RECOVARIT : Leveraging image identity improves adversarial robustness against simple transformations

Anonymous Authors[1]

Abstract

We introduce the regularized adversarial training framework RECOVARIT to improve adversarial robustness against simple transformations. The regularizer leverages the image identity (ID) of adversarial and original examples and can be combined with any type of adversarial training. It penalizes the variance of the predictions for images that share the same ID. The effectiveness of this conditional variance penalty is theoretically motivated by robustness guarantees in low-dimensional linear settings. Empirically, we show that RECOVARIT significantly improves adversarial robustness against simple transformations for a wide range of settings on various datasets at no additional computational overhead.

1. Introduction

As deployment of machine learning (ML) systems in the real world has steadily increased over recent years, more and more emphasis is placed on robustness and reliability of the algorithms. This is particularly important in applications where incorrect predictions could result in harming humans, such as self-driving cars and medical diagnosis. For example, the vision in the car should correctly classify an obstacle independent of the background, or a cancer cell should be marked as such irrespective of the position of the patient in the measuring device.

The requirement for reliability in the presence of perturbations is generally captured in the concept of *adversarial robustness* which has become a valuable evaluation metric alongside predictive accuracy on the original test set (the *natural test accuracy*). Models which achieve high natural test accuracy in many applications such as neural networks, have been shown to fail dramatically in terms of adversarial robustness. This behavior has lead to a surge of interest and related papers in the machine learning community.

While a lot of research on adversarial robustness has focused on additive perturbations in the high-dimensional input space with bounded ℓ_p norm (Szegedy et al., 2014; Moosavi-Dezfooli et al., 2016; Kurakin et al., 2016a; Papernot et al., 2017; Carlini & Wagner, 2017; Madry et al., 2018; Samangouei et al., 2018; Sinha et al., 2018; Raghunathan et al., 2018; Wong & Kolter, 2018; Mirman et al., 2018), relatively few papers have rigorously addressed robustness of ML systems to *simple transformations*, which can be described by a low-dimensional parameter. When capturing natural or medical images for example, slight relative rotations and translations of the object may occur. These can be noticeable but are generally ignored by human perception. Convolutional Neural Networks (CNN) have been shown to be sensitive not only to high-dimensional perturbations in the input space, but also to simple perturbations (Fawzi & Frossard, 2015; Engstrom et al., 2017; Pei et al., 2017; Geirhos et al., 2018; Alcorn et al., 2018). While this vulnerability is also observed in simpler models (Papernot et al., 2016) and our framework can generally be applied to any parameterized function space, this paper focuses on neural networks in the context of computer vision due to their tremendous success and widespread use in applications.

Driven by the motivation of increasing natural test accuracy, previous work has addressed the quest for invariance of neural networks to simple spatial transformations such as rotations and translations in various ways. The approaches involve either (i) a change in network architecture or (ii) a change in data. In principle these two approaches can be combined ad libitum. The idea in the first line of work is to change the structure of the neural network to hard-code a known invariance such as a rotation, scaling or translation (Jaderberg et al., 2015; Laptev et al., 2016; Marcos et al., 2017; Weiler et al., 2018). Our framework is based on the second approach (ii) that uses the concept of data augmentation (Baird, 1992; Yaeger et al., 1997). When an invariance to a particular transformation is desired, the augmented procedure performs gradient updates on randomly transformed versions of the original training samples in each training step, inducing the model to correctly classify a sam-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

ple irrespective of the transformation. A major advantage of augmentation based methods is their applicability to any kind of invariance. For example, sometimes the best invariance has to be learned via data because prior knowledge is not available. In such cases, one could augment the original training set by automatically generated samples as proposed in recent work (Tran et al., 2017; Antoniou et al., 2018) or learn an augmentation policy e.g. using reinforcement learning (Ratner et al., 2017; Cubuk et al., 2018).

Besides increasing natural test accuracy, data augmentation is also a core concept in *adversarial training*, a standard defense technique introduced in the context of attacks in high-dimensional space. Adding a slight twist to standard data augmentation, the update step in adversarial training uses examples that are freshly generated in every iteration depending on the current model. It has been shown to improve adversarial robustness of the final classifier for both high-dimensional perturbations and simple transformations (see e.g. (Szegedy et al., 2014; Engstrom et al., 2017)). We propose a technique based on adversarial training to substantially improve on adversarial robustness for simple transformations.

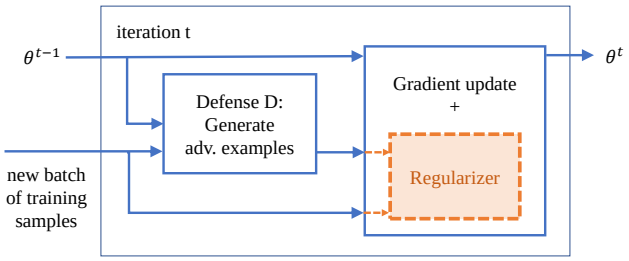


Figure 1. Schematic for regularized adversarial training. Standard adv. training with a given defense mechanism is illustrated with solid blue lines. In RECOVARIT, we propose to add a conditional variance regularizer which is shown in orange with dashed lines.

1.1. Our contributions

In the schematic in Fig. 1, we highlight the modifications (dashed lines) we propose to standard adversarial training (solid lines). Specifically, in addition to the two steps in regular adversarial training, consisting of the *defense mechanism* creating adversarial examples and the gradient update (potentially on both original and adversarial examples), we propose to add a regularizer to the gradient update. Our idea is based on the observation that both standard and adversarial data augmentation fail to use an important piece of information: that different transformed samples are generated from the same original image and thus have the same identity (ID). Exploiting image ID should therefore allow for better adversarial robustness.

This intuition can be made more precise using a probabilistic generative model. In this model, an ideal classifier should output prediction probabilities that are invariant to nuisance transformations. Having multiple transformed examples with the same identity can be used to explicitly enforce this property on our target classifier using a regularizer which penalizes the conditional variance given a fixed image ID. This conditional variance penalty was introduced in the context of domain shift robustness in (Heinze-Deml & Meinshausen, 2017). We abbreviate our framework as "RECOVARIT" as it combines regularization with conditional variance for adversarial training.

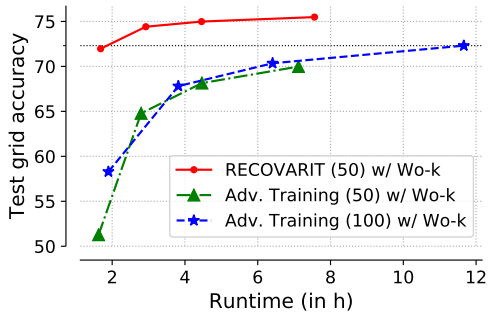


Figure 2. Adversarial accuracy (test grid acc.) as a function of runtime on CIFAR-10 for our proposed framework RECOVARIT and two versions of adversarial training. The different points for each method correspond to various strengths of the chosen defense mechanism. Further details are discussed in Sec. 4.

Special cases of RECOVARIT have been explored in the context of ℓ_∞ robustness in (Kannan et al., 2018; Engstrom et al., 2018; Mosbach et al., 2018) with little to no improvement for the strongest attacks. We show that for simple transformations which involve translations and rotations, the story is different. For low-dimensional perturbations, RECOVARIT exhibits the following features for a variety of defenses and datasets.

- **Gain in adversarial robustness:** When used on top of adversarial training with status quo defenses, it achieves $\sim 20\%$ relative adversarial error reduction across three datasets computed using the state of the art attack.
- **Computational efficiency:** RECOVARIT requires the same or less amount of training time compared to unregularized adversarial training while improving performance significantly as shown in Fig. 2.
- **Outperforms data augmentation:** In a mixed-batch setting detailed in Sec. 2.2, RECOVARIT with randomly augmented samples achieves relative error reductions of $24 - 45\%$ while the natural accuracy degrades at most by the standard error compared to standard data augmentation. This observation suggests using RECOVARIT as a replacement for standard data augmentation without having to sacrifice runtime.

2. Regularized adversarial training

In this section we define the adversarial setting and introduce our general framework for regularized adversarial training. We focus on classification as the prediction task of interest.

2.1. Data generating model

We assume that our observations $X \in \mathbb{R}^d$, ID $\in \mathbb{Z}$, class $Y \in \mathbb{R}$ are random variables drawn from a joint distribution $\mathbb{P}$ over (ID, X, Y), modeled as a directed graphical model as depicted in Fig. 3. The sources of randomness are Δ, ID. The generative probabilistic model is rooted in the intuition that the process of collecting an observation always starts with randomly picking an object with a certain ID with associated attributes $Z = [Z^{\text{style}}, Z^{\text{core}}]$ (usually unobserved) which determine its identity uniquely. The class Y is a deterministic function of the core attributes Z^{core} and $X^{\text{ID}} \in \mathbb{R}^d$ is an arbitrary function of Z and assumed to be deterministic as well. The observed image $X \in \mathbb{R}^d$ is a randomly transformed version of X^{ID}, i.e. $X = \mathcal{T}(X^{\text{ID}}, \Delta)$, where the perturbation $\Delta \in \mathcal{S} \subset \mathbb{R}^q$ arises from the concrete measurement process. For instance, the transformation $\mathcal{T}(\cdot, \Delta)$ with perturbation Δ could consist of a rotation with degree $\Delta \in [-30°, 30°]$.

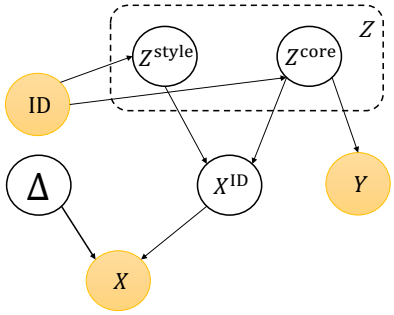


Figure 3. Generative model for observations ID, X, Y. ID is a unique identifier variable encoding the identity of the underlying object. $\mathcal{T}(X^{\text{ID}}, \Delta)$ with perturbation Δ describes the transformation of X^{ID} arising in the measurement process recording X.

To further illustrate the model, consider images recorded by a self-driving car. These could be categorized into classes such as truck, person, sidewalk, etc. A particular truck has relevant attributes Z^{core} such as relative height, rear and front shape as well as nuisance attributes Z^{style} such as color. A transformation could be the background color. No matter what time of day or how bright the sky—a truck should always be recognized as such. We define this goal more formally below. In medical imaging, the prediction target of interest could be the segmentation of a tumor, X^{ID} would correspond to the image in the orientation which makes it easiest for the doctor to diagnose, Z^{core} to the characteristics of the actual tissue and Z^{style} would be related to the specifics of the imaging machine. Δ could be the position of the patient inside the machine. Transformations could in principle be of any kind—noise, filters, occlusions, rotations, scaling, pose, background. In this paper, we consider low-dimensional transformations with $q \ll d$ which could naturally occur in the capturing process and compact invariance sets $\mathcal{S}$.

2.2. Adversarial loss and training

The classifier f_θ is parameterized by $\theta \in \mathbb{R}^p$ and we denote the sample-wise loss function on an observed pair (X, Y) as $\ell(f_\theta(X), Y)$ with examples being the 0-1 or cross entropy loss for classification. In adversarial robustness, the goal is to achieve the same 0-1 loss for all observable images X with the same identity, that is for transformations with any perturbation in the support of the distribution over Δ. This allows the practitioner to know that even the most adversarial perturbation Δ would not have influenced the probability of correct classification. The following loss function, common in the robust optimization literature (Ben-Tal et al., 2009), captures this adversarial concept[1]

$$\mathcal{L}_{\text{adv}}(\theta) = \mathbb{E} \max_{\Delta' \in \mathcal{S}} \ell(f_\theta(\mathcal{T}(X, \Delta')), Y). \tag{1}$$

Ideally, we want to find the parameter which minimizes the adversarial loss in Eq. (1) with the expectation taken over the distribution $\mathbb{P}$. In reality, we only have access to a finite number of $i = 1, \ldots, n$ unique observed tuples (ID_i, X_i, Y_i) with $n > p$ and aim to find the minimizer of $\mathcal{L}_{\text{adv}}$ with respect to the empirical distribution. To solve the optimization problem, it is common practice in large-scale applications to use a first-order optimization algorithm, such as stochastic gradient descent (SGD), where the updates in each iteration are computed using minibatches. For a new batch I^t of unique indices, the corresponding batch-wise loss reads $\mathcal{L}^t_{\text{adv}}(\theta) := \frac{1}{|I^t|} \sum_{i \in I^t} \max_{\Delta' \in \mathcal{S}} \ell(f_\theta(\mathcal{T}(X_i, \Delta')), Y_i)$. The gradient of $\mathcal{L}^t_{\text{adv}}$ can be rewritten as follows

$$\nabla_\theta \mathcal{L}^t_{\text{adv}}(\theta) = \nabla_\theta \mathcal{L}^t(\theta; B^t_{\text{adv}}), \tag{2}$$

where the right hand side is defined by the average loss

$$\mathcal{L}^t(\theta; B^t_{\text{adv}}) = \frac{1}{|B^t_{\text{adv}}|} \sum_{(x,y) \in B^t_{\text{adv}}} \ell(f_{\theta^{t-1}}(x), y). \tag{3}$$

The batch B^t_{adv} consists of the adversarial examples $X^\star_i(\theta^t)$ with respect to the parameter at iteration t

$$B^t_{\text{adv}} = \cup_{i \in I^t} \{(X^\star_i(\theta^{t-1}), Y_i)\}. \tag{4}$$

[1] Strictly speaking, in order to be robust against all $\Delta \in \mathcal{S}$ on X^{ID}, we have to choose a larger set $\widetilde{\mathcal{S}}$ than the domain $\mathcal{S}$ of Δ such that $\{\mathcal{T}(X^{\text{ID}}, \Delta) \mid \Delta \in \mathcal{S}\} \subset \{\mathcal{T}(\mathcal{T}(X^{\text{ID}}, \Delta), \Delta') \mid \Delta \in \mathcal{S}, \Delta' \in \widetilde{\mathcal{S}}\}$. Also note that in Eq. (1), Δ' is independent of the random variable Δ which is drawn from $\mathbb{P}$.

We refer to B_{adv} as the full adversarial batch. Mathematically the adversarial examples are defined as $X_i^\star(\theta^t) := \mathcal{T}(X_i, \Delta^\star(\theta^t))$ given $\Delta^\star(\theta^t) := \arg\max_{\Delta\in\mathcal{S}} \ell(f_{\theta^t}(\mathcal{T}(X_i,\Delta)), Y_i)$. Equality (2) follows from Danskin's Theorem (Danskin, 1966), with a directly relevant discussion in (Madry et al., 2018; Sinha et al., 2018).

2.3. Regularization with conditional variance for adversarial training (RECOVARIT)

The gradient step with the loss defined in Eq. (3) involves finding the adversarial example $X_i^\star(\theta^t)$ in every iteration t for each sample i. Effective *defense mechanisms* which aim to find it scalably in practice often run a limited number of first-order steps (sometimes k of them with random restarts) or choose among k random draws from the invariance set $\mathcal{S}$ (see e.g. (Szegedy et al., 2014; Engstrom et al., 2017) and Sec. 3.4.1). However, unless special assumptions are made on the loss function and perturbations, the inner maximization problem in Eq. (1) is in general non-concave and hence even first-order methods are not even guaranteed to converge, let alone to a global optimum (Sinha et al., 2018). This implies that in practice, the solution found by adversarial training, as defined in Eq. (3), may in general be far from the global optima of the population loss in Eq. (1).

In the quest to find a solution to mitigate this issue, we first present a generalization result for the minimizer of the natural loss regularized with a conditional variance penalty. We examine a simple linear setting and assume we observe M randomly sampled transformations per unique image ID. Let us furthermore make the following assumptions:

(A1) $\mathcal{T}(X^{\text{ID}}, \Delta) = X^{\text{ID}} + W\Delta$ where matrix $W \in \mathbb{R}^{p\times q}$ has full rank q.

(A2) The function space over which we optimize is parameterized as $f_\theta(x) = f(\theta^\top x)$ for some $f : \mathbb{R} \to \mathbb{R}$. We consider the constrained optimization over $\Theta := \{\theta : \mathbb{E}\,\text{Var} f_\theta \mathcal{T}(X,\Delta) = 0\}$.

(A3) The loss is uniformly bounded, i.e. $\|\ell\|_\infty \le B$.

We define the following computable minimizer

$$\widehat{\theta} := \arg\min_{\theta\in\Theta} \frac{1}{n(M+1)} \sum_{i=1}^{n}\sum_{m=0}^{M} \ell(f_\theta(\mathcal{T}(X^{\text{ID}}, \Delta_{i,m})), Y_i) + \lambda \underbrace{[f_\theta(\mathcal{T}(X^{\text{ID}}, \Delta_{i,m})) - \mathbb{E}_{\ i} f_\theta(\mathcal{T}(X^{\text{ID}}, \Delta))]^2}_{\text{empirical conditional variance}} \quad (5)$$

where $\mathbb{E}_{\ i}$ is the average over $\Delta_{i,m}$, and let $\theta^\star = \arg\min_{\theta\in\Theta} \mathbb{E}_{\text{ ID}} \max_{\Delta\in\mathcal{S}} \ell(f_\theta(\mathcal{T}(X^{\text{ID}},\Delta)), Y)$.

Theorem 1. *Under the assumptions (A1)-(A3) and regular conditions on the marginal distribution of Δ, we have with probability at least $1-\delta$ that*

$$|\mathcal{L}_{adv}(\widehat{\theta}) - \mathcal{L}_{adv}(\theta^\star)| \le c_1\sqrt{\frac{B\log\delta}{n}} + c_2\sqrt{\frac{B}{n}} D(f,B),$$

where $D(f,B)$ is a quantity dependent on f, B.

The detailed assumptions and proof can be found in Sec. A.1. Albeit for a simple setting, Theorem 1 gives intuition why penalizing the conditional variance in principle might help to find a classifier which achieves close to optimal adversarial test accuracy.

Based on the regularized estimator in Eq. (5), RECOVARIT exploits the fact that adversarial defense mechanisms often involve generating multiple perturbations $\Delta_{i,m}^t$ per sample i in each batch t and runs a gradient update on the loss

$$\mathcal{L}_R^t(\theta; B^t) = \mathcal{L}^t(\theta; B^t) + \lambda \frac{1}{|I^t|(M+1)} \sum_{i\in I^t}\sum_{m=0}^{M} [f_{\theta^{t-1}}(\mathcal{T}(X_i, \Delta_{i,m}^t)) - \mathbb{E}_i^t f_{\theta^{t-1}}(\mathcal{T}(X_i,\Delta))]^2 \quad (6)$$

where $\mathbb{E}_i^t$ is the average over the perturbations at iteration t. The batch B^t may be either fully adversarial as in Eq. (4) or mixed with original samples, that is $B^t = B_{\text{mix}}^t$ with

$$B_{\text{mix}}^t = \cup_{i\in I^t}\{(\mathcal{T}(X_i, \Delta_{i,m}^t), Y_i)\}_{m=0}^M \quad (7)$$

where $\Delta_{i,1}^t, \ldots, \Delta_{i,M}^t$ are transformations dependent on θ^{t-1} and $\Delta_{i,0}^t = 0$. We introduce the mixed batch variant of adversarial training in Eq. (7) since it has been observed in (Goodfellow et al., 2014; Kurakin et al., 2016b), that with $B^t = B_{\text{mix}}^t$ and $M = 1$, adversarial training with the batch-wise loss in Eq. (3) leads to better natural test accuracy compared to $B = B_{\text{adv}}^t$, without sacrificing adversarial robustness.

The second term in Eq. (6) corresponds to the conditional variance of $f_\theta(\mathcal{T}(X,\Delta))$ conditioned on ID taken over the empirical distribution of Δ. In practice, we apply the conditional variance penalty to the logits of the classifier. We present empirical evidence of the effectiveness of adversarial training using the regularized loss (6) in Sec. 4.

2.4. Evaluation with ϵ-adversarial accuracy

In order to compare adversarial robustness of the different methods, we compute the *adversarial accuracy*

$$A_\theta := 1 - \mathbb{E}\max_{\Delta'\in\mathcal{S}} \mathbb{I}\{f_\theta(\mathcal{T}(X,\Delta')) \ne Y\} \quad (8)$$
$$\le \mathbb{P}(q_\theta(X) = 0) \quad \text{w/} \quad q_\theta(X) = P(f_\theta(\mathcal{T}(X,\Delta')) \ne Y)$$

and $\mathbb{I}\{A\} = 1$ if A holds and 0 otherwise and P is a probability distribution over Δ', independent of the random variable Δ drawn from $\mathbb{P}$.[2] The random variable $q_\theta(X)$

[2]For discrete sets $\mathcal{S}$, the inequality becomes an equality.

is the proportion of transformations in $\mathcal{S}$ with which the classifier can be fooled on the image X. We now consider *ϵ-adversarial accuracy* as a strict generalization of the upper bound in Eq. (8) which is defined via the cumulative distribution function of q_θ at ϵ:

$$A_\theta(\epsilon) = \mathbb{P}(q_\theta(X) \leq \epsilon). \tag{9}$$

We argue that this weaker adversarial measure can be useful to report for small ϵ, since a significant proportion of samples are not robust against all $\Delta \in \mathcal{S}$. For these samples, we can provide an upper bound on the proportion of transformations which could fool the classifier.

3. Experimental setup

We now present the performance evaluation of RECOVARIT for various datasets and defense mechanisms.

3.1. Implementing RECOVARIT

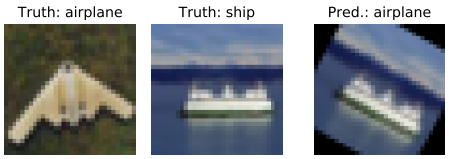
We first provide a high-level overview of RECOVARIT in Algorithm 1. An epoch is defined as a complete pass over the original training set with n unique images. The total number of epochs is denoted by T. In every epoch, we see M different adversarial examples per original image, which results in the algorithm seeing a total of $T \cdot M \cdot n$ different images over the course of the entire training when using B_{adv} and $T \cdot M \cdot n + n$ different images with B_{mix}.

Algorithm 1 Training with RECOVARIT

Input: training set $(X_1, Y_1), \ldots, (X_n, Y_n)$, transformation $\mathcal{T}(\cdot, \cdot)$ and invariance set $\mathcal{S}$, defense mechanism D, M, reg. param. λ, function space of f_θ
for epoch $t = 1$ **to** T **do**
  for batch $j = 1$ **to** $\lfloor n/|I^t| \rfloor$ **do**
    Get the next batch I^t of unique images
    for each image $i = 1$ **to** $|I^t|$ **do**
      Compute M adv. examples $\mathcal{T}(X_i, \Delta_m^t)$ using D
    end for
    Compute gradient of reg. loss (6) with B_{adv}^t or B_{mix}^t
    Perform gradient update
  end for
end for

The regularization term in Eq. (6) can be computed efficiently as discussed further below. Furthermore, Algorithm (1) allows flexibility in the choice of the transformation space and invariance set $\mathcal{S}$, the adversarial defense mechanism, loss function ℓ, parameterized function space for f_θ depending on the best practices in the field of interest.

Naming convention In the experiments, we always ensure that the batch size b, defined as the number of samples entering the loss $\mathcal{L}^t(\theta; B^t)$, stays constant across methods while the number of examples entering the regularizer differs between 0, b and $2b$. We refer to the version with a full adversarial batch $B^t = B_{\text{adv}}^t$ as RECOVARIT (100) and with $B^t = B_{\text{mix}}^t$ as RECOVARIT (50). This notation also holds for adversarial training. For the full name of the procedure, the defense mechanism is added as a suffix. Below we report numbers for RECOVARIT(50) mainly as the improvements in accuracy compared to RECOVARIT (100) were small at an increase in runtime. Unregularized adversarial training is run for both settings.



(a) (b) (c)

Figure 4. Example images and classifications by the Standard model. (a) An image that is misclassified for all rotations in the considered grid. (b) An image that is correctly classified for most of the rotations in the considered grid. (c) One rotation for which the image shown in (b) is misclassified as "airplane".

3.2. Function spaces and datasets

The experiments in this paper are conducted using deep neural networks, parameterized by weights θ, for functions f_θ and ℓ is the cross-entropy loss. We consider the datasets SVHN (Netzer et al., 2011), CIFAR-10 and CIFAR-100 (Krizhevsky & Hinton, 2009). For all experimental results in Sec. 4, we use a ResNet-32 (He et al., 2016) for SVHN and CIFAR-10 and a ResNet-50 for CIFAR-100. For a subset of the experiments we also report results using a VGG architecture (Simonyan & Zisserman, 2015). All models are implemented in TensorFlow (Abadi et al., 2015) and we describe the training procedure in Sec. A.2.[3]

We train the baseline models with standard data augmentation schemes consisting of random left-right flipping, random translations of $\pm$4px and per image standardization. Below we refer to the models trained in this fashion as "Standard". For the models trained with one of the defenses described in Sec. 3.4.1, we apply random left-right flipping and per image standardization; translations are part of the adversarial training which is why they are not additionally included in the standard data augmentation step.

3.3. Transformations and invariance set $\mathcal{S}$

The transformations $\mathcal{T}(\cdot, \cdot)$ that we consider in our experiments, are rotations and translations in 2 dimensions, thus corresponding to perturbations $\Delta \in \mathbb{R}^3$. As we aim to study the robustness with respect to transformations which may occur naturally and to ensure visual similarity, we allow rotations of up to only 30 degrees and translations of up to 3 pixels in all directions, corresponding to approximately

[3] Code is submitted in the Supp. Mat..

10% of the image size.[4] The empty space that results from translating and rotating an image is filled with black pixels.[5] One pair of example images is shown in panels (b) and (c) in Fig. 4, where the original image is depicted along with a transformed version $\mathcal{T}(\cdot, \Delta)$ with $\Delta \in \mathcal{S}$.

3.4. Attacks and defenses

The attacks and defenses we choose are largely following the setup in (Engstrom et al., 2017).

3.4.1. Considered defenses

As defenses we compare regularized and unregularized variants of adversarial training.

Worst-of-k. In worst-of-k (Wo-k) adversarial training, at iteration t, k different perturbations are randomly sampled for each X_i and the one with the highest loss is chosen as $X_i^\star(\theta^t)$. Choosing the worst transformation from these k randomly generated ones approximates the inner maximization step in Eq. (1). This scheme is motivated by the fact that grid search, as described in Sec. 3.4.2, is computationally infeasible at training time. For $k = 1$, Adv. Training(100) w/ Wo-k reduces to data augmentation as ubiquitously used in practice. In other words, the worst-of-1 defense is not actually a form of adversarial training as only one random perturbation is sampled to create X_i^* which is no longer dependent on θ^t. However, for notational simplicity, we have decided not to introduce a separate name for this case.

First order. In analogy to common practice for ℓ_p adversarial training, the adversarial example for an image X_i is found via projected gradient descent (PGD), that is by iteratively taking steps in direction of the gradient of the loss $\ell(f_\theta(\mathcal{T}(X_i, \Delta), Y_i)$ with respect to the perturbation Δ. If the solution lies outside of the allowed invariance set $\mathcal{S}$, it is projected back into the set. We consider 5 steps, starting from a random initialization, with step sizes of $[0.03, 0.03, 0.3]$ for the three transformation parameters.

3.4.2. Considered attacks

In order to compute the inner maximization in definition (8) we study the following two attacks.

Grid search. For the grid search attack, the compact perturbation set $\mathcal{S}$ is discretized to find $X_i^\star(\theta) = \mathcal{T}(X_i, \Delta^\star(\theta))$, i.e. the perturbation $\Delta \in \mathcal{S}$ resulting in the largest loss ℓ. This method is computationaly feasible for simple transformations as the parameter space is small. We consider a default grid of 5 values per translation direction and 31 values for rotation, yielding 775 transformed examples that are evaluated for each X_i. We refer to the accuracy attained under this attack as *grid accuracy*. We also performed a finer grid search attack with a total of 7500 transformations on a subset of experiments with results summarized in Table 6. It only leads to minor reductions in accuracy compared to the coarser grid which are smaller for RECOVARIT than for Adv. Training. Due to computational considerations, we report all results in the sequel on the default grid.

First order. For the first-order attack (FO), we create adversarial examples using PGD as described in Sec. 3.4.1.

4. Empirical Results

In this section we summarize insightful observations from our extensive empirical study. All relevant numbers discussed are contained in Tables 1–4. It includes the natural test accuracy (standard accuracy on the test set) and test grid accuracy of standard training, unregularized adversarial training and RECOVARIT with first-order and worst-of-k defenses (for $k \in \{1, 10\}$). Results for the FO attack are reported in Sec. A.3. We compare the methods by reporting absolute and relative error reductions (defined as $\frac{\text{absolute error drop}}{\text{prior error}}$). It is insightful to present both numbers since the absolute values vary drastically between datasets.

For RECOVARIT we report results for the regularization parameter λ which yields the best test grid accuracy. In practice λ is a hyperparameter that needs to be tuned. As shown in Fig. 7, test grid accuracy is relatively robust in a large range of λ values, suggesting that well-performing values of λ are not difficult to find in practice.

We report averages computed over five training runs with identical hyperparameter settings. All plots contain both the mean and the standard error, although the latter is often indiscernible. The runtimes are based on single-GPU training on a node equipped with an NVIDIA GeForce GTX 1080 Ti and two 10-core Xeon E5-2630v4 processors.

Test grid accuracy and runtime The effectiveness of RECOVARIT is clearly visible in Fig. 2 where we plot the test grid accuracy of Adv. Training (50), Adv. Training (100) and RECOVARIT (50) on CIFAR-10 with Wo-k defenses. For the same method, the marked points with increasing runtime correspond to increasing k with $k \in \{1, 5, 10, 20\}$. Even for $k = 20$ the gains in accuracy for Adv. Training methods can barely beat the worst performing regularized variant—RECOVARIT(50) w/ Wo-1—which runs $5\times$ faster. These trends carry over to the other datasets (see Fig. 8). Training with FO defenses was omitted in the fig-

[4] This transformation is implemented using the differentiable spatial transformer block of (Jaderberg et al., 2015).

[5] (Engstrom et al., 2017) additionally analyze a "black canvas" setting where the images are padded with zeros prior to applying the transformation, ensuring that no information is lost due to cropping. Their experiments show that the reduced accuracy of the models cannot be attributed to this effect. Since both versions yield similar results, we report results on the first version, having input images of the same size as the original.

Table 1. Mean accuracies of models trained with Wo-k and FO defenses and evaluated on the natural test set (column "Natural") and against the grid search attack (column "Grid") on the CIFAR datasets. Results on SVHN can be found in Table 3, additional results for CIFAR are contained in Table 4.

	Training \ Evaluation	Natural	Grid
CIFAR-10	Adv. Training(50) w/ Wo-10	93.44%	68.14%
	Adv. Training(100) w/ Wo-10	92.05%	70.35%
	RECOVARIT(50) w/ Wo-10	91.13%	**75.89%**
	Adv. Training(50) w/ FO	92.19%	64.26%
	Adv. Training(100) w/ FO	91.83%	69.74%
	RECOVARIT(50) w/ FO	89.70%	**77.72%**
CIFAR-100	Adv. Training(50) w/ Wo-10	73.03%	35.93%
	Adv. Training(100) w/ Wo-10	68.79%	38.21%
	RECOVARIT(50) w/ Wo-10	68.54%	**49.30%**
	Adv. Training(50) w/ FO	71.11%	33.40%
	Adv. Training(100) w/ FO	68.87%	37.87%
	RECOVARIT(50) w/ FO	68.44%	**52.58%**

Table 2. Mean accuracies of different models trained with Wo-1 defenses (corresponding to data augmentation) and "Standard" training (cf. Section 3.2). The models are evaluated on the natural test set (column "Natural") and against the grid search attack (column "Grid") on the CIFAR-10 and CIFAR-100 datasets.

	Training \ Evaluation	Natural	Grid
SVHN	Standard	95.48%	18.85%
	Adv. Training(50) w/ Wo-1	96.19%	80.43%
	Adv. Training(100) w/ Wo-1	93.97%	82.60%
	RECOVARIT(50) w/ Wo-1	96.19%	**90.48%**
CIFAR-10	Standard	92.11%	9.52%
	Adv. Training(50) w/ Wo-1	92.96%	51.27%
	Adv. Training(100) w/ Wo-1	89.93%	58.29%
	RECOVARIT(50) w/ Wo-1	89.43%	**71.97%**
CIFAR-100	Standard	70.23%	5.09%
	Adv. Training(50) w/ Wo-1	72.38%	23.43%
	Adv. Training(100) w/ Wo-1	66.62%	28.53%
	RECOVARIT(50) w/ Wo-1	66.54%	**45.97%**

ure as they are computationally less efficient with runtimes ranging from 14.8 to 25.8 hours.

From Tables 1–4 we can even make a more general statement: the best RECOVARIT method (RECOVARIT (50) w/ FO) improves the test grid accuracy over the best Adv. Training method (Adv. Training(100) w/ Wo-20) from 72.31% to 77.72% on CIFAR-10 (40.09% to 52.58% on CIFAR-100 and 90.57% to 92.42% on SVHN) corresponding to a relative error reduction of $\sim 20\%$. In fact, the worst overall RECOVARIT method (RECOVARIT (50) w/ Wo-1) is either comparable (SVHN and CIFAR-10) or higher (CIFAR-100) in test grid accuracy as the best of Adv. Training with a runtime gain of 10 hours or more.

Effect of regularizer We now compare the direct effect of adding the conditional variance regularizer. For this purpose we compare Adv. Training (50) with RECOVARIT (50) for fixed defenses. The relative test grid error reductions across different defense mechanisms and datasets are all above 22%. Regarding the trade-off with natural accuracy, we can extract a clear trend from Adv. Training (50) to RECOVARIT (50) for all defenses: as natural accuracy decreases slightly, there is a significant increase in test grid accuracy. More precisely, the absolute gain in test grid accuracy is $3-5\times$ the drop in absolute natural accuracy.

Comparison with standard data augmentation Recall that Adv. Training(100) w/ Wo-1 corresponds to data augmentation with randomly chosen rotations and translations which proves to be a weak defense in terms of test grid accuracy, as can be seen in Table 2. Using RECOVARIT in this setting, we obtain $8, 13, 17\%$ absolute test grid error reduction (and $45, 33, 24\%$ relative error reduction) for SVHN, CIFAR-10 and CIFAR 100, respectively, while the absolute natural test error increases by an absolute percentage of at most 0.5%. On the other hand, Adv. Training(50) w/ Wo-1 yields an improvement in natural accuracy of absolute $\sim 2, 3, 6\%$, sacrificing absolute drops in test grid accuracy of $2, 7, 5\%$, compared to Adv. Training(100) w/ Wo-1. This suggests that even in not purposefully adversarially robust settings, (full batch) standard data augmentation should generally be replaced by a mixed-batch setting—be it to improve natural or adversarial test accuracy. We also note that using data augmentation which only partially covers the transformations used in the attack—as done here for "Standard" training containing data augmentation with translations but without rotations—yields very poor results in terms of test grid accuracy.

ϵ-adversarial robustness. Fig. 10 shows the performance of the Wo-10 defenses in terms of ϵ-adversarial accuracy, as defined in Eq. (9), for CIFAR-100. We see that RECOVARIT not only improves adversarial accuracy for

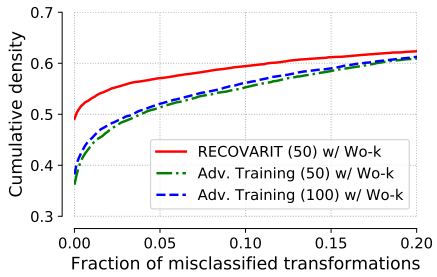


Figure 5. ϵ-adversarial accuracy for worst-of-10 defenses on CIFAR-100. For a range of small ϵ, RECOVARIT improves ϵ-adversarial accuracy compared to the Adv. Training defenses.

$\epsilon = 0$ (corresponding to grid accuracy), but also for a range of small values of ϵ. Fig. 10 and 11 show the equivalent plot for all datasets and $k \in \{1, 10\}$.

There are many more interesting experiments we have conducted for subsets of the defenses and datasets. For example we have compared the train and test grid accuracies, the effect of regularization on a different network architecture, a larger number of adversarial examples per original image M and RECOVARIT(100) w/ Wo-10. A detailed discussion of these experiments can be found in Sec. A.3.

5. Related works

Regularized augmentation using image ID Regularization using the conditional variance term has been previously studied in different contexts independently: for distributional robustness (Heinze-Deml & Meinshausen, 2017), domain generalization (Motiian et al., 2017), ℓ_p adversarial training (Kannan et al., 2018) (adversarial logit pairing (ALP)) and robustness against simple transformations (Cheng et al., 2019).

The first three papers are based on *statically* augmenting the dataset with random independent draws of transformations at the beginning of training. They effectively minimize the regularized loss in Eq. (5), with increased storage requirements by a factor of M. (Cheng et al., 2019) propose the regularizer for increasing natural test accuracy on an object detection setting with $M \in \{3, 35\}$. (Heinze-Deml & Meinshausen, 2017) proved statistical guarantees for the regularized estimator.

ALP, on the other hand, corresponds to *adaptive* augmentation for adversarial accuracy (i.e. adversarial training) in the special case of Eq. (6) with $M = 1$. Depending on the dataset, whether using the original, mixed or fully adversarial batch in the cross entropy loss and the particular choice of attack, ALP has led to no or small improvements in the context of ℓ_∞ robustness in (Engstrom et al., 2018; Mosbach et al., 2018). It remains to be better understood which variants of ALP can improve adversarial robustness in this context.

Robustness against simple transformations A long line of work has focused on inducing equivariant representations to decrease non-adversarial losses, such as the prediction accuracy on the original or randomly transformed test sets such as rotated MNIST (Larochelle et al., 2007).

For neural networks, one possibility besides data augmentation is to modify existing architectures, for example by adding a layer which explicitly constitutes a spatial transformer (Jaderberg et al., 2015), achieving e.g. rotation group equivariance by designing group invariant filters following the spirit of convolutional layers (Marcos et al., 2017; Weiler et al., 2018) or using a combination of static data augmentation as in (Cheng et al., 2019) and network modification (Laptev et al., 2016).

Approaches targeting adversarial accuracy for simple transformations have used attacks and defenses in the spirit of PGD (either on transformation space (Engstrom et al., 2017) or on input space projecting to transformation manifold (Kanbak et al., 2018)) and simple random or grid search (Engstrom et al., 2017; Pei et al., 2017). To the best of our knowledge, the aforementioned networks with a strong inductive bias towards a particular invariance have not been rigorously evaluated in terms of adversarial robustness.

6. Discussion and future work

We have shown that RECOVARIT in combination with first-order, worst-of-k and standard augmentation defenses improves adversarial accuracy against low-dimensional simple transformation attacks by a significant amount on all considered datasets. In particular, RECOVARIT improves the highest reported test grid accuracy to date for CIFAR-10 by an absolute percentage of 7%. Importantly, the regularizer used in RECOVARIT can be computed efficiently.

There are many open questions which naturally arise from our experimental results and will be subject of future work. One immediate theoretical follow-up question concerns the effect of the conditional variance regularizer on the generalization guarantees for adversarial training which have been analyzed in (Sinha et al., 2018). Experimentally, one next step is to perform an even more extensive study including other standard architectures such as Wide ResNet (Zagoruyko & Komodakis, 2016) or ShakeShake (Gastaldi, 2017), and larger datasets such as ImageNet to explore the universality of our conclusions in this work. The second avenue is to combine RECOVARIT with specialized rotation-invariant networks and evaluate how much gain regularization with conditional variance would yield in these architectures when evaluated in terms of adversarial robustness.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Alcorn, M. A., Li, Q., Gong, Z., Wang, C., Mai, L., Ku, W.-S., and Nguyen, A. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. *arXiv preprint arXiv:1811.11553*, 2018.

Antoniou, A., Storkey, A., and Edwards, H. Data augmentation generative adversarial networks. In *Workshop submission for International Conference for Learning Representations*, 2018.

Baird, H. S. Document image defect models. In *Structured Document Image Analysis*, pp. 546–556. Springer, 1992.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. *Robust optimization*, volume 28. Princeton University Press, 2009.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57. IEEE, 2017.

Cheng, G., Han, J., Zhou, P., and Xu, D. Learning rotation-invariant and fisher discriminative convolutional neural networks for object detection. *IEEE Transactions on Image Processing*, 28(1):265–278, 2019.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

Danskin, J. M. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. A rotation and a translation suffice: Fooling cnns with simple transformations. *arXiv preprint arXiv:1712.02779*, 2017.

Engstrom, L., Ilyas, A., and Athalye, A. Evaluating and understanding the robustness of adversarial logit pairing. *arXiv preprint arXiv:1807.10272*, 2018.

Fawzi, A. and Frossard, P. Manitest: Are classifiers really invariant? In *British Machine Vision Conference (BMVC)*, 2015.

Gastaldi, X. Shake-shake regularization. *arXiv preprint arXiv:1705.07485*, 2017.

Geirhos, R., Temme, C. R., Rauber, J., Schütt, H. H., Bethge, M., and Wichmann, F. A. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 7549–7561, 2018.

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

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Heinze-Deml, C. and Meinshausen, N. Conditional variance penalties and domain shift robustness. *arXiv preprint arXiv:1710.11469*, 2017.

Jaderberg, M., Simonyan, K., Zisserman, A., et al. Spatial transformer networks. In *Advances in neural information processing systems*, pp. 2017–2025, 2015.

Kanbak, C., Moosavi-Dezfooli, S.-M., and Frossard, P. Geometric robustness of deep networks: analysis and improvement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4441–4449, 2018.

Kannan, H., Kurakin, A., and Goodfellow, I. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical Report 4, University of Toronto, 2009.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016a.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016b.

Laptev, D., Savinov, N., Buhmann, J. M., and Pollefeys, M. Ti-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 289–297, 2016.

Larochelle, H., Erhan, D., Courville, A., Bergstra, J., and Bengio, Y. An empirical evaluation of deep architectures on problems with many factors of variation. In *Proceedings of the 24th international conference on Machine learning*, pp. 473–480. ACM, 2007.

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

Marcos, D., Volpi, M., Komodakis, N., and Tuia, D. Rotation equivariant vector field networks. In *ICCV*, pp. 5058–5067, 2017.

Mirman, M., Gehr, T., and Vechev, M. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning*, pp. 3575–3583, 2018.

Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.

Mosbach, M., Andriushchenko, M., Trost, T., Hein, M., and Klakow, D. Logit pairing methods can fool gradient-based attacks. *arXiv preprint arXiv:1810.12042*, 2018.

Motiian, S., Piccirilli, M., Adjeroh, D. A., and Doretto, G. Unified deep supervised domain adaptation and generalization. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 2, pp. 3, 2017.

Netzer, Y., , Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, pp. 5, 2011.

Papernot, N., McDaniel, P., and Goodfellow, I. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519. ACM, 2017.

Pei, K., Cao, Y., Yang, J., and Jana, S. Towards practical verification of machine learning: The case of computer vision systems. *arXiv preprint arXiv:1712.01785*, 2017.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. In *International Conference on Learning Representations*, 2018.

Ratner, A. J., Ehrenberg, H., Hussain, Z., Dunnmon, J., and Ré, C. Learning to compose domain-specific transformations for data augmentation. In *Advances in Neural Information Processing Systems*, pp. 3236–3246, 2017.

Samangouei, P., Kabkab, M., and Chellappa, R. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In *International Conference on Learning Representations*, 2018.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Sinha, A., Namkoong, H., and Duchi, J. Certifiable distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Tran, T., Pham, T., Carneiro, G., Palmer, L., and Reid, I. A bayesian data augmentation approach for learning deep models. In *Advances in Neural Information Processing Systems 30*, pp. 2797–2806. Curran Associates, Inc., 2017.

Van Der Vaart, A. W. and Wellner, J. A. Weak convergence. In *Weak convergence and empirical processes*, pp. 16–28. Springer, 1996.

Weiler, M., Hamprecht, F. A., and Storath, M. Learning steerable filters for rotation equivariant cnns. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Wong, E. and Kolter, Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pp. 5283–5292, 2018.

Yaeger, L. S., Lyon, R. F., and Webb, B. J. Effective training of a neural network character classifier for word recognition. In Mozer, M. C., Jordan, M. I., and Petsche, T. (eds.), *Advances in Neural Information Processing Systems 9*, pp. 807–816. MIT Press, 1997.

Zagoruyko, S. and Komodakis, N. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

A. Supplementary material

A.1. Proof of Theorem 1

Before we prove Theorem 1 let us first rigorously state and explain some assumptions and definitions.

Regularity assumption (A4) The regularity condition mentioned in the main statement of the theorem is that the marginal density of Δ (with respect to Lebesgue measure) is in an ϵ-ball around the origin.

Definition of $D(f, B)$: The quantity $D(f, B)$ is defined as follows:

$$D(f, B) := \int_0^1 \sqrt{\log N(\mathcal{F}_\ell, B\epsilon, \|\cdot\|_\infty)} d\epsilon$$

where $\mathcal{F}_\ell := \{g : g(x, y) = \ell(f_\theta(x), y) \mid f_\theta \in \mathcal{F}_{\text{inv}}\}$ and N is the covering number of $\mathcal{F}_\ell$ with respect to $\|\cdot\|_\infty$.

Function and parameter spaces: We define the function space for some fixed $f : \mathbb{R} \to \mathbb{R}$

$$\mathcal{F}_{\text{inv}} = \{f_\theta : f_\theta(x) = f(\theta^\top x) \text{ and } \mathbb{E}\,\text{Var} f_\theta(\mathcal{T}(X, \Delta)) = 0\} = \{f_\theta : f_\theta(x) = f(\theta^\top x) \text{ and } \theta \in \Theta\}$$

The actually searchable space in practice replaces the second conditional variance with its finite sample variant

$$\mathcal{F}_{n,\text{inv}} = \{f_\theta : f_\theta(x) = f(\theta^\top x) \text{ and } \widehat{\mathbb{E}}\widehat{\text{Var}} f_\theta(\mathcal{T}(X, \Delta)) = 0\}$$

where the second equality holds iff $f_\theta(\mathcal{T}(X_i, \Delta_{i,m})) = f_\theta(X)$ for all $i = 1, \ldots, n$ and $m = 1, \ldots, M$. Recall that $\widehat{\mathbb{E}}, \widehat{\text{Var}}$ are the empirical mean and variance.

We now restate the theorem with the more rigorously defined assumption and definition

Theorem 2. *Under the assumptions (A1)-(A4), we have with probability at least $1 - \delta$, that*

$$|\mathcal{L}_{adv}(\widehat{\theta}) - \mathcal{L}_{adv}(\theta^\star)| \leq c_1 \sqrt{\frac{B \log \delta}{n}} + c_2 \sqrt{\frac{B}{n}} D(f, B). \tag{10}$$

This statement implies that the empirical adversarial loss at the constrained estimator is close to the population adversarial loss at the global population optimum $\theta^\star$.

Proof. For simplicity, let us define the population and empirical adversarial loss

$$\mathcal{L}_{\text{adv}}(\theta) = \mathbb{E}_{\mathbb{P}} \max_{\Delta' \in \mathcal{S}} \ell(f_\theta(\mathcal{T}(X, \Delta')), Y) \qquad \widehat{\mathcal{L}}_{\text{adv}}(\theta) = \frac{1}{n} \sum_{i=1}^n \max_{\Delta' \in \mathcal{S}} \ell(f_\theta(\mathcal{T}(X_i, \Delta')), Y_i) \tag{11}$$

and the population and empirical natural loss

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbb{P}} \ell(f_\theta(X), Y) \qquad \widehat{\mathcal{L}}(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(f_\theta(X_i), Y_i)$$

Instead of thinking parameter spaces of parameterized function space, let us reason on the function spaces directly. We define the following population and empirically optimal functions

$$f_{\widehat{\theta}} = \underset{f_\theta \in \mathcal{F}_{n,\text{inv}}}{\arg\min}\, \widehat{\mathcal{L}}_{\text{adv}}(\theta) \qquad f_{\widehat{\theta}^\star} = \underset{f_\theta \in \mathcal{F}_{\text{inv}}}{\arg\min}\, \widehat{\mathcal{L}}_{\text{adv}}(\theta) \qquad f_{\theta^\star} = \underset{f_\theta \in \mathcal{F}_{\text{inv}}}{\arg\min}\, \mathcal{L}_{\text{adv}}(\theta)$$

We now rewrite the left hand side of equation (10)

$$\begin{aligned} \mathcal{L}_{\text{adv}}(\widehat{\theta}) - \widehat{\mathcal{L}}_{\text{adv}}(\theta^\star) = &\underbrace{\mathcal{L}_{\text{adv}}(\widehat{\theta}) - \mathcal{L}_{\text{adv}}(\widehat{\theta^\star})}_{T_0} + \mathcal{L}_{\text{adv}}(\widehat{\theta^\star}) - \widehat{\mathcal{L}}_{\text{adv}}(\widehat{\theta^\star}) \\ &+ \underbrace{\widehat{\mathcal{L}}_{\text{adv}}(\widehat{\theta^\star}) - \widehat{\mathcal{L}}_{\text{adv}}(\theta^\star)}_{T_1} + \widehat{\mathcal{L}}_{\text{adv}}(\theta^\star) - \mathcal{L}_{\text{adv}}(\theta^\star) \end{aligned} \tag{12}$$

The following lemma controls T_0

Lemma A.1. *It holds that* $f_{\widehat{\theta}} = f_{\widehat{\theta}^\star}$.

Proof. Since by assumption and construction we have $P(q < n) = 1$, in Theorem 1 in (Heinze-Deml & Meinshausen, 2017) we have $p_n = 1$. As a consequence $\widehat{\theta} = \widehat{\theta}^\star$ with probability 1. □

Furthermore, observe that for all $f_\theta \in \mathcal{F}_{\text{inv}}$, we have that $\mathcal{L}_{\text{adv}}(\theta) = \mathcal{L}(\theta)$ and $\widehat{\mathcal{L}}_{\text{adv}}(\theta) = \widehat{\mathcal{L}}(\theta)$. Therefore we have

$$f_{\theta^\star} = \operatorname*{arg\,min}_{f_\theta \in \mathcal{F}_{\text{inv}}} \mathcal{L}(\theta) \qquad f_{\widehat{\theta}^\star} = \operatorname*{arg\,min}_{f_\theta \in \mathcal{F}_{\text{inv}}} \widehat{\mathcal{L}}(\theta) \tag{13}$$

Together with the fact that $T_1 \leq 0$ by optimality of $\widehat{\theta}^\star$ and Eq. (12), we thus have

$$\begin{aligned} \mathcal{L}_{\text{adv}}(\widehat{\theta}) - \widehat{\mathcal{L}}_{\text{adv}}(\theta^\star) &\leq |\mathcal{L}_{\text{adv}}(\widehat{\theta}^\star) - \widehat{\mathcal{L}}_{\text{adv}}(\widehat{\theta}^\star)| + |\widehat{\mathcal{L}}_{\text{adv}}(\theta^\star) - \mathcal{L}_{\text{adv}}(\theta^\star)| \\ &\leq \sup_{f_\theta \in \mathcal{F}_{\text{inv}}} |\mathcal{L}(\theta) - \widehat{\mathcal{L}}(\theta)| \end{aligned} \tag{14}$$

where the last inequality holds because $f_{\theta^\star}, f_{\widehat{\theta}^\star} \in \mathcal{F}_{\text{inv}}$.

Finally, we can bound $\sup_{f_\theta \in \mathcal{F}_{\text{inv}}} |\mathcal{L}(\theta) - \widehat{\mathcal{L}}(\theta)|$ using standard empirical process theory by which we obtain

$$P(\sup_{f_\theta \in \mathcal{F}_{\text{inv}}} |\mathcal{L}_{\text{adv}}(\theta) - \widehat{\mathcal{L}}_{\text{adv}}(\theta)| \geq 2\mathcal{R}(\mathcal{F}_{\text{inv}}) + \delta) \leq \mathrm{e}^{\frac{-n\delta^2}{2b^2}}$$

from which it follows that

$$P\left(\sup_{f_\theta \in \mathcal{F}_{\text{inv}}} |\mathcal{L}_{\text{adv}}(\theta) - \widehat{\mathcal{L}}_{\text{adv}}(\theta)| \geq c_1\sqrt{\frac{B\log\delta}{n}} - c_2\sqrt{\frac{B}{n}}D(f,B)\right) \leq \delta$$

where we make use of the assumption that the loss ℓ is uniformly bounded by B and $D(f,B)$ is also known as Dudley's entropy integral (see e.g. (Bartlett & Mendelson, 2002; Van Der Vaart & Wellner, 1996)). This concludes the proof of the theorem. □

A.2. Training procedure

For the gradient step in Algorithm 1 we run a standard minibatch SGD update with a momentum term with parameter 0.9 and weight decay parameter 0.0002. We use an initial learning rate of 0.1 which is divided by 10 after half and three-quarters of the training steps. Independent of the defense method, we fix the number of iterations to 80000 for SVHN and CIFAR-10, and to 120000 for CIFAR-100. Note that this design choice results in different number of epochs when varying the number of adversarial examples M while holding the number of data points entering $\mathcal{L}^t(\theta; B^t)$ constant.

A.3. Additional experimental results

In this section we discuss additional experimental results we collected and analyses we performed.

Computational cost of adding regularizer. We now take a closer look at the runtimes of Adv. Training vs. RECOVARIT with the same Wo-k defense. Fig. 6 shows the runtime of the different worst-of-k defenses for different k. The runtime is mainly driven by the generation of the adversarial examples $X_i^\star(\theta^t)$, explaining the increase with k and the longer runtimes of Adv. Training (100) compared to Adv. Training (50) and RECOVARIT (50). Importantly, we observe that the computation of the regularizer does not increase the runtime noticeably. Moreover, as discussed in Section 4, for the same runtime we always achieve better accuracy.

Trade-off between grid and natural accuracy Across various defenses we compare the trade-off between natural and grid accuracy for mixed and full batch Adv. Training and RECOVARIT. Comparing the best of RECOVARIT with best of Adv. Training in grid accuracy shows the following. For CIFAR-10, at a relative grid error reduction of 20%, we observe a relative natural test error increase of 24%. For the other two datasets the relative grid error reduction ranges between 20% (SVHN) and 21% (CIFAR-100) while the drop in natural accuracy is smaller than two standard errors, suggesting that a gain in grid accuracy is not necessarily coupled with a drop in natural accuracy.

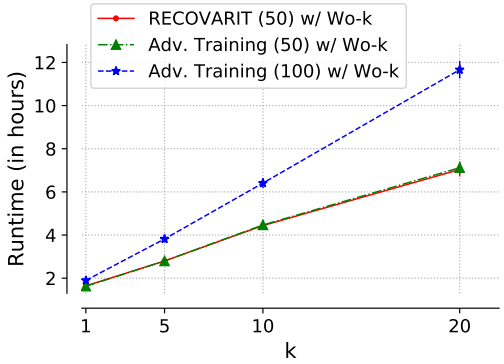


Figure 6. Runtimes on CIFAR-10 for RECOVARIT(50) w/ Wo-k, Adv. Training(50) w/ Wo-k and Adv. Training(100) w/ Wo-k. RECOVARIT does not increase the runtime noticeably.

Table 3. Mean accuracies of different models trained with and without defenses and evaluated on the natural test set (column "Natural") and against the grid search attack (column "Grid") on the SVHN dataset. Standard errors are shown in parentheses.

	Training \ Evaluation	Natural	Grid
SVHN	Standard	95.48% (0.15%)	18.85% (1.27%)
	Adv. Training(50) w/ Wo-1	96.19% (0.06%)	80.43% (0.14%)
	Adv. Training(100) w/ Wo-1	93.97% (0.09%)	82.60% (0.23%)
	RECOVARIT(50) w/ Wo-1 ($\lambda = 3.5$)	96.19% (0.07%)	**90.48% (0.15%)**
	Adv. Training(50) w/ Wo-10	96.56% (0.07%)	88.83% (0.10%)
	Adv. Training(100) w/ Wo-10	95.92% (0.03%)	89.75% (0.17%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 1.5$)	96.41% (0.07%)	**92.17% (0.11%)**
	Adv. Training(50) w/ Wo-20	96.60% (0.06%)	89.72% (0.17%)
	Adv. Training(100) w/ Wo-20	96.29% (0.14%)	90.57% (0.20%)
	RECOVARIT(50) w/ Wo-20 ($\lambda = 1.5$)	96.39% (0.04%)	**92.48% (0.05%)**
	Adv. Training(50) w/ FO	96.27% (0.00%)	84.81% (0.01%)
	Adv. Training(100) w/ FO	96.06% (0.10%)	87.29% (0.09%)
	RECOVARIT(50) w/ FO ($\lambda = 2.5$)	96.30% (0.09%)	**92.42% (0.20%)**

Table 4. Mean accuracies of different models trained with and without defenses and evaluated on the natural test set (column "Natural") and against the grid search attack (column "Grid") on the CIFAR-10 and CIFAR-100 datasets. Standard errors are shown in parentheses.

	Training \ Evaluation	Natural	Grid
CIFAR-10	Standard	92.11% (0.18%)	9.52% (0.66%)
	Adv. Training(50) w/ Wo-1	92.96% (0.17%)	51.27% (0.45%)
	Adv. Training(100) w/ Wo-1	89.93% (0.18%)	58.29% (0.60%)
	RECOVARIT(50) w/ Wo-1 ($\lambda = 3.5$)	89.43% (0.28%)	**71.97% (0.11%)**
	Adv. Training(50) w/ Wo-10	93.44% (0.19%)	68.14% (0.48%)
	Adv. Training(100) w/ Wo-10	92.05% (0.25%)	70.35% (0.17%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 1$)	91.13% (0.13%)	**75.89% (0.23%)**
	Adv. Training(50) w/ Wo-20	93.46% (0.17%)	69.98% (0.22%)
	Adv. Training(100) w/ Wo-20	92.15% (0.28%)	72.31% (0.20%)
	RECOVARIT(50) w/ Wo-20 ($\lambda = 1$)	90.67% (0.12%)	**76.72% (0.21%)**
	Adv. Training(50) w/ FO	92.19% (0.23%)	64.26% (0.25%)
	Adv. Training(100) w/ FO	91.83% (0.19%)	69.74% (0.27%)
	RECOVARIT(50) w/ FO ($\lambda = 3$)	89.70% (0.10%)	**77.72% (0.35%)**
CIFAR-100	Standard	70.23% (0.18%)	5.09% (0.25%)
	Adv. Training(50) w/ Wo-1	72.38% (0.17%)	23.43% (0.47%)
	Adv. Training(100) w/ Wo-1	66.62% (0.37%)	28.53% (0.25%)
	RECOVARIT(50) w/ Wo-1 ($\lambda = 0.75$)	66.54% (0.26%)	**45.97% (0.22%)**
	Adv. Training(50) w/ Wo-10	73.03% (0.13%)	35.93% (0.24%)
	Adv. Training(100) w/ Wo-10	68.79% (0.34%)	38.21% (0.10%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 0.25$)	68.54% (0.27%)	**49.30% (0.33%)**
	Adv. Training(50) w/ Wo-20	73.33% (0.22%)	38.06% (0.19%)
	Adv. Training(100) w/ Wo-20	69.15% (0.49%)	40.09% (0.31%)
	RECOVARIT(50) w/ Wo-20 ($\lambda = 0.25$)	68.04% (0.27%)	**49.98% (0.31%)**
	Adv. Training(50) w/ FO	71.11% (0.37%)	33.40% (0.21%)
	Adv. Training(100) w/ FO	68.87% (0.19%)	37.87% (0.12%)
	RECOVARIT(50) w/ FO ($\lambda = 0.5$)	68.44% (0.39%)	**52.58% (0.20%)**

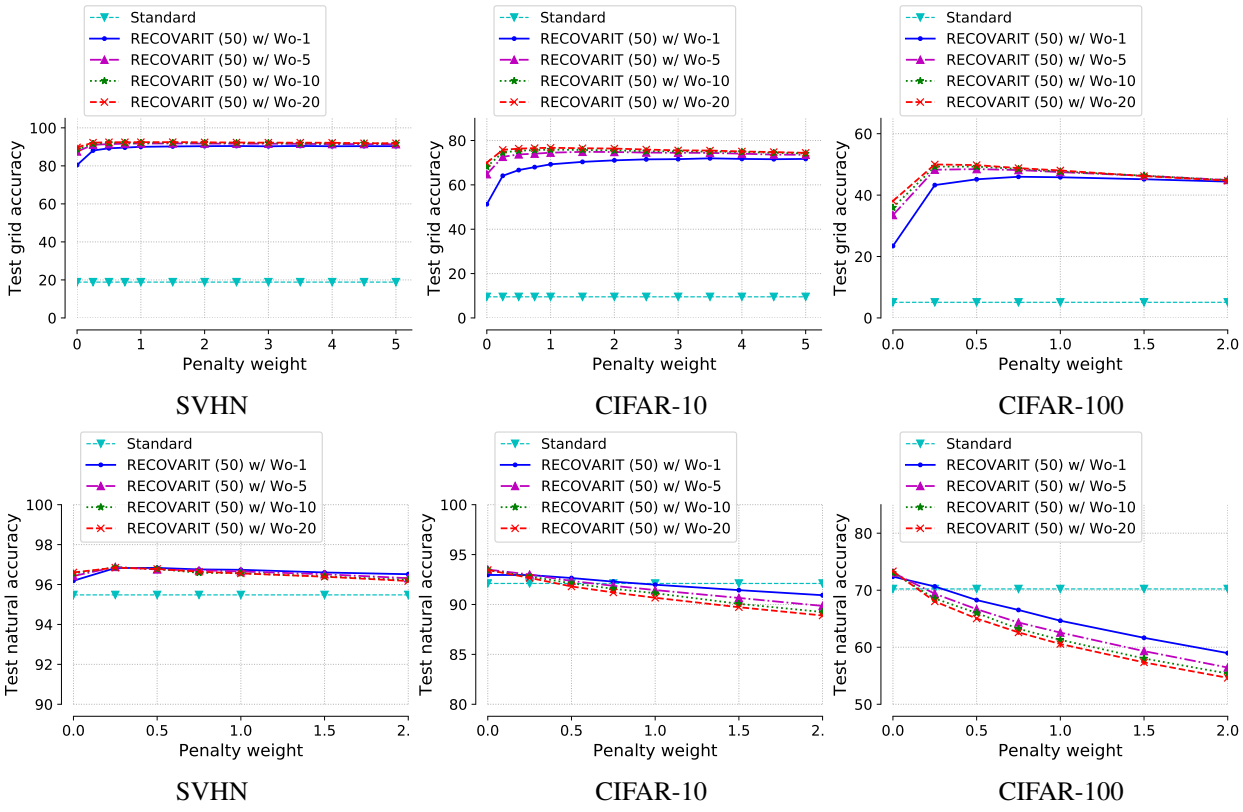


Figure 7. Test grid accuracy (first row) and test natural accuracy (second row) as a function of the regularization parameter λ for the SVHN (first column), CIFAR-10 (second column) and CIFAR-100 (third column) datasets. The test grid accuracy is relatively robust in across a large range of λ values while natural test accuracy decreases with larger values of λ.

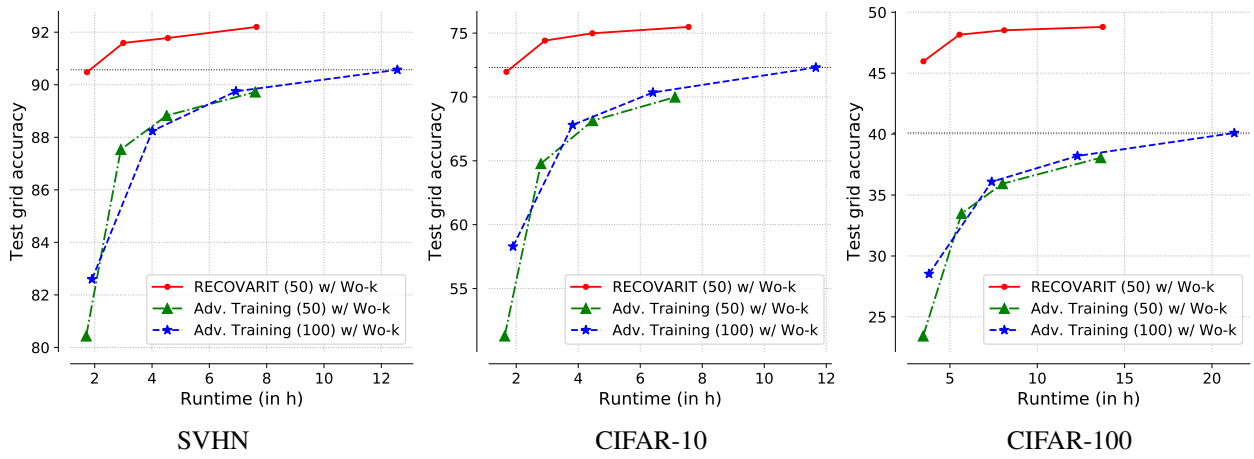


Figure 8. Test grid accuracy as a function of runtime on the SVHN, CIFAR-10 and CIFAR-100 datasets. RECOVARIT(50) w/ Wo-k achieves comparable or higher test grid accuracy while requiring less runtime. For the same method, the marked points with increasing runtime correspond to increasing k with $k \in \{1, 5, 10, 20\}$. Even for $k = 20$ the gains in accuracy for Adv. Training methods can barely beat the worst performing regularized variant—RECOVARIT(50) w/ Wo-1—which runs $5\times$ faster.

Weakness of first order attack. Table 5 shows the accuracies of various models trained with FO defenses and evaluated against the FO and the grid search attack on all datasets. We observe that the FO attack constitutes are very weak attack since the associated accuracies are much larger than for the grid search attack. In other words, the FO attack only yields a very loose upper bound on the adversarial accuracy. This stands in stark contrast to ℓ_∞ attacks and has first been noted and discussed in (Engstrom et al., 2017). Interestingly, using the first order method as a *defense* mechanism proves to be very effective in terms of grid accuracy. When used in combination with RECOVARIT this defense yields the largest overall accuracies as shown and discussed in Section 4. Also see Tables 3 and 4 for an overview of the results.

Generalization gap for grid accuracy. Table 7 shows the train and test grid accuracy for various models trained with the worst-of-10 defense. We observe that the gap between these two quantities is smaller for RECOVARIT than for Adv. Training. This suggests that RECOVARIT indeed helps to steer the estimate closer towards to the population parameter of interest $\theta^\star$.

Grid and natural accuracy of worst-of-k defenses as a function of k Comparing mixed and full batch Adv. Training and RECOVARIT for the worst-of-k defenses, Fig. 9 shows the test grid accuracy as a function of k. The mean test grid accuracy for standard training is shown as a horizontal line. Standard training achieves a test grid accuracy that lies well below the performance of the considered defenses. Comparing the defenses shows that RECOVARIT improves the test grid accuracy for all values of k and all datasets. We observe a small trade-off between increased test grid accuracy and test natural accuracy.

VGG architecture. In Table 10 we report the performance using a VGG architecture instead of a Resnet-32 for SVHN and CIFAR-10. Again, RECOVARIT outperforms Adv. Training for all values of k. The absolute performance of the VGG architecture is slightly larger than for the Resnet-32 and the differences between the various defenses are smaller. Also the test grid performance of the standard model is much larger than for the Resnet-32. This raises the interesting question what architectural property this increase in robustness can be attributed to.

Spatial transformer. As mentioned in Section 1, one approach to achieve the desired invariance with respect to simple transformations is to use a specialized network architecture. As one instance of this class of models, we ran experiments with the spatial transformer (Jaderberg et al., 2015). However, we could not find a hyperparameter setting for which the model showed a reasonable performance with respect to test grid accuracy. Therefore, we omit the results in this version of this work.

RECOVARIT (100) In Table 8, we show results for RECOVARIT(100) w/ Wo-10 compared to Adv. Training(50) w/ Wo-10, RECOVARIT(50) w/ Wo-10 and Adv. Training(100) w/ Wo-10. RECOVARIT(100) w/ Wo-10 improves upon RECOVARIT(50) w/ Wo-10 by an absolute percentage of 0.91%.

Since this small improvement comes at the cost of larger runtimes, we focused on RECOVARIT(50) w/ Wo-10 for the full set of experiments.

Larger number of adversarial examples per original image ($M = 3$). In Table 9, we explore using $M = 3$ adversarial examples per original image for two different batch sizes $b \in \{128, 256\}$. We denote these variants as Adv. Training(75) and RECOVARIT(75). For $b = 128$, $M = 3$ yields improvements compared to $M = 1$ of 1.48% for Adv. Training and no improvements for RECOVARIT. For $b = 256$, $M = 3$ yields improvements compared to $M = 1$ of 2.09% for Adv. Training and of 0.57% for RECOVARIT.

Table 5. Mean accuracies of different models trained with FO defenses and evaluated on the natural test set (column "Natural"), against the FO attack (column "FO") and against the grid search attack (column "Grid") on the SVHN, CIFAR-10 and CIFAR-100 datasets. While the test accuracy for the FO attack is only slightly lower than the natural accuracy in most cases, the grid accuracy is significantly smaller. In other words, the FO attack only yields a loose upper bound on adversarial accuracy while it is a competitive defense in terms of test grid accuracy when used together with RECOVARIT.

	Training \ Evaluation	Natural	FO	Grid
SVHN	Adv. Training(50) w/ FO	96.27% (0.00%)	95.26% (0.04%)	84.81% (0.01%)
	Adv. Training(100) w/ FO	96.06% (0.10%)	95.46% (0.10%)	87.29% (0.09%)
	RECOVARIT(50) w/ FO ($\lambda = 2.5$)	96.30% (0.09%)	95.92% (0.13%)	**92.42% (0.20%)**
CIFAR-10	Adv. Training(50) w/ FO	92.19% (0.23%)	88.84% (0.27%)	64.26% (0.25%)
	Adv. Training(100) w/ FO	91.83% (0.19%)	89.87% (0.10%)	69.74% (0.27%)
	RECOVARIT(50) w/ FO ($\lambda = 3$)	89.70% (0.10%)	88.15% (0.21%)	**77.72% (0.35%)**
CIFAR-100	Adv. Training(50) w/ FO	71.11% (0.37%)	65.01% (0.32%)	33.40% (0.21%)
	Adv. Training(100) w/ FO	68.87% (0.19%)	65.56% (0.12%)	37.87% (0.12%)
	RECOVARIT(50) w/ FO ($\lambda = 0.5$)	68.44% (0.39%)	66.04% (0.40%)	**52.58% (0.20%)**

Table 6. Mean accuracies for different models evaluated against two different grid search attacks. The column "Grid (775)" shows the test grid accuracy using the default grid containing 5 values per translation direction and 31 values for rotation, yielding a total of 775 transformed examples that are evaluated for each X_i. The column "Grid (7500)" shows the test grid accuracy on a much finer grid with 10 values per translation direction and 75 values for rotation, resulting 7500 transformed examples. We observe that the test grid accuracy only decreases slightly for the finer grid and the reduction in accuracy is smaller for RECOVARIT than for Adv. Training. Due to computational reasons we use the grid containing 775 values for all other experiments.

	Training \ Evaluation	Grid (775)	Grid (7500)
SVHN	Adv. Training(50) w/ Wo-10	88.83% (0.10%)	88.02% (0.12%)
	Adv. Training(100) w/ Wo-10	89.75% (0.17%)	89.29% (0.15%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 1.5$)	92.17% (0.11%)	91.79% (0.12%)
CIFAR-10	Adv. Training(50) w/ Wo-10	68.14% (0.48%)	65.69% (0.28%)
	Adv. Training(100) w/ Wo-10	70.35% (0.16%)	68.28%(0.16%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 1$)	75.89% (0.23%)	74.58% (0.16%)
CIFAR-100	Adv. Training(50) w/ Wo-10	35.93% (0.24%)	33.62% (0.23%)
	Adv. Training(100) w/ Wo-10	38.21% (0.10%)	36.04% (0.21%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 0.25$)	49.30% (0.33%)	47.95% (0.23%)

Table 7. Generalization gap in terms of grid accuracy. The column "Train Grid" reports the train grid accuracy; the column "Test Grid" reports the test grid accuracy. We observe that the gap between these two quantities is smaller for RECOVARIT than for Adv. Training. This suggests that RECOVARIT indeed helps to steer the estimate towards $\theta^\star$.

	Training \ Evaluation	Train Grid	Test Grid
SVHN	Adv. Training(50) w/ Wo-10	90.71% (0.07%)	88.83% (0.10%)
	Adv. Training(100) w/ Wo-10	93.09% (0.08%)	89.75% (0.17%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 1.5$)	92.67% (0.08%)	92.17% (0.11%)
CIFAR-10	Adv. Training(50) w/ Wo-10	77.19% (0.32%)	68.14% (0.48%)
	Adv. Training(100) w/ Wo-10	85.60% (0.09%)	70.35% (0.16%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 0.25$)	83.83% (0.24%)	74.60% (0.23%)
CIFAR-100	Adv. Training(50) w/ Wo-10	54.15% (0.40%)	35.93% (0.24%)
	Adv. Training(100) w/ Wo-10	72.87% (0.27%)	38.21% (0.10%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 0.25$)	63.97% (0.06%)	49.30% (0.33%)

Table 8. Accuracies for Adv. Training and RECOVARIT trained by Wo-10 defenses. In addition to the results reported in the main text, we report the performance of RECOVARIT (100) here.

	Training \ Evaluation	Natural	Grid
CIFAR-10	Adv. Training(50) w/ Wo-10	93.44% (0.19%)	68.14% (0.48%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 1$)	91.13% (0.13%)	75.89% (0.23%)
	Adv. Training(100) w/ Wo-10	92.05% (0.25%)	70.35% (0.17%)
	RECOVARIT(100) w/ Wo-10 ($\lambda = 1.5$)	90.85% (0.14%)	76.80% (0.20%)

Table 9. Accuracies for Adv. Training and RECOVARIT trained by Wo-10 defenses. We compare two batch sizes ($b = 128$ vs. $b = 256$) and two different numbers of adversarial examples M per original image ($M = 1$ and $M = 3$).

		Training \ Evaluation	Natural	Grid
CIFAR-10	$b = 128$	Adv. Training(50) w/ Wo-10	93.44% (0.19%)	68.14% (0.48%)
		RECOVARIT(50) w/ Wo-10	91.13% (0.13%)	75.89% (0.23%)
		Adv. Training(75) w/ Wo-10	93.00% (0.15%)	69.62% (0.39%)
		RECOVARIT(75) w/ Wo-10 ($\lambda = 0.5$)	90.79% (0.15%)	75.39% (0.08%)
	$b = 256$	Adv. Training(50) w/ Wo-10	93.26% (0.24%)	68.18% (0.37%)
		RECOVARIT(50) w/ Wo-10 ($\lambda = 2$)	90.80% (0.19%)	76.91% 0.16%)
		Adv. Training(75) w/ Wo-10	93.43% (0.10%)	70.27% (0.13%)
		RECOVARIT(75) w/ Wo-10 ($\lambda = 1.5$)	90.24% (0.12%)	77.48% (0.13%)

Table 10. Mean accuracies of different models trained with a VGG architecture with and without defenses. RECOVARIT outperforms Adv. Training for all values of k. The absolute performance of the VGG architecture is larger than for the Resnet-32 and the differences between the various defenses are smaller. Standard errors are shown in parentheses.

	Training \ Evaluation	Natural	Grid
SVHN	Standard	94.14% (0.20%)	23.15% (0.60%)
	Adv. Training(50) w/ Wo-1	95.20% (0.11%)	78.64% (0.41%)
	Adv. Training(100) w/ Wo-1	91.94% (0.28%)	83.15% (0.39%)
	RECOVARIT(50) w/ Wo-1 ($\lambda = 1$)	95.90% (0.12%)	**86.78% (0.45%)**
	Adv. Training(50) w/ Wo-10	95.02% (0.24%)	87.37% (0.60%)
	Adv. Training(100) w/ Wo-10	93.29% (0.24%)	89.00% (0.38%)
	RECOVARIT(50) w/ Wo-10 ($\lambda = 0.5$)	96.12% (0.08%)	**90.73% (0.20%)**
CIFAR-10	Standard	92.15% (0.24%)	20.55% (0.54%)
	Adv. Training(50) w/ Wo-1	92.96% (0.19%)	64.07% (0.25%)
	Adv. Training(100) w/ Wo-1	91.49% (0.12%)	70.13% (0.17%)
	RECOVARIT(50) w/ Wo-1 ($\lambda = 1.5$)	91.57% (0.10%)	**72.22% (0.37%)**
	Adv. Training(50) w/ Wo-10	93.11% (0.20%)	74.16% (0.17%)
	Adv. Training(100) w/ Wo-10	92.17% (0.13%)	76.30% (0.24%)
	RECOVARIT(50) w/ Wo-10($\lambda = 0.25$)	93.04% (0.10%)	**78.05% (0.37%)**
	Adv. Training(50) w/ FO	91.34% (0.14%)	70.44% (0.62%)
	Adv. Training(100) w/ FO	90.91% (0.13%)	75.24% (0.41%)
	RECOVARIT(50) w/ FO($\lambda = 1.5$)	90.88% (0.18%)	**77.21% (0.30%)**

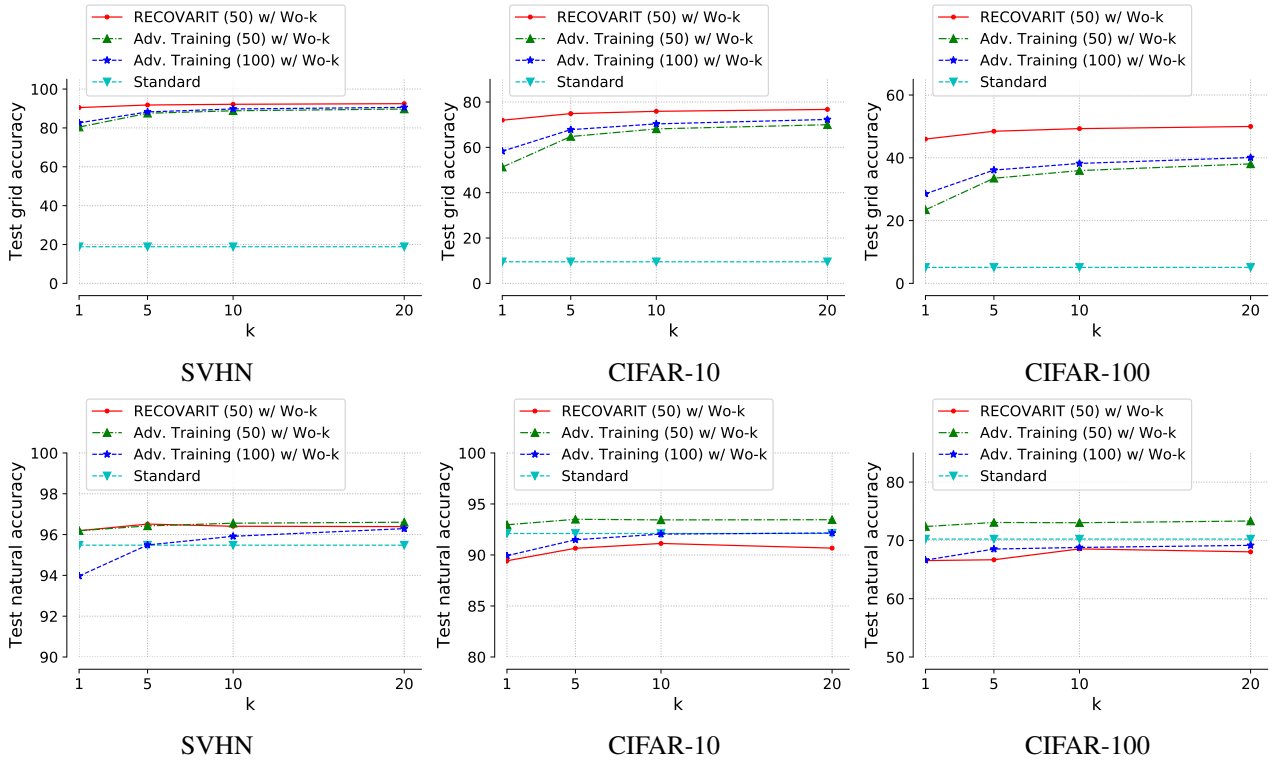


Figure 9. Test grid accuracy (first row) and natural test accuracy (second row) for worst-of-k defenses with different values of k for the SVHN (first column), CIFAR-10 (second column) and CIFAR-100 (third column) datasets. RECOVARIT outperforms the other defenses for all k. As k increases the accuracy gain in test grid accuracy diminishes for all methods. There is a small trade-off in terms of natural test accuracy.

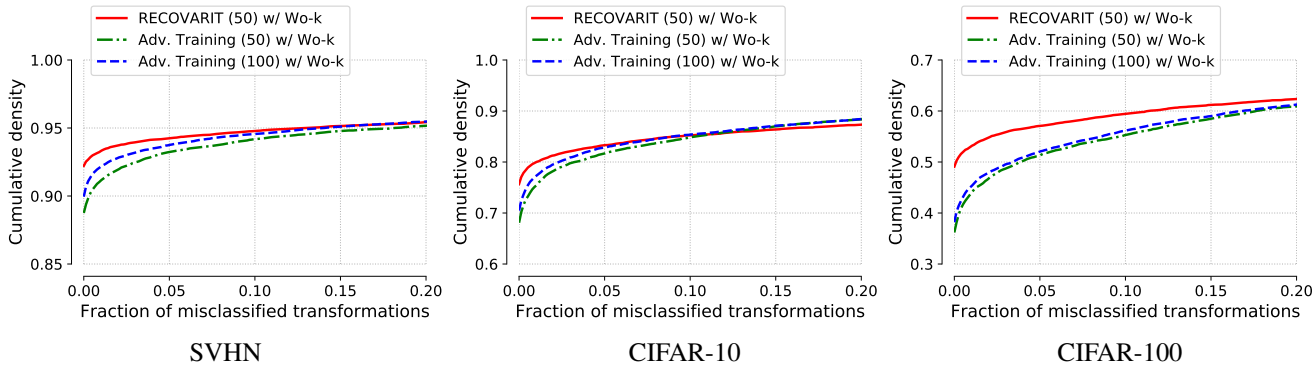


Figure 10. ϵ-adversarial accuracy for worst-of-10 defenses. We show the CDFs for the fraction ϵ of misclassified transformations. For a range of small values of ϵ, RECOVARIT improves ϵ-adversarial accuracy compared to the Adv. Training defenses.

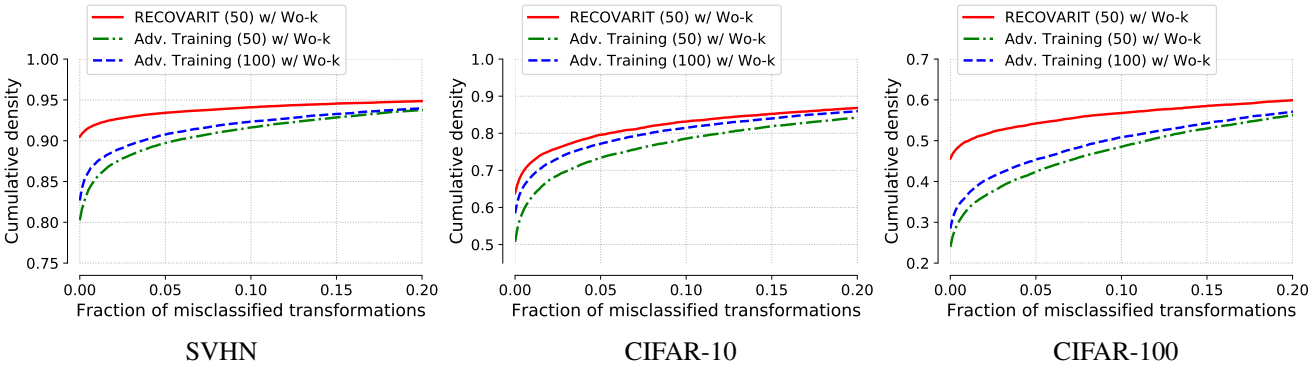


Figure 11. ϵ-adversarial accuracy for worst-of-1 defenses. We show the CDFs for the fraction ϵ of misclassified transformations. For a range of small values of ϵ, RECOVARIT improves ϵ-adversarial accuracy compared to the Adv. Training defenses.

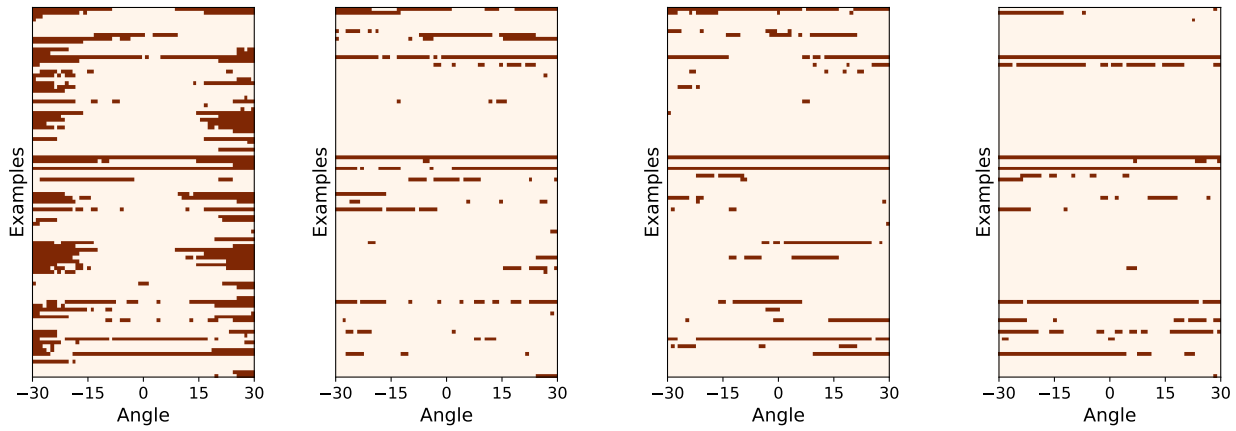


Figure 12. For 100 randomly chosen examples from the CIFAR-10 dataset, we show which rotations lead to a misclassification by various models trained with and without the Wo-1 defense (corresponding to data augmentation). Each row corresponds to one example and each column to one angle. A dark red square indicates that the corresponding example was misclassified after being rotated by the corresponding angle. The visualization for Adv. Training(50) w/ Wo-1 is more fragmented than for Adv. Training(100) w/ Wo-1 and RECOVARIT(50) w/ Wo-1.

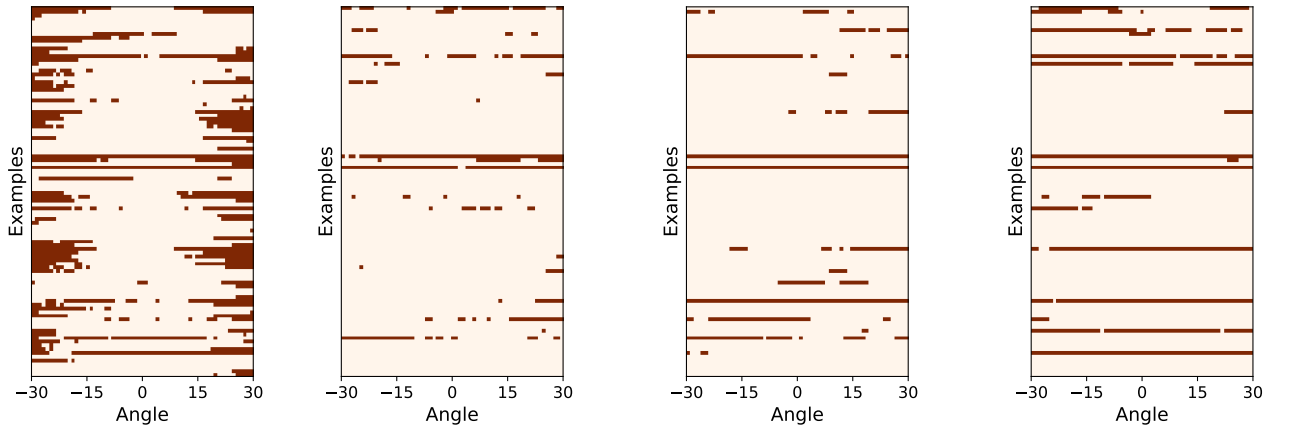


Figure 13. For 100 randomly chosen examples from the CIFAR-10 dataset, we show which rotations lead to a misclassification by various models trained with and without the Wo-10 defense. Each row corresponds to one example and each column to one angle. A dark red square indicates that the corresponding example was misclassified after being rotated by the corresponding angle. The visualization for Adv. Training(50) w/ Wo-10 is more fragmented than for Adv. Training(100) w/ Wo-10 and RECOVARIT(50) w/ Wo-10.