Supplementary Material for Disentangling Adversarial Robustness and Generalization

David Stutz[1] Matthias Hein[2] Bernt Schiele[1]
[1]Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken
[3]University of Tübingen, Tübingen
{david.stutz,schiele}@mpi-inf.mpg.de, matthias.hein@uni-tuebingen.de

A. Overview

In the main paper, we study the relationship between adversarial robustness and generalization. Based on the distinction between regular and on-manifold adversarial examples, we show that 1. regular adversarial examples leave the underlying manifold of the data; 2. on-manifold adversarial examples exist; 3. on-manifold robustness is essentially generalization; 4. and regular robustness is independent of generalization. For clarity and brevity, the main paper focuses on the L_∞ attack by Madry et al. [16] and the corresponding adversarial training variant applied to simple convolutional neural networks. For on-manifold adversarial examples, we approximate the manifold using class-specific VAE-GANs [13, 18]. In this document, we present comprehensive experiments demonstrating that our findings generalize across attacks, adversarial training variants, network architectures and to class-agnostic VAE-GANs.

A.1. Contents

In Section B, we present additional details regarding our experimental setup, corresponding to Section 3.1 of the main paper: in Section B.1, we discuss details of our synthetic FONTS datasets and, in Section B.2, we discuss our VAE-GAN implementation. Then, in Section C we extend the discussion of Section 3.2 with further results demonstrating that adversarial examples leave the manifold. Subsequently, in Section D, we show and discuss additional on-manifold adversarial examples to supplement the examples shown in Fig. 2 of the main paper. Then, complementing the discussion in Sections 3.4 and 3.5, we consider additional attacks, network architectures and class-agnostic VAE-GANs. Specifically, in Section E, we consider the L_2 variant of the white-box attack by Madry et al. [16], the L_2 white-box attack by Carlini and Wagner [2], and black-box transfer attacks. In Section F, we present experiments on multi-layer perceptrons and, in Section G, we consider approximating the manifold using class-agnostic VAE-GANs. In Section H, corresponding to Section 3.6, we consider different variants of regular and on-manifold adversarial training. Finally, in Section I, we discuss our definition of adversarial examples in the context of related work by Tsipras et al. [21], as outlined in Section 3.5.

B. Experimental Setup

We provide technical details on the introduced synthetic FONTS dataset, Section B.1, and our VAE-GAN implementation, Section B.2.

B.1. FONTS Dataset

Our FONTS dataset consists of randomly rotated characters "A" to "J" from different fonts, as outlined in Section 3.1 of the main paper. Specifically, we consider 1000 Google Fonts as downloaded from the corresponding GitHub repository[1]. We manually exclude fonts based on symbols, or fonts that could not be rendered correctly in order to obtain a cleaned dataset consisting of clearly readable letters "A" to "J"; still, the 1000 fonts exhibit significant variance. The obtained, rendered letters are transformed using translation, shear, scaling and rotation: for each letter and font, we create 112 transformations, uniformly sampled in $[-0.2, 0.2]$, $[-0.5, 0.5]$, $[0.75, 1.15]$, and $[-\pi/2, \pi/2]$, respectively. As a result, with 1000 fonts and 10 classes, we obtain 1.12Mio images of size 28×28, splitted into 960k training images and 160k test images (of which we use 40k in the main paper); thus, the dataset has four times the size of EMNIST [3]. For simplicity, the transformations are applied using a spatial transformer network [9] by assembling translation $[t_1, t_2]$, shear $[\lambda_1, \lambda_2]$, scale s and rotation r into an affine transformation matrix,

$$\begin{bmatrix} \cos(r)s - \sin(r)s\lambda_1 & -\sin(r)s + \cos(r)s\lambda_1 & t_1 \\ \cos(r)s\lambda_2 + \sin(r)s & -\sin(r)s\lambda_2 + \cos(r)s & t_2 \end{bmatrix}, \quad (1)$$

making the generation process fully differentiable. Overall, FONTS offers full control over the manifold, i.e., the transformation parameters, font and class, with differentiable generative model, i.e., decoder.

[1] https://github.com/google/fonts

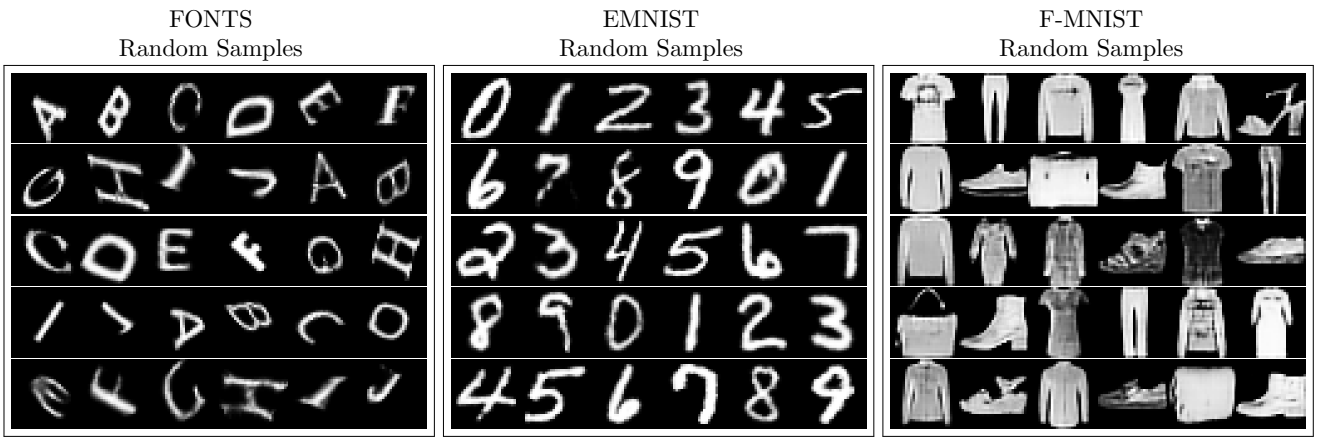

Figure 10: For FONTS (left), EMNIST (middle) and F-MNIST (right), we show random samples from the learned, class-specific VAE-GANs used to craft on-manifold adversarial examples. Our VAE-GANs generate realistic looking samples; although we also include problematic samples illustrating the discrepancy between true and approximated data distribution.

B.2. VAE-GAN Variant

As briefly outlined in Section 3.1 of the main paper, we use class-specific VAE-GANs [13, 18] to approximate the class-manifolds on all datasets, i.e., FONTS, EMNIST [3], F-MNIST [22] and CelebA [15]. In contrast to [13], however, we use a reconstruction loss on the image, not on the discriminator's features; in contrast to [18], we use the standard Kullback-Leibler divergence to regularize the latent space. The model consists of an encoder enc, approximating the posterior $q(z|x) \approx p(z|x)$ of latent code z given image x, a (deterministic) decoder dec, and a discriminator dis. During training, the sum of the following losses is minimized:

$$\mathcal{L}_{\text{enc}} = \mathbb{E}_{q(z|x)}\left[\lambda \|x - \mathrm{dec}(z)\|_1\right] + \mathrm{KL}(q(z|x)|p(z)) \quad (2)$$

$$\mathcal{L}_{\text{dec}} = \mathbb{E}_{q(z|x)}\left[\lambda \|x - \mathrm{dec}(z)\|_1 - \log(\mathrm{dis}(\mathrm{dec}(z)))\right] \quad (3)$$

$$\begin{aligned}\mathcal{L}_{\text{dis}} = &- \mathbb{E}_{p(x)}\left[\log(\mathrm{dis}(x))\right] \\ &- \mathbb{E}_{q(z|x)}\left[\log(1 - \mathrm{dis}(\mathrm{dec}(z)))\right]\end{aligned} \quad (4)$$

using a standard Gaussian prior $p(z)$. Here, $q(z|x)$ is modeled by predicting the mean $\mu(x)$ and variance $\sigma^2(x)$ such that $q(z|x) = \mathcal{N}(z; \mu(x), \mathrm{diag}(\sigma^2(x)))$ and the weighting parameter λ controls the importance of the L_1 reconstruction loss relative to the Kullback-Leibler divergence KL and the adversarial loss for decoder and discriminator. As in [12], we use the reparameterization trick with one sample to approximate the expectations in Eq. (2), (3) and (4), and the Kullback-Leibler divergence $\mathrm{KL}(q(z|x)|p(z))$ is computed analytically.

The encoder, decoder and discriminator consist of three (four for CelebA) (de-) convolutional layers (4×4 kernels; stride 2; 64, 128, 256 channels), followed by ReLU activations and batch normalization [8]; the encoder uses two fully connected layers to predict mean and variance; the discriminator uses two fully connected layers to predict logits. We tuned λ to dataset- and class-specific values: on FONTS, $\lambda = 3$ worked well for all classes, on EMNIST, $\lambda = 2.5$ except for classes "0" ($\lambda = 2.75$), "1" ($\lambda = 5.6$) and "8" ($\lambda = 2.25$), on F-MNIST, $\lambda = 2.75$ worked well for all classes, on CelebA $\lambda = 3$ worked well for both classes. Finally, we trained our VAE-GANs using ADAM [11] with learning rate 0.005 (decayed by 0.9 every epoch), weight decay 0.0001 and batch size 100 for 10, 30, 60 and 30 epochs on FONTS, EMNIST, F-MNIST and CelebA, respectively. We also consider class-agnostic VAE-GANs trained using the same strategy with $\lambda = 3$ for FONTS, $\lambda = 3$ on EMNIST, $\lambda = 2.75$ on F-MNIST and $\lambda = 3$ on CelebA, see Section G for results.

In Fig. 10, we include random samples of the class-specific VAE-GANs. Especially on EMNIST and FONTS, our VAE-GANs generate realistic looking samples with sharp edges. However, we also show several problematic random samples, illustrating the discrepancy between the true data distribution and the approximation – as particularly highlighted on FONTS.

C. Adversarial Example Distance to Manifold

Complementing Section 3.2 of the main paper, we provide additional details and results regarding the distance of regular adversarial examples to the true or approximated manifold, including a theoretical argument of adversarial examples leaving the manifold.

On FONTS, with access to the true manifold in form of a perfect decoder dec, we iteratively obtain the latent code $\tilde{z}$ yielding the manifold's closest image to the given adversarial example $\tilde{x}$ as

$$\tilde{z} = \underset{z}{\operatorname{argmin}} \ \|\tilde{x} - \mathrm{dec}(z)\|_2^2. \quad (5)$$

We use 100 iterations of ADAM [10], with a learning rate of 0.09, decayed every 10 iterations by a factor 0.95. We found that additional iterations did not improve the results. The

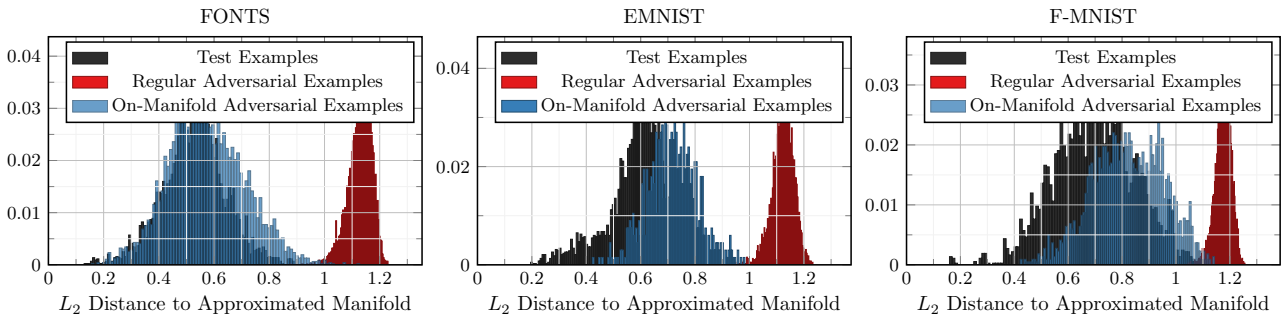

Figure 11: On FONTS (left), EMNIST (middle) and F-MNIST (right) we plot the distance of adversarial examples to the approximated manifold. We show normalized histograms of the L_2 distance of adversarial examples to their projection, as described in the text. Regular adversarial examples exhibit a significant distance to the manifold; clearly distinguishable from on-manifold adversarial examples and test images. We also note that, depending on the VAE-GAN approximation, on-manifold adversarial examples are hardly distinguishable from test images.

obtained projection $\pi(\tilde{x}) = \text{dec}(\tilde{z})$ is usually very close to the original test image x for which the adversarial example was crafted. The distance is then computed as $\|\tilde{x} - \pi(\tilde{x})\|_2$; we refer to the main paper for results and discussion.

If the true manifold is not available, we locally approximate the manifold using 50 nearest neighbors $x_1, \ldots, x_{50}$ of the adversarial example $\tilde{x}$. In the main paper, we center these nearest neighbors at the test image x, i.e., consider the sub-space spanned by $x_i - x$. Here, we show that the results can be confirmed when centering the nearest neighbors at their mean $\bar{x} = 1/50 \sum_{i=1}^{50} x_i$ and considering the subspace spanned by $x_i - \bar{x}$ instead. In this scenario, the test image x is not necessarily part of the approximated manifold anymore. The projection onto this sub-space can be obtained by solving the least squares problem; specifically, we consider the vector $\delta = \tilde{x} - x$, i.e., we assume that the "adversarial direction" originates at the mean $\bar{x}$. Then, we solve

$$\beta^* = \underset{\beta}{\text{argmin}} \ \|X\beta - \delta\|_2^2 \tag{6}$$

where the columns X_i are the vectors $x_i - \bar{x}$. The projection $\pi(\tilde{x})$ is obtained as $\pi(\tilde{x}) = X\beta^*$; the same approach can be applied to projecting the test image x. Note that it is crucial to consider the adversarial direction δ itself, instead of the adversarial example $\tilde{x}$ because $\|\delta\|_2$ is small by construction, i.e., the projections of $\tilde{x}$ and x are very close. In Fig. 11, we show results using this approximation on FONTS, EMNIST and F-MNIST. Regular adversarial examples can clearly be distinguished from test images and on-manifold adversarial examples. Note, however, that we assume access to both the test image x and the corresponding adversarial example $\tilde{x}$ such that this finding cannot be exploited for detection. We also notice that the discrepancy between the distance distributions of test images and on-manifold adversarial examples reflects the approximation quality of the used VAE-GANs.

C.1. Intuition and Theoretical Argument

Having empirically shown that regular adversarial examples tend to leave the manifold, often in a nearly orthogonal direction, we also discuss a theoretical argument supporting this observation. The main assumption is that the training loss is constant on the manifold (normally close to zero) due to training and proper generalization, i.e., low training and test loss. Thus, the loss gradient is approximately orthogonal to the manifold as this is the direction to increase the loss most efficiently.

More formally, let $f(x)$ denote the classifier which – for simplicity – takes inputs $x \in \mathbb{R}^d$ and predicts outputs $y \in \mathbb{R}^K$ for K classes. We assume both the classifier as well as the used loss, e.g., cross-entropy loss, to be differentiable. We further expect the data to lie on a manifold $\mathcal{M}$ and the loss to be constant on $\mathcal{M} \cap B(x, \epsilon)$ with

$$B(x, \epsilon) = \{x' \in \mathbb{R}^d : \|x' - x\| \leq \epsilon\}. \tag{7}$$

Let

$$g(x) = \mathbb{E}\left[\mathcal{L}(f(x), y)|x\right] \tag{8}$$

be the conditional expectation of the loss $\mathcal{L}$; then, by the mean value theorem, there exists $\theta(x') \in [0, 1]$ for each $x' \in \mathcal{M} \cap B(x, \epsilon)$ such that

$$0 = g(x') - g(x) \tag{9}$$
$$= \langle \nabla g(\theta(x')x + (1 - \theta(x'))x'), x' - x \rangle \tag{10}$$

As this holds for all $\epsilon > 0$ and as $\epsilon \to 0$, every vector $x' - x$ becomes a tangent of $\mathcal{M}$ at x and

$$\lim_{\epsilon \to 0} \nabla g(\theta(x')x + (1 - \theta(x'))x') = \nabla g(x), \tag{11}$$

it holds that $\nabla g(x)$ is orthogonal to the tangent space of $\mathcal{M}$ at x. As $\nabla g(x)$ is the gradient of the expected loss, it implies that adversarial examples, as computed, e.g., using first-order gradient-based approaches such as Eq. (12), leave the manifold $\mathcal{M}$ in order to fool the classifier $f(x)$.

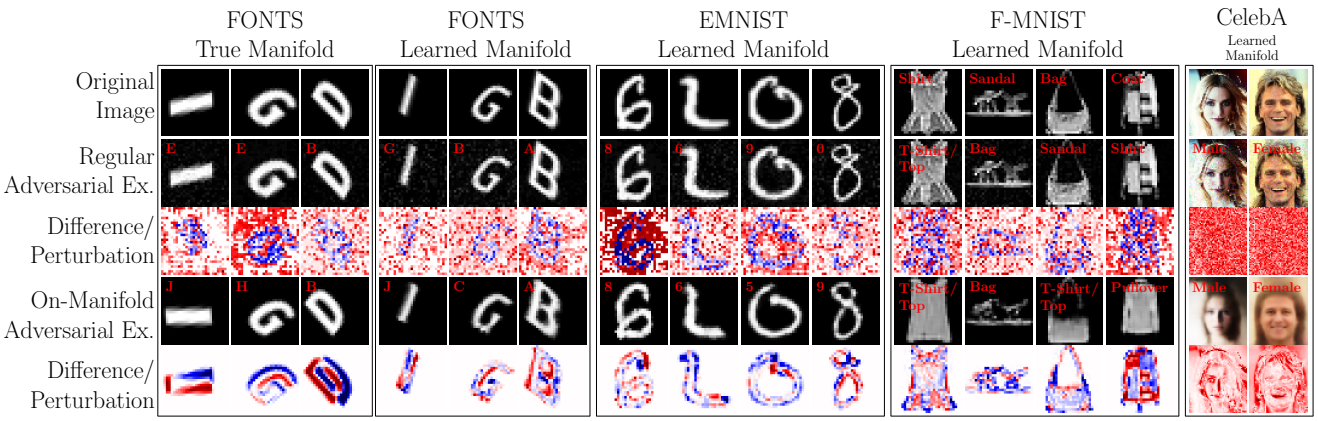

Figure 12: Regular and on-manifold adversarial examples on FONTS, EMNIST, F-MNIST and CelebA. On FONTS, the manifold is known; on the other datasets, class manifolds have been approximated using VAE-GANs. Notice that the crafted on-manifold adversarial examples correspond to meaningful manipulations of the image – as long as the learned class-manifolds are good approximations. This can best be seen considering the (normalized) difference images (or the magnitude thereof for CelebA).

D. On-Manifold Adversarial Examples

In Fig. 12, we show additional examples of regular and on-manifold adversarial examples, complementing the examples in Fig. 2 of the main paper. On FONTS, both using the true and the approximated manifold, on-manifold adversarial examples reflect the underlying invariances of the data, i.e., the transformations employed in the generation process. This is in contrast to the corresponding regular adversarial examples and their (seemingly) random noise patterns. We note that regular and on-manifold adversarial examples can best be distinguished based on their difference to the original test image – although both are perceptually close to the original image. Similar observations hold on EMNIST and F-MNIST. However, especially on F-MNIST and CelebA, the discrepancy between true images and on-manifold adversarial examples becomes visible. This is the "cost" of approximating the underlying manifold using VAE-GANs. More examples can be found in Fig. 22 at the end of this document.

E. L_2 and Transfer Attacks

In the main paper, see Section 3.1, we primarily focus on the L_∞ white-box attack by Madry et al. [16]. Here, we further consider the L_2 variant, which, given image x with label y and classifier f, maximizes the training loss, i.e.,

$$\max_\delta \mathcal{L}(f(x+\delta), y) \text{ s.t. } \|\delta\|_2 \leq \epsilon, \tilde{x}_i \in [0,1], \qquad (12)$$

to obtain an adversarial example $\tilde{x} = x + \delta$. We use $\epsilon = 1.5$ for regular adversarial examples and $\epsilon = 0.3$ for on-manifold adversarial examples. For optimization, we utilize projected ADAM [11]: after each iteration, $\tilde{x}$ is projected onto the L_2-ball of radius ϵ using

$$\tilde{x}' = \tilde{x} \cdot \max\left(1, \frac{\epsilon}{\|\tilde{x}\|_2}\right) \qquad (13)$$

and clipped to $[0,1]$. We use a learning rate of 0.005 and we note that ADAM includes momentum, as suggested in [4]. Optimization stops as soon as the label changes, or runs for a maximum of 40 iterations. The perturbation δ is initialized randomly as follows:

$$\delta = u\epsilon \frac{\delta'}{\|\delta'\|_2}, \quad \delta' \sim \mathcal{N}(0, I), u \sim U(0,1). \qquad (14)$$

Here, $U(0,1)$ refers to the uniform distribution over $[0,1]$. This results in δ being in the ϵ-ball and uniformly distributed over distance and direction. Note that this is in contrast to sampling uniformly wrt. the volume of the ϵ-ball. The same procedure applies to the L_∞ attack where the projection onto the ϵ-ball is achieved by clipping. The attack can also be used to obtain on-manifold adversarial examples, as described in Section 3.3 of the main paper. Then, optimization in Eq. (12) is done over the perturbation ζ in latent space, with constraint $\|\zeta\|_2 \leq \eta$. The adversarial example is obtained as $\tilde{x} = \text{dec}(z+\zeta)$ with z being the latent code of image x and dec being the true or approximated generative model, i.e., decoder.

We also consider the L_2 white box attack by Carlini and Wagner [2]. Instead of directly maximizing the training loss, Carlini and Wagner propose to use a surrogate objective on the classifier's logits l_y:

$$F(\tilde{x}, y) = \max(-\kappa, l_y(\tilde{x}) - \max_{y' \neq y} l_{y'}(\tilde{x})). \qquad (15)$$

Compared to the training loss, which might be close to zero for a well-trained network, F is argued to provide more useful gradients [2]. Then,

$$\min_\delta F(x+\delta, y) + \lambda\|\delta\|_2 \text{ s.t. } \tilde{x}_i \in [0,1] \qquad (16)$$

is minimized by reparameterizing δ in terms of $\delta = {}^1\!/\!_2(\tanh(\omega)+1) - x$ in order to ensure the image-constraint,

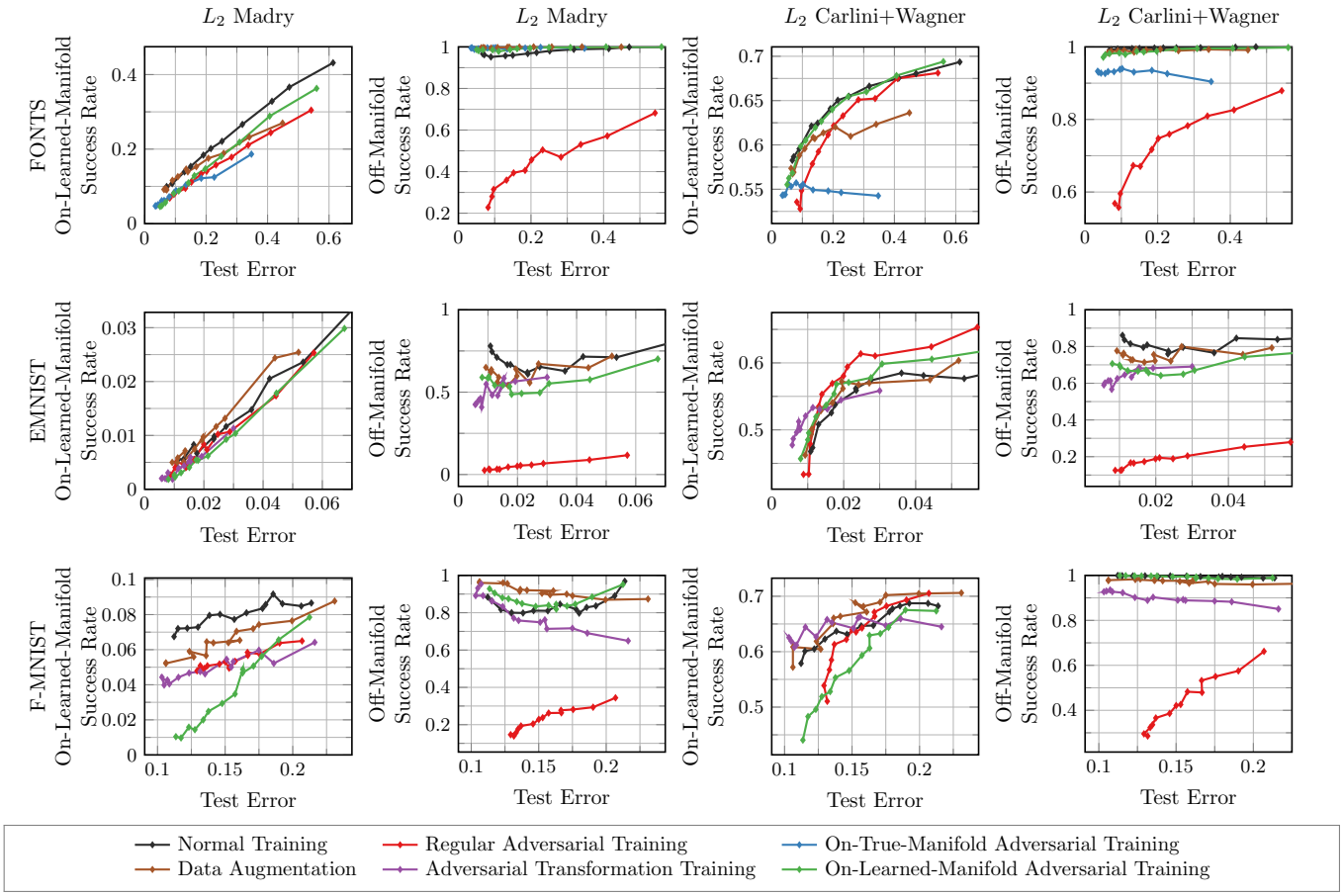

Figure 13: L_2 attacks of Madry et al. [16] and Carlini and Wagner [2] on FONTS, EMNIST and F-MNIST. In all cases, we plot regular or on-manifold success rate against test error. Independent of the attack, we can confirm that on-manifold robustness is strongly related to generalization, while regular robustness is independent of generalization.

i.e., $\tilde{x}_i \in [0, 1]$. In practice, we empirically chose $\kappa = 1.5$, use 120 iterations of ADAM [11] with learning rate 0.005 and $\lambda = 1$. Again, this attack can be used to obtain on-manifold adversarial examples, as well.

As black-box attack we transfer L_∞ Madry adversarial examples from a held out model, as previously done in [14, 23, 17]. The held out transfer model is trained normally, i.e., without any data augmentation or adversarial training, on 10k training images for 20 epochs (as outlined in Section 3.1 of the main paper). The success rate of these transfer attacks is computed with respect to images that are correctly classified by both the transfer model and the target model.

Extending the discussion of Sections 3.4 and 3.5 of the main paper, Fig. 13 shows results on FONTS, EMNIST and F-MNIST considering both L_2 attacks, i.e., Madry et al. [16] and Carlini and Wagner [2]. In contrast to the L_∞ Madry attack, we observe generally lower success rates. Nevertheless, we can observe a clear relationship between on-manifold success rate and test error. The exact form of this relationship, however, depends on the attack; for the L_2 Madry attack, the relationships seems to be mostly linear (especially on FONTS and EMNIST), while it seems non-linear for the L_2 Carlini and Wagner attack. Furthermore, the independence of regular robustness and generalization can be confirmed, i.e., regular success rate is roughly constant when test error varies – again, with the exception of regular adversarial training. Finally, for completeness, in Fig. 15, we illustrate that the Carlini+Wagner L_2 attack also results in regular adversarial examples leaving the manifold.

In Fig. 14, we also consider the black-box case, i.e., without access to the target model. While both observations from above can be confirmed, especially on FONTS and EMNIST, the results are significantly less pronounced. This is mainly due to the significantly lower success rate of transfer attacks – both regarding regular and on-manifold adversarial examples. Especially on EMNIST and F-MNIST, success rate may reduce from previously 80% or higher to 10% or lower. This might also explain the high variance on EMNIST and F-MNIST regarding regular robustness. Overall, we demonstrate that our claims can be confirmed in both white- and black-box settings as well as using different attacks [16, 2] and norms.

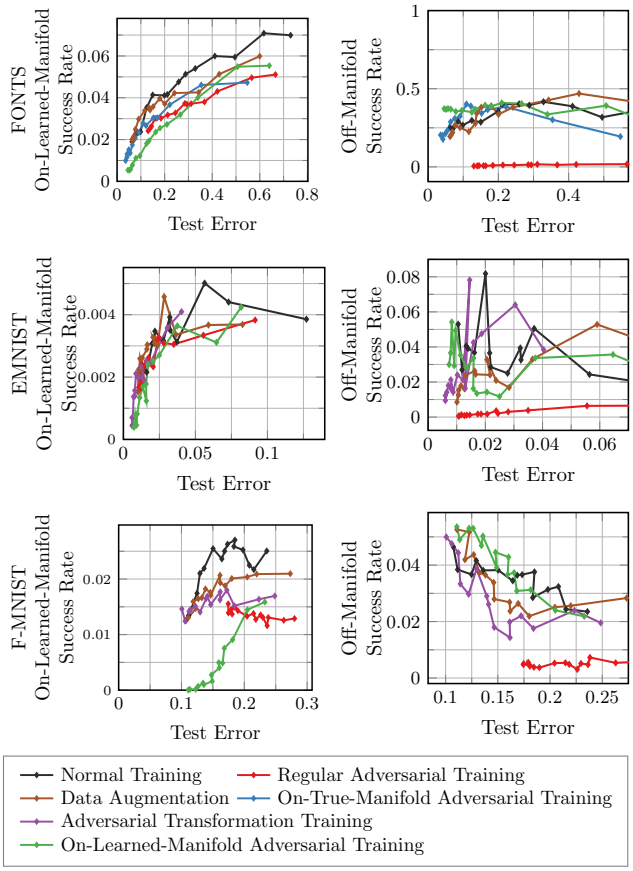

Figure 14: Transfer attacks on FONTS, EMNIST and F-MNIST. We show on-manifold (left) and regular success rate (right) plotted against test error. In spite of significantly lower success rates, transfer attacks also allow to confirm the strong relationship between on-manifold success rate and test error, while – at least on FONTS and EMNIST– regular success rate is independent of test error.

F. Influence of Network Architecture

Also in relation to the discussion in Sections 3.4 and 3.5 of the main paper, Fig. 16 shows results on FONTS, EMNIST and F-MNIST using multi-layer perceptrons instead of convolutional neural networks. Specifically, we consider a network with 4 hidden layers, using 128 hidden units each; each layer is followed by ReLU activations and batch normalization [8]; training strategy, however, remains unchanged. Both of our claims, i.e., that on-manifold robustness is essentially generalization but regular robustness is independent of generalization, can be confirmed. Especially regarding the latter, results are more pronounced using multi-layer perceptrons: except for regular adversarial training, success rate stays nearly constant at 100% irrespective of test error. Overall, these results suggest that our claims generally hold for the class of (deep) neural networks, irrespective of architectural details.

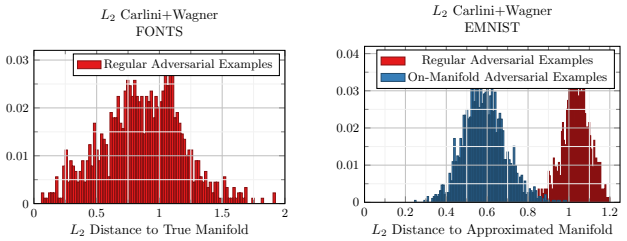

Figure 15: Distance of Carlini+Wagner adversarial examples to the true, on FONTS (left), or approximated, on EMNIST (right), manifold. As before, we show normalized histograms of the L_2 distance of adversarial examples to their projections onto the manifold. Even for different attacks and the L_2 norm, regular adversarial examples seem to leave the manifold.

In order to further validate our claims, we also consider variants of two widely used, state-of-the-art architectures: ResNet-13 [7] and VGG [19]. For VGG, however, we removed the included dropout layers. The main reason is that randomization might influence robustness, e.g., see [1]. Additionally, we only use 2 stages of model A, see [19], in order to deal with the significantly lower resolution of 28×28 on FONTS, EMNIST and F-MNIST; finally, we only use 1024 hidden units in the fully connected layers. Fig. 17 shows results on FONTS and F-MNIST (which are significantly more difficult than EMNIST) confirming our claims.

G. From Class Manifolds to Data Manifold

In the context of Sections 3.3 and 3.4 of the main paper, we consider approximating the manifold using class-agnostic VAE-GANs. Instead of the class-conditionals $p(x|y)$ of the data distribution, the marginals $p(x)$ are approximated, i.e., images of different classes are embedded in the same latent space. Then, however, ensuring label invariance, as required by our definition of on-manifold adversarial examples, becomes difficult:

Definition 1 (On-Manifold Adversarial Example)**.** Given the data distribution p, an on-manifold adversarial example for x with label y is a perturbed version $\tilde{x}$ such that $f(\tilde{x}) \neq y$ but $p(y|\tilde{x}) > p(y'|\tilde{x}) \forall y' \neq y$.

Therefore, we attempt to ensure Def. 1 through a particularly small L_∞-constraint on the perturbation, specifically $\|\zeta\|_\infty \leq \eta$ with $\eta = 0.1$ where ζ is the perturbation applied in the latent space. Still, as can be seen in Fig. 18, on-manifold adversarial examples might cross class boundaries, i.e., they change their actual label rendering them invalid according to our definition.

In Fig. 19, we clearly distinguish between on-*class*-manifold and on-*data*-manifold adversarial training, corresponding to the used class-specific or -agnostic VAE-GANs. Robustness, however, is measured wrt. on-data-manifold adversarial examples. As can be seen, the positive

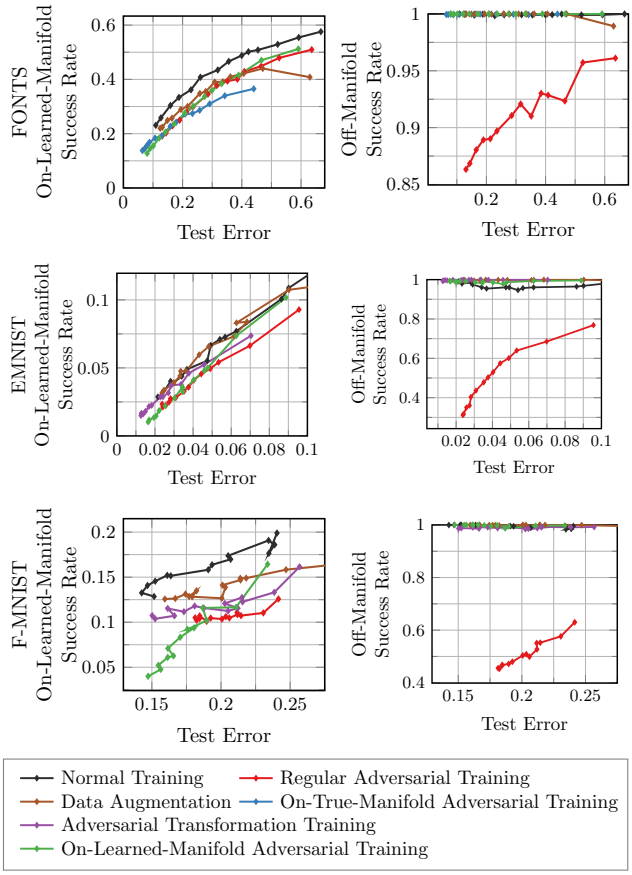

Figure 16: Experiments with multilayer-perceptrons on FONTS, EMNIST and F-MNIST. We plot on-manifold (left) or regular success rate (right) against test error. On-manifold robustness is strongly related to generalization, while regular robustness seems mostly independent of generalization.

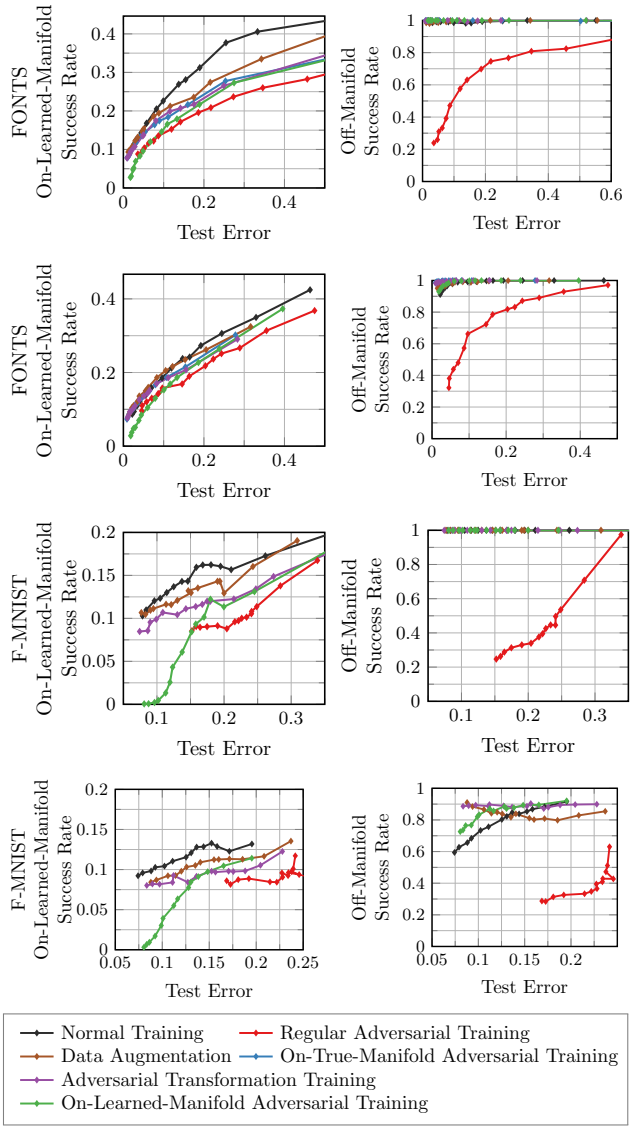

Figure 17: Experiments with ResNet-13 (top) and VGG (bottom) on FONTS and F-MNIST. We plot on-manifold (left) or regular success rate (right) against test error. As in Fig. 16, our claims can be confirmed for these network architectures, as well.

effect of on-manifold adversarial training diminishes when using on-data-manifold adversarial examples during training. Both, on FONTS and EMNIST, generalization slightly decreases in comparison to normal training because adversarial examples are not useful for learning the task if label invariance cannot be ensured. When evaluating robustness against on-data-manifold adversarial examples, however, the relation of on-data-manifold robustness to generalization can clearly be seen. Overall, this shows that this relationship also extends to more general, less strict definitions of on-manifold adversarial examples.

H. Baselines and Adversarial Training Variants

In the main paper, see Section 3.1, we consider the adversarial training variant by Madry et al. [16], i.e.,

$$\min_{w} \sum_{n=1}^{N} \max_{\|\delta\|_\infty \leq \epsilon} \mathcal{L}(f(x_n + \delta; w), y_n), \tag{17}$$

where f is the classifier with weights w, $\mathcal{L}$ is the cross-entropy loss and x_n, y_n are training images and labels. In contrast to [16], we train on 50% clean and 50% adversarial examples [20, 6]. The inner optimization problem is run for full 40 iterations, as described in Section E without early stopping. Here, we additionally consider the *full variant*, i.e., training on 100% adversarial examples; and the *weak variant*, i.e., stopping the inner optimization prob-

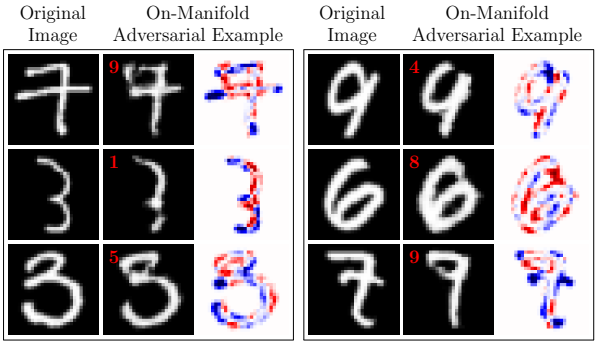

Figure 18: On-manifold adversarial examples crafted using class-agnostic VAE-GANs on EMNIST. We show examples illustrating the problematic of unclear class boundaries within the learned manifold. On-manifold adversarial examples are not guaranteed to be label invariant, i.e., they may change the actual, true label according to the approximate data distribution.

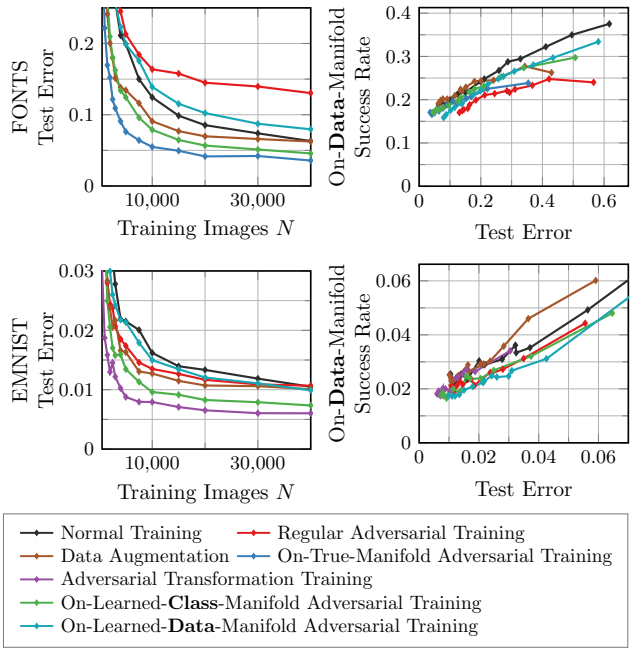

Figure 19: Test error and on-data-manifold success rate on FONTS and EMNIST. Using class-agnostic VAE-GANs, without clear class boundaries, on-manifold adversarial training looses its effectiveness – the on-manifold adversarial examples cross the true class boundaries too often. The strong relationship between on-manifold robustness and generalization can still be confirmed.

lem as soon as the label changes. Additionally, we consider random perturbations as baseline, i.e., choosing the perturbations δ uniformly at random without any optimization. The same variants and baselines apply to on-manifold adversarial training and adversarial transformation training.

In Section 3.6 of the main paper, we observed that different training strategies might exhibit different robustness-generalization characteristics. For example, regular adversarial training renders the learning problem harder: in addition to the actual task, the network has to learn (seemingly) random but adversarial noise directions leaving the manifold. In Fig. 20, we first show that training on randomly perturbed examples (instead of adversarially perturbed ones) is not effective, neither in image space nor in latent space. This result highlights the difference between random and adversarial noise, as also discussed in [5]. For regular adversarial training, the strength of the adversary primarily influences the robustness-generalization trade-off; for example, the weak variant increases generalization while reducing robustness. Note that this effect also depends on the difficulty of the task, e.g., FONTS is considerably more difficult than EMNIST. For on-manifold adversarial training, in contrast, the different variants have very little effect; generalization is influenced only slightly, while regular robustness is – as expected – not influenced.

I. Definition of Adversarial Examples

Adversarial examples are assumed to be label-invariant, i.e., the actual, true label does not change. For images, this is usually enforced using a norm-constraint on the perturbation – e.g., cf. Eq. (12); on other modalities, however, this norm-constraint might not be sufficient. In Section 3.3 of the main paper, we provide a definition for on-manifold adversarial examples based on the true, underlying data distribution – as restated in Def. 1. Here, we use this definition to first discuss a simple and intuitive example before considering the theoretical argument of [21], claiming that robust *and* accurate models are not possible on specific datasets; an argument in contradiction to our results

Let the observations x and labels y be drawn from a data distribution p, i.e., $x, y \sim p(x, y)$. Then, given a classifier f we define adversarial examples as follows:

Definition 2 (Adversarial Example)**.** Given the data distribution p, an adversarial example for x with label y is a perturbed version $\tilde{x}$ such that $f(\tilde{x}) \neq y$ but $p(y|\tilde{x}) > p(y'|\tilde{x}) \forall y' \neq y$.

In words, adversarial examples must not change the actual, true label wrt. the data distribution. Note that this definition is identical to Def. 1 for on-manifold adversarial examples. For the following toy examples, however, the data distribution has non-zero probability on the whole domain or we only consider adversarial examples $\tilde{x}$ with $p(\tilde{x}) > 0$ such that Def. 2 is well-defined. We leave a more general definition of adversarial examples for future work.

We illustrate Def. 2 on an intuitive, binary classification task. Specifically, the classes $y = 1$ and $y = -1$ are uniformly distributed, i.e., $p(y = 1) = p(y = -1) = 0.5$ and observations are drawn from point masses on 0 and ϵ:

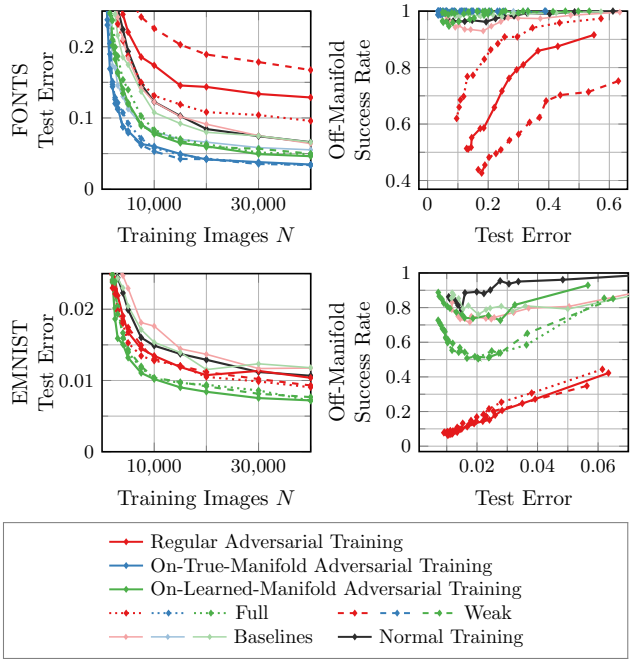

Figure 20: Adversarial training variants and baselines on FONTS and EMNIST. For adversarial training, we consider the *full variant*, i.e., training on 100% adversarial examples, and the *weak variant*, i.e., stopping the inner optimization problem of Eq. (17) as son as the first adversarial example is found. For regular adversarial training, the strength of the adversary determines the robustness-generalization trade-off; for on-manifold adversarial training, the ideal strength depends on the approximation quality of the used VAE-GANs.

$$p(x = 0|y = 1) = 1 \tag{18}$$
$$p(x = \epsilon|y = -1) = 1 \tag{19}$$

This problem is linearly separable for any $\epsilon > 0$; however, it seems that no classifier will be adversarially robust against perturbations of absolute value ϵ. For simplicity, we consider the observation $x = 0$ with $y = 1$ and the adversarial example $\tilde{x} = x + \epsilon = \epsilon$. Then, verifying Def. 2 yields a contradiction:

$$0 = p(y = 1|x = \epsilon) \not> p(y = -1|x = \epsilon) = 1. \tag{20}$$

It turns out, $\tilde{x} = \epsilon$ is not a proper adversarial example. This example illustrates that an exact definition of adversarial examples, e.g., Def. 2, is essential to study the robustness of such toy datasets.

I.1. Discussion of [21]

In [21], Tsipras et al. argue that there exists an inherent trade-off between regular robustness and generalization based on a slightly more complex toy example; we follow the notation in [21]. Specifically, for labels $y = 1$ and $y = -1$ with $p(y = 1) = p(y = -1) = 0.5$, the observations $x \in \{-1, 1\} \times \mathbb{R}$ are drawn as follows[2]:

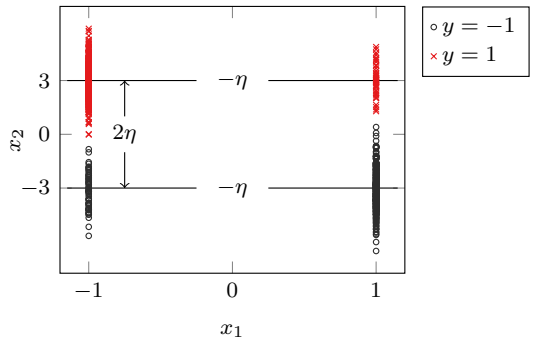

Figure 21: Illustration of the toy dataset considered by Tsipras et al. in [21] and defined in Eq. (21). For labels $y = 1$ and $y = -1$, the two-dimensional observations $x \in \{-1, 1\} \times \mathbb{R}$ are plotted. The first dimension, i.e., x_1, mirrors the label with probability 0.9; the second dimension, i.e., x_2, is drawn from a Gaussian $\mathcal{N}(y3, I)$, i.e., η from the text is 3. As illustrated on the left, perturbing an observation x with label $y = 1$ but $x_1 = -1$ by $2\eta = 6$ results in an adversarial example $\tilde{x}$ indistinguishable from observations with label $y = -1$.

$$p(x_1|y) = \begin{cases} p & \text{if } x_1 = y \\ 1 - p & \text{if } x_1 = -y \end{cases}, \qquad p(x_2|y) = \mathcal{N}(x_2; y\eta, 1) \tag{21}$$

where η defines the degree of overlapping between the two classes and $p \geq 0.5$. Fig. 21 illustrates this dataset for $p = 0.9$ and $\eta = 3$. For a L_∞-bounded adversary with $\epsilon \geq 2\eta$, Tsipras et al. show that no model can be both accurate and robust. Specifically, for x with $y = 1$ but $x_1 = -1$ and $x_2 = \eta$, we consider replacing x_2 with $\tilde{x}_2 = x_2 - 2\eta = -\eta$, as considered in [21]. However, this adversary does not produce proper adversarial examples according to our definition. Indeed,

$$\begin{aligned} p(y = 1|x = \tilde{x}) &= p(y = 1|x_1 = -1) \cdot p(y = 1|x_2 = -\eta) \\ &= (1 - p) \cdot \mathcal{N}(x_2 = -\eta; \eta, 1) \\ &\not> p \cdot \mathcal{N}(x_2 = -\eta; -\eta, 1) \\ &= p(y = -1|x_1 = -1) \cdot p(y = -1|x_2 = -\eta) \\ &= p(y = -1|x = \tilde{x}) \end{aligned} \tag{22}$$

which contradicts our definition. Thus, in light of Def. 2, the suggested trade-off of Tsipras et al. is questionable. However, we note that this argument explicitly depends on our

[2]Note that, for simplicity and convenience, we consider the 2-dimensional case; Tsipras et al. consider the general D-dimensional case, where x_1 remains unchanged and $x_2, \ldots, x_D$ are drawn from the corresponding Gaussian, cf. (21).

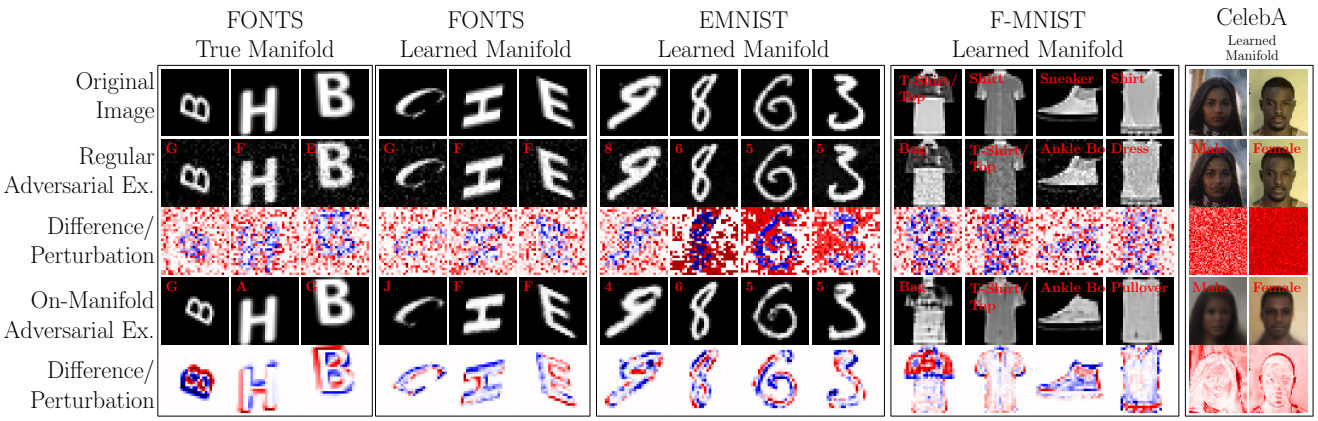

Figure 22: Regular and on-manifold adversarial examples on FONTS, EMNIST, F-MNIST and CelebA. On FONTS, the manifold is known; otherwise, class manifolds have been approximated using VAE-GANs. In addition to the original test images, we also show the adversarial examples and their (normalized) difference (or the magnitude thereof for CelebA).

definition of proper and invalid adversarial examples, i.e., Def. 2; other definitions of adversarial examples or adversarial robustness, e.g., in the context of the adversarial loss defined in [21], may lead to different conclusions.

References

[1] Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv.org*, abs/1802.00420, 2018. 6

[2] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *SP*, 2017. 1, 4, 5

[3] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. *arXiv.org*, abs/1702.05373, 2017. 1, 2

[4] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018. 4

[5] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Robustness of classifiers: from adversarial to random noise. In *NIPS*, 2016. 8

[6] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv.org*, abs/1412.6572, 2014. 7

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[8] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 2, 6

[9] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *NIPS*, 2015. 1

[10] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv.org*, abs/1412.6980, 2014. 2

[11] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 2, 4, 5

[12] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 2

[13] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *ICML*, 2016. 1, 2

[14] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *ICLR*, 2017. 5

[15] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 2

[16] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *ICLR*, 2018. 1, 4, 5, 7

[17] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *AsiaCCS*. ACM, 2017. 5

[18] Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, and Shakir Mohamed. Variational approaches for auto-encoding generative adversarial networks. *arXiv.org*, abs/1706.04987, 2017. 1, 2

[19] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv.org*, abs/1409.1556, 2014. 6

[20] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv.org*, abs/1312.6199, 2013. 7

[21] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv.org*, abs/1805.12152, 2018. 1, 8, 9, 10

[22] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv.org*, abs/1708.07747, 2017. 2

[23] Cihang Xie, Zhishuai Zhang, Jianyu Wang, Yuyin Zhou, Zhou Ren, and Alan L. Yuille. Improving transferability of adversarial examples with input diversity. *CoRR*, abs/1803.06978, 2018. 5