{p}(y_{\text{true}}|r(\boldsymbol{x}_{\text{adv}})), \tag{2}$$

where $r(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$ is the purification function.

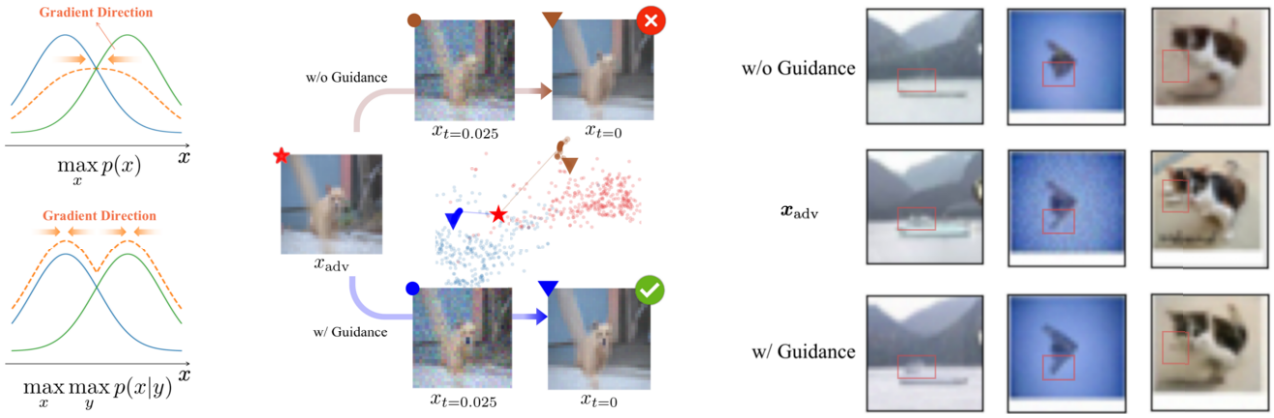
In practice, the above objective cannot be optimized directly since the ground truth label y_{true} is in general unknown, and so is the clean data $\boldsymbol{x}$. A substitute idea may be using the label of a nearby but not too close data $\boldsymbol{x}'$ with high likelihood $p(\boldsymbol{x}')$ instead. The reasons include two aspects: first, the classification model is more trustworthy in high-density areas, thus can eliminate the effect of adversarial perturbation; second, compared with far-away data, a data with moderate distance from $\boldsymbol{x}_{\text{adv}}$ is more likely to share the same label with the clean data $\boldsymbol{x}$. The search for such nearby data is non-trivial, therefore as an alternative, we can use $r(\boldsymbol{x}_{\text{adv}})$ itself as the nearby data and take $\arg\max_y \hat{p}(y|r(\boldsymbol{x}_{\text{adv}}))$ as an approximation of y_{true}.

According to this idea, the ideal $r(\boldsymbol{x}_{\text{adv}})$ for solving Eq.(2) should balance the following requirements:

1. Maximizing the likelihood $p(r(\boldsymbol{x}_{\text{adv}}))$. This helps to remove the adversarial noise.
2. Maximizing the classifier confidence $\max_y \hat{p}(y|r(\boldsymbol{x}_{\text{adv}}))$. This helps to preserve the essential information for classification.
3. Controlling the distance $\|r(\boldsymbol{x}_{\text{adv}}) - \boldsymbol{x}_{\text{adv}}\|$. This helps to avoid significant semantic changes.

Discussion of existing works. Existing adversarial purification methods usually utilize a generative model $\hat{p}(\boldsymbol{x})$ to approximate $p(\boldsymbol{x})$, and try to maximize $\hat{p}(r(\boldsymbol{x}_{\text{adv}}))$ while keeping the distance $\|r(\boldsymbol{x}_{\text{adv}}) - \boldsymbol{x}_{\text{adv}}\|$ small enough. We show a few examples here. DefenseGAN [32] is an early work for such purification, it uses generative adversarial nets as a generative model and optimizes $\min_{\mathbf{z}} \|G(\mathbf{z}) - \boldsymbol{x}_{\text{adv}}\|_2$ for purification. The l_2 norm is used for controlling the distance, and $r(\boldsymbol{x}_{\text{adv}}) = G(\mathbf{z})$ is guaranteed to have a high likelihood with the generator $G(\cdot)$. DiffPure [28] is a recently proposed adversarial purification method, it uses diffusion probabilistic models as a generative model. The purification process is a stochastic differential equation, whose main part includes a score function update which essentially increases the likelihood of $r(\boldsymbol{x}_{\text{adv}})$. Meanwhile, it has been shown that the l_2 distance is implicitly controlled by the global update steps.

We find that the requirement of classifier confidence maximization is widely overlooked in existing works. A possible explanation



(a) Objective (b) Purification Path (c) Purified Image

Figure 1: The distinction between the existing purification method and classifier-confidence guided purification is elucidated in terms of (a) purification objective and (b) visualization of the purification path and (c) the resultant purified image. In (a), the blue curve and green curve represent $p(x|y=0)$ and $p(x|y=1)$, respectively. The orange dotted line indicates the optimization objective ($p(x)$ or $\max_y p(x|y)$) and the direction of the gradient is shown accordingly by the orange arrow. This comparison underscores the importance of classifier confidence guidance in directing the purification process toward the category center. The purification approach outlined in (b) demonstrates that classifier guidance effectively preserves essential predictive information, which is crucial for successful classification. Furthermore, the purified images shown in (c) serve as evidence that classifier guidance retains information necessary for enhanced purification quality.

might be that this requirement is partially addressed through density maximization. This is the case when there is little overlap between different classes of data, such that high likelihood generally means high classifier confidence. However, when such overlap exists, i.e. the decision boundary crosses high probability density areas, maximizing likelihood alone can be problematic. Consider the case where areas around the decision boundary have the highest density, the purification process without classifier confidence guidance will drive nearby samples towards the boundary, causing potential label shift especially when combined with stochastic defense strategies. A specific example is shown in Fig. 1. As a result, we suggest directly addressing the need for information preservation by maximizing the classifier confidence simultaneously.

4 Classifier-Confidence Guided Purification

Motivated by the objective of adversarial purification discussed in Section 3, we propose a **C**lassifier-c**O**nfidence g**U**ided **P**urification (COUP) method with score-based diffusion models to achieve the objective of adversarial purification.

4.1 Methodology

In order to meet the three requirements in Section 3, we address each of them separately: we use a score-based diffusion model and apply the denoising process to maximize the likelihood of purified image (i.e. $\max_{\boldsymbol{r}} \hat{p}(\boldsymbol{r}(\boldsymbol{x}_{\text{adv}}))$); we query the classifier during purification and maximize the classifier confidence (i.e. $\max_{\boldsymbol{r}} \max_y \hat{p}(y|\boldsymbol{r}(\boldsymbol{x}_{\text{adv}}))$); we control the distance by choosing appropriate global update steps t^*. We will first introduce the diffusion model used for denoising, and then explain how we utilize the classifier confidence for adversarial purification.

4.1.1 Score-based Diffusion Models

Diffusion probabilistic models are deep generative models that have recently shown remarkable generation ability. Among existing diffusion models, Score SDE is a unified architecture that models the diffusion process by a stochastic differential equation (SDE) and the denoising process by a corresponding reverse-time SDE. Specifically, the forward SDE is formalized as

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}, t)\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}. \tag{3}$$

where t is the time, $\boldsymbol{f}(\boldsymbol{x}, t)$ is the drift function, $g(t)$ is the diffusion coefficient, $\boldsymbol{w}$ is a standard Wiener process (Brownian motion). The effect of forward SDE is to progressively inject Gaussian noise into the input, and eventually transfer the original data to a Gaussian distribution. The corresponding reverse-time SDE is

$$\mathrm{d}\boldsymbol{x} = [\boldsymbol{f}(\boldsymbol{x}, t) - g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})]\mathrm{d}t + g(t)\mathrm{d}\overline{\boldsymbol{w}}, \tag{4}$$

where $\overline{\boldsymbol{w}}$ is a standard reverse-time Wiener process. $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$ is parameterized by a neural model $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$, which is also called the score function. The score function is the core of the reverse process, driving data $\boldsymbol{x}$ towards higher likelihood by improving $\log p_t(\boldsymbol{x})$, where $p_t(\boldsymbol{x})$ can be viewed as an approximation of $p(\boldsymbol{x})$.

4.1.2 Purification with Guidance of Classifier Confidence

The reverse-time SDE has been used for adversarial purification in previous diffusion-based purification methods. Our key idea is to introduce the guidance signal $\max_y \hat{p}(y|\boldsymbol{x})$ into the reverse-time SDE, such that we use $\log \hat{p}(\boldsymbol{x}) + \lambda \cdot \log \max_y \hat{p}(y|\boldsymbol{x})$ to replace $\log \hat{p}(\boldsymbol{x})$ in the score function $\boldsymbol{s}_\theta(\boldsymbol{x}, t) = \nabla_{\boldsymbol{x}} \log \hat{p}(\boldsymbol{x})$. Therefore, the purification update rule becomes

$$\begin{aligned}\mathrm{d}\boldsymbol{x} = {} & g(t)\mathrm{d}\overline{\boldsymbol{w}} \\ & + \{\boldsymbol{f}(\boldsymbol{x}, t) - g(t)^2[\boldsymbol{s}_\theta(\boldsymbol{x}, t) + \lambda \underbrace{\nabla_{\boldsymbol{x}} \log \max_y \hat{p}(y|\boldsymbol{x})}_{\text{classifier guidance}}]\}\mathrm{d}t,\end{aligned} \tag{5}$$

where $\hat{p}(y|\boldsymbol{x})$ is the classifier confidence estimated by a fully trained classifier. The coefficient $\lambda > 0$ is determined by how much we can trust the classifier. The more accurate the classifier is, the larger value we can take for λ. More discussions on the choice of λ can be found in section 5.4.

Algorithm 1 Classifier-Confidence Guided Purification (COUP) Algorithm.

Input: Perturbed example $\boldsymbol{x}_{\text{adv}}$.
Output: Purified example $\boldsymbol{x}_{\text{ben}}$, predicted label $\hat{y}$.
Required: Trained classifier $f_{\text{cls}}(\boldsymbol{x}) \approx \max_y \hat{p}(y|\boldsymbol{x})$ score function $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}, t) \approx \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$ of fully trained diffusion model, optimal timestep t^*, and a regularization parameter λ.
Set up drift function: $\boldsymbol{f}(\boldsymbol{x}, t) \leftarrow -\frac{1}{2}\beta(t)\boldsymbol{x} - \beta(t)\{\boldsymbol{s}_\theta(\boldsymbol{x}, t) + \lambda \nabla_{\boldsymbol{x}} \log[f_{\text{cls}}(\boldsymbol{x})]\}$
Set up diffusion function: $g(t) \leftarrow \sqrt{\beta(t)}$
Solve SDE for purification according to Eq. 5: $\hat{\boldsymbol{x}}_{\text{ben}} \leftarrow \text{SDE}(\boldsymbol{x}_{\text{adv}}, \boldsymbol{f}(\boldsymbol{x}, t), g(t), t^*, 0)$
Classification: $\hat{y} \leftarrow \arg\max_y f_{\text{cls}}(\hat{\boldsymbol{x}}_{\text{ben}}, y)$
return predicted label $\hat{y}$

In practice, we use VP-SDE [37], such that the drift function and diffusion coefficient are

$$\boldsymbol{f}(\boldsymbol{x}, t) = -\frac{1}{2}\beta(t)\boldsymbol{x}, \quad g(t) = \sqrt{\beta(t)}, \tag{6}$$

where $\beta(t)$ is a linear interpolation from $\beta_{\min}$ to $\beta_{\max}$. The purification process starts from $\boldsymbol{x}_{t^*} = \boldsymbol{x}_{\text{adv}}$ at time $t = t^*$ and ends at time $t = 0$ to get $\boldsymbol{x}_0$. The global purification steps t^* controls the distance between $\boldsymbol{x}_{t^*}$ and $\boldsymbol{x}_0$, which we will later explain in section 4.3.

4.2 The COUP Algorithm

According to the purification rule of Eq. 5, we design our Classifier-cOnfidence gUided Purification (COUP) algorithm in Algo. 1. Our algorithm first set up the drift function and diffusion scale of guided reverse-time SDE according to Eq. 6. Then, we adopt an SDE process from $t = t^*$ to $t = 0$ to get the purified image $\hat{\boldsymbol{x}}_{\text{ben}}$, where the input is adversarial example $\boldsymbol{x}_{\text{adv}}$. Finally, we can use the trained classifier to predict the label of the purified image $\hat{\boldsymbol{x}}_{\text{ben}}$. We omit the forward diffusion process since it yields no positive impact on the objectives of adversarial purification discussed in Section 3, and may cause a potential semantic shift. A detailed discussion can be found in Appendix C.1.

Note that COUP can use the off-the-shelf diffusion model and the fully trained classifier. In other words, we combine the off-the-shelf generative model and the trained classifier to achieve higher robustness.

4.2.1 Adaptive Attack

In order to evaluate our defense method against strong attacks, we propose an augmented SDE to compute the gradient of COUP for gradient-based attacks. In other words, we expose our purification strategy to the attacker to obtain strict robustness evaluation. In this way, we can make a fair comparison with other adversarial defense methods. In Section 4.3, we discuss the key idea of adaptive attack. Suppose $\hat{\boldsymbol{x}}_{\text{ben}}$ is the input of classifier, $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{x}}_{\text{ben}}}$ can be obtained easily obtained. Then we can get the full gradient $\frac{\partial \mathcal{L}}{\partial \boldsymbol{x}_{\text{adv}}}$ according to the augmented SDE. According to the SDE in Eq. 5 with input $\boldsymbol{x}_{\text{adv}}$ and output $\hat{\boldsymbol{x}}_{\text{ben}}$, the augmented SDE is

$$\begin{pmatrix} \boldsymbol{x}_{\text{adv}} \\ \frac{\partial \mathcal{L}}{\partial \boldsymbol{x}_{\text{adv}}} \end{pmatrix} = \text{sdeint}\left(\begin{pmatrix} \hat{\boldsymbol{x}}_{\text{ben}} \\ \frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{x}}_{\text{ben}}} \end{pmatrix}, \tilde{\boldsymbol{f}}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{w}}, 0, t^* \right)$$

where $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{x}}_{\text{ben}}}$ is the gradient of the objective $\mathcal{L}$ w.r.t. the output $\hat{\boldsymbol{x}}_{\text{ben}}$ of the SDE, defined in Eq. 5, and

$$\tilde{\boldsymbol{f}}([\boldsymbol{x}; \boldsymbol{z}], t) = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}, t) - g(t)^2\{\boldsymbol{s}_\theta(\boldsymbol{x}, t) + \nabla_{\boldsymbol{x}} \log[f_{\text{cls}}(\boldsymbol{x})]\} \\ \{\frac{\partial \boldsymbol{f}(\boldsymbol{x}, t) - g(t)^2 \boldsymbol{s}_\theta(\boldsymbol{x}, t)}{\partial \boldsymbol{x}} - g(t)^2 \nabla_{\boldsymbol{x}}^2 \log[f_{\text{cls}}(\boldsymbol{x})]\}\boldsymbol{z} \end{pmatrix},$$

$$\tilde{\boldsymbol{g}}(t) = \begin{pmatrix} -g(t)\mathbf{1}_d \\ \mathbf{0}_d \end{pmatrix}, \quad \tilde{\boldsymbol{w}}(t) = \begin{pmatrix} -\boldsymbol{w}(1-t) \\ -\boldsymbol{w}(1-t) \end{pmatrix},$$

with $\mathbf{1}_d$ and $\mathbf{0}_d$ representing the d-dimensional vectors of all ones and all zeros, respectively. Empirically, we use the stochastic adjoint method [26] to compute the pathwise gradients of Score SDE.

4.3 Analysis of COUP

In this section, we further analyze the effectiveness of our method. First, we show that under the guidance of classifier confidence, our method can better preserve information for classification. Second, under the guided reverse-time VP-SDE, the distance $\|r(\boldsymbol{x}_{\text{adv}}) - \boldsymbol{x}_{\text{adv}}\|$ can be bounded through controlling t^*.

To show that our proposed confidence guidance helps to preserve data information, we give theoretical analysis on a simple case where such guidance can be proved to alleviate the label shift problem. Consider a 1-dimension SDE $\mathrm{d}x = f(x, t)\mathrm{d}t + g(t)\mathrm{d}w$ with starting point $x_{t=0} = x_0 > 0$ and final solution $x_{t=1}$, which is simulated using the Euler method with step size $\Delta t = 1/n$. Denote as $P_{<0}(x_0, f, g)$ the label flip probability such that there exist $t^* \in [0, 1]$ satisfying $x_{t^*} < 0$, we have the following proposition:

Proposition 1. *If for any $t \in [0, 1]$ and $x > 0$, there is $f_0(x, t) < f_1(x, t)$ and $f_0(x, t)$ is strictly monotonically increasing w.r.t. x, then*

$$P_{<0}(x_0, f_1, g) < P_{<0}(x_0, f_0, g). \tag{7}$$

Proposition 1 supports the claim that forces pushing the data away from the decision boundary are helpful to avoid the label shift problem. Consider the case where the data is composed of two classes: one distribution follows $N(\mu, \sigma^2)$ and another follows $N(-\mu, \sigma^2)$, the conditions in Proposition 1 would be satisfied using a VP-SDE (as f_0) and a corresponding SDE with guidance (as f_1), since the added gradient of classifier confidence is always positive on $(0, +\infty)$. In this case, Proposition 1 shows that with the guidance from the ground-truth classifier, it is less likely for a sample to change its label during the purification process. We provide the proof of proposition 1 in Appendix A.1.

Next, we show that the distance between the input sample $\boldsymbol{x}$ and the purified sample $r(\boldsymbol{x})$ can be bounded under our proposed method, thus can avoid severe semantic changes during purification. The result of proposition 2 indicates that for an an adversarial example $\boldsymbol{x}_{\text{adv}}$, the distance $\|r(\boldsymbol{x}_{\text{adv}}) - \boldsymbol{x}_{\text{adv}}\|$ has an upper bound, which is monotonically increasing w.r.t. t^*. As a result, the maximal distance can be controlled by adjusting t^*.

Proposition 2. *Under the assumption that $\|\boldsymbol{s}_\theta(\boldsymbol{x}, t)\| \le \frac{1}{2}C_s$, $\|\nabla_{\boldsymbol{x}} p(\cdot|\boldsymbol{x})\| \le \frac{1}{2}C_p$, and $\|\boldsymbol{x}\| \le C_x$, the denoising error of our guided reverse variance preserving SDE (VP-SDE) can be bounded as*

$$\begin{aligned} \|\boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{x}\| \le \gamma(t^*)(C_s + C_p) + (e^{\gamma(t^*)} - 1)C_x \\ + \sqrt{e^{2\gamma(t^*)} - 1}\,\|\boldsymbol{\epsilon}\|, \end{aligned} \tag{8}$$

Table 1: Accuracy and Robustness against **AutoAttack** under $l_\infty(\epsilon = 8/255)$ threat model on CIFAR-10. The model architecture is reverse-time VP-SDE with $t^* = 0.1$. We use the fully trained WideResNet-28-10 for the classifier.

Defense	Accuracy (%)	**Robustness** (%)
-	96.09	0.00
AWP - w/o Aug [42]	85.36	59.18
GAIRAT [48]	89.36	59.96
Adv. Train - Aug [30]	87.33	61.72
AWP - Aug [42]	88.25	62.11
Adv. Train - Tricks [13]	89.48	62.70
Adv. Train - Aug [14]	87.50	65.24
DiffPure [28]	89.02	70.64
Our COUP	90.04	**73.05**

$\gamma(t^*) := \int_0^{t^*} \frac{1}{2}\beta(s)\mathrm{d}s$, $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ *is the noise added by the reverse-time Wiener process.*

5 Experiments

In this section, we mainly evaluate the adversarial robustness of COUP against AutoAttack [6], including both black-box and white-box attacks. Furthermore, we analyze the mechanism of the classifier confidence guidance through a case study and ablation study to verify the effectiveness of COUP. Besides, we combine our work with the state-of-the-art adversarial training method for further promotion.

5.1 Experimental Settings

Dataset and Models We evaluate our method on CIFAR-10 and CIFAR-100 [22] dataset. To make a fair comparison with other diffusion-based purification methods, we follow the settings of DiffPure [28], evaluating the robustness on randomly sampled 512 images. As for the purification model, we use variance preserving SDE (VP-SDE) of Score SDE [37] with $t^* = 0.1$ for l_∞ threat model and $t^* = 0.075$ for l_2. We select two backbones of the classifier, including WRN-28-10 (WideResNet-28-10) and WRN-70-16 (WideResNet-70-16).

Baselines We compare our method with (1) robust optimization methods, that is the adversarial defense based on discriminative models, also including using generative models for data augmentation; (2) adversarial purification methods based on generative models before classification. Since some diffusion-based adversarial purification methods do not support gradient computation and do not design an adaptive attack, we compare the SOTA method [28] supported by AutoAttack [6].

Evaluation Method We evaluate our method against the AutoAttack [6] against the l_∞ and l_2 threat models and Backward Pass Differentiable Approximation (BPDA) [1]. Since our method contains Brownian motion, we use both *standard* (including three white-box attacks APGD-ce, APGD-t, FAB-t, and one black-box attack Square) and *rand* mode (including two white-box attack APGD-ce, APGD-dlr with EOT=20) of AutoAttack, choosing the worse one to eliminate the 'fake robustness' brought by randomness. Since the white-box plays stronger attack behavior, we evaluate our algorithm across different classifier backbones and other analysis experiments against APGD-ce, one of the white-box in AutoAttack.

Table 2: Accuracy and Robustness against **AutoAttack** under $l_2(\epsilon = 0.5)$ threat model on CIFAR-10. The model architecture is reverse-time VP-SDE with $t^* = 0.075$. We use the fully trained WideResNet-28-10 for the classifier.

Defense	Accuracy (%)	**Robustness** (%)
-	96.09	0.00
Adv. Train - DDN [31]	89.05	66.41
Adv. Train - MMA [9]	88.02	67.77
AWP [42]	88.51	72.85
PORT [33]	90.31	75.39
RATIO [2]	92.23	77.93
Adv. Train - Aug[30]	91.79	78.32
DiffPure [28]	91.03	78.58
Our COUP	92.58	**83.13**

Table 3: Accuracy and Robustness against **BPDA + EOT** [1] under $l_\infty(\epsilon = 8/255)$ threat model on CIFAR-10. The model architecture is reverse-time VP-SDE with $t^* = 0.1$. We use the fully trained WideResNet-28-10 for the classifier.

Defense	Accuracy (%)	**Robustness** (%)
-	96.09	0.00
PixelDefend [36]	95.00	9.00
ME-Net [45]	94.00	15.00
Purification - EBM[16]	84.12	54.90
ADP [46]	86.14	70.01
GDMP [39]	93.50	76.22
DiffPure [28]	89.02	81.40
Our COUP	90.04	**83.20**

5.2 Comparison with Related Work

5.2.1 AutoAttack

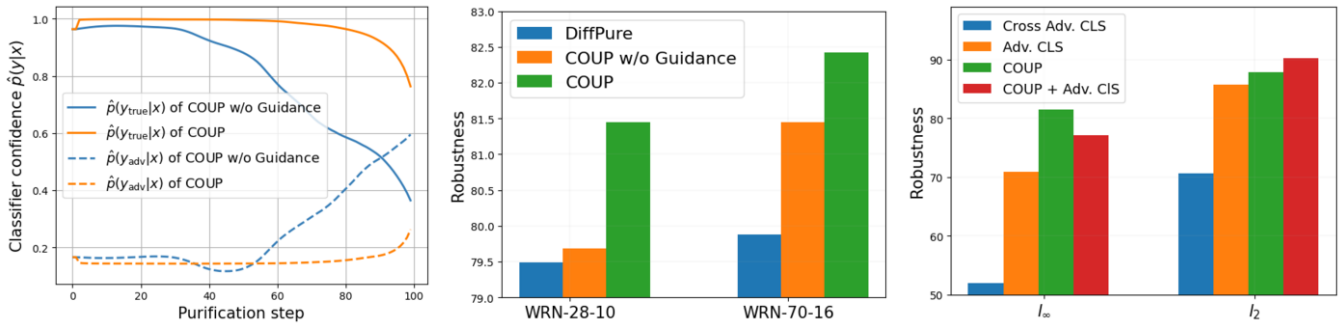
We evaluate our COUP against AutoAttack and compare the robustness with other advanced defense methods on CIFAR-10 according to the results proposed in robustbench [7]. DiffPure [28] is the most related work to ours. According to the results in Table 1 and Table 2, our COUP achieves better robustness (+2.41% for l_∞ and +4.55% for l_2) as well as better accuracy (+1.02% for l_∞ and +1.55% for l_2), showing the effectiveness of classifier guidance enhancing the adversarial robustness through better purification.

Moreover, we also adapt our COUP to the DDPM architecture [18] in consideration of the inference efficiency and evaluate the adversarial robustness against APGD-ce on CIFAR-100. We employ the purification method suggested by Chen et al. [5] and add classifier confidence guidance during likelihood maximization. Our results indicate that COUP attains a 40.55% robustness under the l_∞ threat model with $\epsilon = 8/255$, surpassing the 38.67% achieved by likelihood maximization without classifier guidance.

Besides, owing to the adaptive attack, we can make a fair comparison with other purification algorithms as well as robust optimization methods based on discriminative models. The state-of-the-art method of robust optimization is an improved adversarial training using generated data by diffusion models. COUP is orthogonal to those. We will further discuss the comparison and combination with the SOTA work among them in Section 5.3.

5.2.2 BPDA

We also evaluate the robustness against BPDA + EOT [1] in order to make a comparison with other guided diffusion-based purification methods [43, 39] (since they do not support adaptive attack). The results are represented in Table 3. Considering both Wu



(a) Case Study (b) Ablation Study (c) Combine with Adv. CLS

Figure 2: (a) Expectation of disminative confidence $\hat{p}(y|\boldsymbol{x})$ of WRN-28-10 on label y_{true} and adversarial label y_{adv} over 14 bad cases of reverse-time SDE. (b) Robustness of different denoising methods, including the forward-reverse-time SDE (DiffPure) and the reverse-time SDE only (COUP and COUP w/o Guidance) under different classifier backbones. (c) Robustness of SOTA adversarial training method [40] (marked as Adv. CLS), in both the cross (trained under different l_p norm with evaluation) and non-cross settings, and combined with our COUP against APGD-ce.

et al. [43] and Wang et al. [39] utilize adversarial samples as guidance, we compare with the more proficient GDMP [39] of the two (according to the results from Table 1 of Wu et al. [43]). We test our method on the CIFAR-10 dataset against PGD-20, employing a setting of $\epsilon = 8/255$, $\alpha = 0.007$. The experimental robustness of GDMP achieves 76.22%, DiffPure achieves 81.40%, while our COUP achieves the best of 83.20%. These results verify the effectiveness of our method against BPDA except for adaptive attack and obtain better performance than the adversarial examples guided diffusion-based purification algorithm.

5.3 Experimental Analysis

5.3.1 Case Study

In this part, we plot the curve of $\hat{p}(y_{\text{true}}|\boldsymbol{x})$ and $\hat{p}(y_{\text{adv}}|\boldsymbol{x})$ to show what happens from the view of classifier during purification, where the $\hat{p}(y|\boldsymbol{x})$ is the predict confidence by the classifier of label y. Moreover, we analyze the mechanism of the classifier guidance. To obtain the adversarial examples, we use APGD-ce under the l_∞ threat model to attack the reverse-time SDE (COUP without Guidance) to get bad cases for COUP w/o Guidance. To focus on the purification process, we do not consider Brownian motion at inference time.

According to the analysis in Section 3, we use the predict confidence for ground truth label to evaluate the information preservation degree of the image during purification. Then we plot the curve as shown in Fig. 2a. The rise of $\hat{p}(y_{\text{adv}}|\boldsymbol{x})$ is the reason for successful attack. Meanwhile, the decrease of $\hat{p}(y_{\text{true}}|\boldsymbol{x})$ shows that, during purification, predictive information keeps losing. After 90 steps of purification, $\hat{p}(y_{\text{true}}|\boldsymbol{x})$ suddenly declines to a very low level due to "over purification" and $\hat{p}(y_{\text{adv}}|\boldsymbol{x})$ dominates the prediction confidence, which leads to vulnerability. Next, we further explore the mechanism of classifier guidance. After adding classifier guidance, $\hat{p}(y_{\text{true}}|\boldsymbol{x})$ obtains a rapid rise under the guidance of the classifier at the very beginning. Besides, the guidance of the classifier alleviates the information loss (also weakens the influence of adversarial perturbation) during purification and finally results in correct classification.

5.3.2 Analysis of Information Preservation on Toy Data

To verify the effectiveness of our method for information preservation, we use 2-Gaussian toy data and run a simulation of pure SDE and COUP to show that COUP can alleviate the label shift problem. The data distribution is a 1-dimension uniform mixture of 2 symmetric Gaussian distributions $\mathcal{N}(-0.5, 1)$ and $\mathcal{N}(0.5, 1)$, the data from which we label as $y = 0$ and $y = 1$, respectively. Starting from the point $x_0 = 0.2$, we apply the COUP algorithm with guidance weight ranging from 0 to 10.0. To simulate the adversarial vulnerability of the classifier, we use a noisy classifier $p(y = 1|x) = \frac{p_1(x)}{p_0(x)+p_1(x)+c\cdot n(x)}$, where $p_i(x)$ is the density function of class i, $n(x) = \frac{sin(100x)}{100}$ is the noise and c is the noise level. We apply the Euler method for SDE simulation, using step size $1e{-}3$ and $t^* = 0.1$. We run 100,000 times and estimate the label flip probability, i.e. $p(x_{t^*} < 0)$. The result in Fig. 3b shows that the guidance signal is overall helpful to keep the label unchanged under small or no noise. When the noise level is high, the classifier becomes untrustworthy. Thus, an appropriate λ should be chosen.

5.3.3 Analysis on Different Classifier Architectures

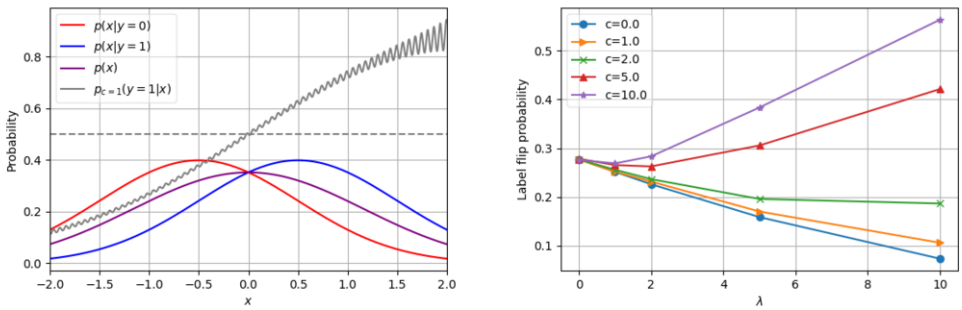
In order to evaluate the effectiveness of our guidance method for different architectures of classifiers, we adapt our purification method to both WRN-28-10 and WRN-70-16 against APGD-ce attack (under l_∞). The results in Fig. 2b show that COUP achieves better robustness on both two classifier backbones. In other words, our method is effective across different classifier architectures.

5.3.4 Ablation Study

In order to demonstrate the effectiveness of classifier guidance, we evaluate the robustness of COUP and COUP w/o Guidance (i.e. reverse-time SDE) against APGD-ce attack (under l_∞) as an ablation. Results in Fig. 2b support that the robustness promotion in Table 1 and Table 2 of our COUP is mainly caused by the classifier guidance instead of the structure of diffusion model (since we remove the forward process from DiffPure).

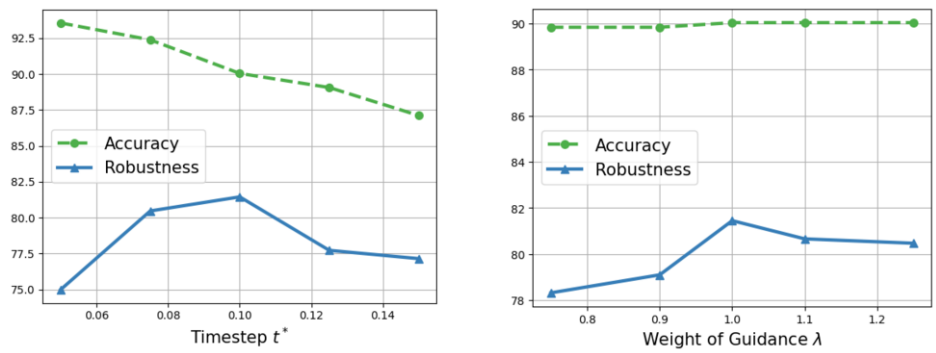
5.4 Hyperparameters

Analysis on Purification Timestep t^* Since the purification timestep t^* is a critical hyperparameter deciding the degree of denoising, we evaluate the robustness against APGD-ce across different t^* and find it performs the best at $t^* = 0.1$. The experimental result



(a) Probability on toy data (b) Label flip probability

Figure 3: (a) The 2-Gaussian toy data and classifier with noise level $c = 1$. (b) Label flip probability under different noise levels c and guidance weight λ on 2-Gaussian toy data.



(a) Accuracy and robustness across t^* (b) Accuracy and robustness across λ

Figure 4: Accuracy and Robustness against APGD-ce under $l_\infty(\epsilon = 8/255)$ threat model for (a) variant purification timestep t^* and (b) variant weight of guidance λ.

in Fig. 4a shows that insufficient purification step or "over purification" both leads to lower robustness. This phenomenon strongly supports that balancing denoising and information preservation is very important. Besides, it is intuitive that accuracy decreases as timestep t^* grows.

Analysis on Guidance Weight λ We experimentally find that COUP obtains the highest robustness against APGD-ce under $\lambda = 1$. That is, the diffusion model and the classifier have equal weight. Note that it implements the same effect as a conditional generative model (according to the Bayes formula: $p(\boldsymbol{x}) \cdot p(y|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|y)}{p(y)}$ since the prior probability $p(y)$ of category y is considered as uniform).

5.4.1 *Comparison and Combination with SOTA of Adversarial Training*

Considering the settings of robust evaluation, we argue that it is unfair to compare our COUP with the adversarial training algorithm. The reason is we do not make any assumption about the attack, while the adversarial training methods are specifically trained for the evaluation attack. Therefore, we additionally evaluate the SOTA [40] of adversarial training under both cross and non-cross settings. Specifically, in the cross-setting, we use the model trained for l_2 (l_∞) to defend the attack under l_∞ (l_2). The results in Table 2c show that it [40] suffers a severe robustness drop under the cross-setting. In other words, its robustness becomes poor when defending against unseen attacks.

Besides, to take advantage of their work [40], we combine our purification method with the adversarially trained classifier. When the classifier has better clean accuracy (95.16% under l_2), it can further improve the robustness against APGD-ce attack. However, worse accuracy under l_∞ (92.44%) may provide inappropriate guidance for purification. Note that, in that case, our purification method COUP further improves their robustness from 70.90% to 77.15%.

6 Conclusion

To address the principal challenge in purification, i.e., achieving a balance between noise removal and information preservation, we employ the concept of the classifier-guided purification method. We discover that classifier-confidence guidance aids in preserving predictive information, which facilitates the purification of adversarial examples towards the category center. Specifically, we introduce Classifier-cOnfidence gUided Purification (COUP) and have assessed its performance against AutoAttack and BPDA, comparing it with other advanced defense algorithms under the RobustBench benchmark. The results demonstrated that our COUP achieved superior adversarial robustness.

Acknowledgements

This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDB0680101, CAS Project for Young Scientists in Basic Research under Grant No. YSBR-034, Innovation Project of ICT CAS under Grant No. E261090, and the project under Grant No. JCKY2022130C039. This paper acknowledges the valuable assistance of Xiaojie Sun in the experiment execution.

References

[1] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018.

[2] M. Augustin, A. Meinke, and M. Hein. Adversarial robustness on in-and out-distribution improves explainability. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 228–245. Springer, 2020.

[3] W. Brendel, J. Rauber, and M. Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017.

[4] N. Carlini, F. Tramer, J. Z. Kolter, et al. (certified!!) adversarial robustness for free! *arXiv preprint arXiv:2206.10550*, 2022.

[5] H. Chen, Y. Dong, Z. Wang, X. Yang, C. Duan, H. Su, and J. Zhu. Robust classification via a single diffusion model. *arXiv preprint arXiv:2305.15241*, 2023.

[6] F. Croce and M. Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.

[7] F. Croce, M. Andriushchenko, V. Sehwag, E. Debenedetti, N. Flammarion, M. Chiang, P. Mittal, and M. Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

[8] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[9] G. W. Ding, Y. Sharma, K. Y. C. Lui, and R. Huang. Mma training: Direct input space margin maximization through adversarial training. *arXiv preprint arXiv:1812.02637*, 2018.

[10] Y. Du and I. Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[12] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[13] S. Gowal, C. Qin, J. Uesato, T. Mann, and P. Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

[14] S. Gowal, S.-A. Rebuffi, O. Wiles, F. Stimberg, D. A. Calian, and T. A. Mann. Improving robustness using generated data. *Advances in Neural Information Processing Systems*, 34:4218–4233, 2021.

[15] W. Grathwohl, K.-C. Wang, J.-H. Jacobsen, D. Duvenaud, M. Norouzi, and K. Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

[16] M. Hill, J. Mitchell, and S.-C. Zhu. Stochastic security: Adversarial defense using long-run dynamics of energy-based models. *arXiv preprint arXiv:2005.13525*, 2020.

[17] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[18] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[19] H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and T. Zhao. Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. *arXiv preprint arXiv:1911.03437*, 2019.

[20] B. Kawar, R. Ganz, and M. Elad. Enhancing diffusion-based image synthesis with robust classifier guidance. *arXiv preprint arXiv:2208.08664*, 2022.

[21] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[22] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[23] C. Laidlaw, S. Singla, and S. Feizi. Perceptual adversarial robustness: Defense against unseen threat models. *arXiv preprint arXiv:2006.12655*, 2020.

[24] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[25] X. Li and S. Ji. Defense-vae: A fast and accurate defense against adversarial attacks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 191–207. Springer, 2019.

[26] X. Li, T.-K. L. Wong, R. T. Chen, and D. K. Duvenaud. Scalable gradients and variational inference for stochastic differential equations. In *Symposium on Advances in Approximate Bayesian Inference*, pages 1–28. PMLR, 2020.

[27] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[28] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*, 2022.

[29] C. Qin, J. Martens, S. Gowal, D. Krishnan, K. Dvijotham, A. Fawzi, S. De, R. Stanforth, and P. Kohli. Adversarial robustness through local linearization. *Advances in Neural Information Processing Systems*, 32, 2019.

[30] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, and T. Mann. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021.

[31] J. Rony, L. G. Hafemann, L. S. Oliveira, I. B. Ayed, R. Sabourin, and E. Granger. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4322–4330, 2019.

[32] P. Samangouei, M. Kabkab, and R. Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *arXiv preprint arXiv:1805.06605*, 2018.

[33] V. Sehwag, S. Mahloujifar, T. Handina, S. Dai, C. Xiang, M. Chiang, and P. Mittal. Robust learning meets generative models: Can proxy distributions improve adversarial robustness? *arXiv preprint arXiv:2104.09425*, 2021.

[34] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[35] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[36] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *arXiv preprint arXiv:1710.10766*, 2017.

[37] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[38] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[39] J. Wang, Z. Lyu, D. Lin, B. Dai, and H. Fu. Guided diffusion model for adversarial purification. *arXiv preprint arXiv:2205.14969*, 2022.

[40] Z. Wang, T. Pang, C. Du, M. Lin, W. Liu, and S. Yan. Better diffusion models further improve adversarial training. *arXiv preprint arXiv:2302.04638*, 2023.

[41] E. Wong, L. Rice, and J. Z. Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

[42] D. Wu, S.-T. Xia, and Y. Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020.

[43] Q. Wu, H. Ye, and Y. Gu. Guided diffusion model for adversarial purification from random noise. *arXiv preprint arXiv:2206.10875*, 2022.

[44] C. Xiao, Z. Chen, K. Jin, J. Wang, W. Nie, M. Liu, A. Anandkumar, B. Li, and D. Song. Densepure: Understanding diffusion models towards adversarial robustness. *arXiv preprint arXiv:2211.00322*, 2022.

[45] Y. Yang, G. Zhang, D. Katabi, and Z. Xu. Me-net: Towards effective adversarial robustness with matrix estimation. *arXiv preprint arXiv:1905.11971*, 2019.

[46] J. Yoon, S. J. Hwang, and J. Lee. Adversarial purification with score-based generative models. In *International Conference on Machine Learning*, pages 12062–12072. PMLR, 2021.

[47] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.

[48] J. Zhang, J. Zhu, G. Niu, B. Han, M. Sugiyama, and M. Kankanhalli. Geometry-aware instance-reweighted adversarial training. *arXiv preprint arXiv:2010.01736*, 2020.