Certifiably Robust Variational Autoencoders

Ben Barrett[1] **Alexander Camuto**[1,3] **Matthew Willetts**[2,3] **Tom Rainforth**[1]
[1]University of Oxford [2]University College London [3]Alan Turing Institute

Abstract

We introduce an approach for training variational autoencoders (VAEs) that are certifiably robust to adversarial attack. Specifically, we first derive actionable bounds on the minimal size of an input perturbation required to change a VAE's reconstruction by more than an allowed amount, with these bounds depending on certain key parameters such as the Lipschitz constants of the encoder and decoder. We then show how these parameters can be controlled, thereby providing a mechanism to ensure *a priori* that a VAE will attain a desired level of robustness. Moreover, we extend this to a complete practical approach for training such VAEs to ensure our criteria are met. Critically, our method allows one to specify a desired level of robustness *upfront* and then train a VAE that is guaranteed to achieve this robustness. We further demonstrate that these *Lipschitz–constrained* VAEs are more robust to attack than standard VAEs in practice.

1 INTRODUCTION

Variational autoencoders (VAEs) are a powerful method for learning deep generative models (Kingma and Welling, 2013; Rezende et al., 2014), finding application in areas such as image and language generation (Razavi et al., 2019; Kim et al., 2018) as well as representation learning (Higgins et al., 2017a).

Like other deep learning methods (Szegedy et al., 2013), VAEs are susceptible to adversarial attacks, whereby small perturbations of an input can induce meaningful, unwanted changes in output. For example, VAEs can be induced to reconstruct images similar to an adversary's target through only moderate perturbation of the input image (Tabacof et al., 2016; Gondim-Ribeiro et al., 2018; Kos et al., 2018).

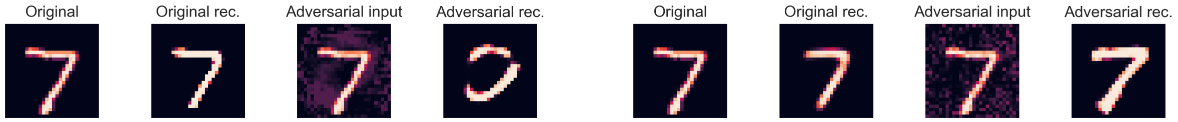
Two reasons why this is particularly undesirable are a) that VAEs have been used to improve the robustness of classifiers (Schott et al., 2018; Ghosh et al., 2019) and b) the encodings of VAEs are commonly used in downstream tasks (Ha and Schmidhuber, 2018; Higgins et al., 2017b). Yet another is that the susceptibility of VAEs to distortion from input perturbations challenges an original ambition for VAEs: that they should capture "semantically meaningful [...] factors of variation in data" (Kingma and Welling, 2019). If this ambition is to be fulfilled, VAEs should be more robust to spurious inputs, and so the robustness of VAEs is intrinsically desirable.

While previous work has already sought to obtain more robust VAEs empirically (Willetts et al., 2021; Cemgil et al., 2020b,a), this work lacks formal guarantees. This is a meaningful worry because in other model classes, robustification techniques showing promise empirically, but lacking guarantees, have later been circumvented by more sophisticated attacks (Athalye et al., 2018; Uesato et al., 2018). It stands to reason that existing techniques for robustifying VAEs might be similarly ineffectual. Further, though previous theoretical work (Camuto et al., 2020) can ascertain robustness *post-training*, it cannot enforce and control robustness *a priori*, before training.

Our work looks to alleviate these issues by providing VAEs whose robustness levels can be controlled and certified by design. To this end, we show how *certifiably robust* VAEs can be learned by enforcing Lipschitz continuity in the encoder and decoder, which explicitly upper-bounds changes in their outputs with respect to changes in input.

We derive two bounds on the robustness of these models, each covering a slightly different setting. First, we derive a per-datapoint lower bound that guarantees a minimum probability for reconstructions of distorted inputs being within some distance of the reconstructions of undistorted inputs. More precisely, this per-

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151. Copyright 2022 by the author(s).



(a) Standard VAE, $||\boldsymbol{\delta}||_2 \leq 3$.

(b) Lipschitz-constrained VAE, $||\boldsymbol{\delta}||_2 \leq 3$.

Figure 1: A maximum damage attack (4) on a standard VAE and Lipschitz-constrained VAE, respectively, for the same perturbation norm constraint. Unlike those of the standard VAE, the Lipschitz-constrained VAE's reconstructions are robust under attack. Appendix E supplements these results with latent space attacks (3).

datapoint lower bound is on the probability that the ℓ_2 distance between an attacked VAE's reconstruction and its original reconstruction is less than some value r. This probability is with reference to the stochasticity of sampling in a VAE's latent space. Using the previous bound, we can then obtain a margin that holds for all inputs. This second, *global* bound means that we can guarantee, for *any* input, that perturbations within the margin induce reconstructions that fall within an r–sized ball of the original reconstruction with *at least* some specified probability ϵ.

The latter margin is the first of its kind for VAEs: a margin that is input-agnostic and can have its value specified *a priori* from setting a small number of network hyperparameters. It thus enables VAEs with chosen levels of robustness.

In summary, our key contributions are to a) develop the first *certifiably robust* VAE approach, wherein Lipschitz continuity constraints are used during training to ensure certain robustness properties are met; b) provide accompanying theory to show that our approach allows a desired level of robustness to be *guaranteed upfront*; and c) experimentally validate that our approach works in practice (see e.g. Figure 1).

2 BACKGROUND

2.1 VAEs

Assume we have a collection of observations $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ with $\mathbf{x} \in \mathcal{X}$, which is generated according to an unknown process involving latent variables $\mathbf{z} \in \mathcal{Z}$. We want to learn a latent variable model with joint density $p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})$, parameterized by θ, that captures this process. Learning θ by maximum likelihood is often intractable; variational inference addresses this intractability by introducing inference model $q_\phi(\mathbf{z}|\mathbf{x})$ (Kingma and Welling, 2019), parameterized by ϕ, which yields the "ELBO", a tractable lower bound on the marginal log likelihood $\log p_\theta(\mathbf{x})$,

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right). \tag{1}$$

Here, $\mathrm{KL}\left(\cdot||\cdot\right)$ denotes the Kullback-Leibler divergence, while θ and ϕ represent the parameters of neural networks — the *decoder* and *encoder network*, respectively, of the VAE — which can be optimized using unbiased gradient estimates obtained through Monte Carlo samples from $q_\phi(\mathbf{z}|\mathbf{x})$.

Given a VAE, we will refer to sampling $\mathbf{z}_i \sim q_\phi(\mathbf{z}|\mathbf{x}_i)$ on input $\mathbf{x}_i$ as the *encoding process*, and, following convention, to $g_\theta(\mathbf{z}_i)$ as a *reconstruction* of $\mathbf{x}_i$, where $g_\theta(\cdot)$ denotes the *deterministic component of the decoder* (Kumar and Poole, 2020).

2.2 Adversarial Attacks On VAEs

In adversarial attacks, an adversary tries to alter the behavior of a model. Although much work has focused on classifiers, adversarial attacks have also been proposed for VAEs, whereby the model is "fooled" into reconstructing an unintended output. More formally, given original input $\mathbf{x}_o$ and the adversary's target output $\mathbf{x}_t$, the attacker seeks a perturbation $\boldsymbol{\delta} \in \mathcal{X}$ such that the VAE's reconstruction of perturbed input $(\mathbf{x}_o + \boldsymbol{\delta})$ is similar to $\mathbf{x}_t$.

The best performing attack on VAEs in the current literature is a *latent space attack* (Tabacof et al., 2016; Gondim-Ribeiro et al., 2018; Kos et al., 2018), where an adversary perturbs input $\mathbf{x}_o$ to have a posterior q_ϕ similar to that of the target $\mathbf{x}_t$, optimizing

$$\arg\min_{\boldsymbol{\delta}} \quad \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}_o + \boldsymbol{\delta})||q_\phi(\mathbf{z}|\mathbf{x}_t)\right) + \lambda||\boldsymbol{\delta}||_2. \tag{2}$$

In (2), the second term implicitly constrains the perturbation norm; in our work, we explicitly constrain this norm by some constant $c \in \mathbb{R}^+$ to ensure more consistent comparisons:

$$\arg\min_{\boldsymbol{\delta}:\ ||\boldsymbol{\delta}||_2 \leq c} \quad \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}_o + \boldsymbol{\delta})||q_\phi(\mathbf{z}|\mathbf{x}_t)\right). \tag{3}$$

We also use another type of attack, the *maximum damage attack* (Camuto et al., 2020), which for $\mathbf{z}_{\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x}_o + \boldsymbol{\delta})$, $\mathbf{z}_{\neg\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x}_o)$, and some constant $c \in \mathbb{R}^+$ optimizes

$$\arg\max_{\boldsymbol{\delta}:\ ||\boldsymbol{\delta}||_2 \leq c} \quad ||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2. \tag{4}$$

2.3 Defining Robustness In VAEs

VAE reconstructions are typically continuous–valued, and a VAE's encoder, $q_\phi(\mathbf{z}|\mathbf{x})$, is usually chosen to be a continuous distribution. Any change to a VAE's input will thus almost surely result in a change in its reconstructions, since changes to the input will translate to changes in $q_\phi(\mathbf{z}|\cdot)$ and, in turn, to changes in the reconstruction $g_\theta(\mathbf{z})$ (Camuto et al., 2020).

This observation rules out established robustness criteria that specify robustness using margins around inputs within which model outputs are constant (Cohen et al., 2019; Salman et al., 2019). To further complicate matters, VAEs are probabilistic: a VAE's outputs will vary even under the same input. To account for these considerations, we employ the robustness criterion of Camuto et al. (2020):

Definition 2.1. ((r, ϵ)-robustness) For $r \in \mathbb{R}^+$ and $\epsilon \in [0, 1)$, a model f operating on a point $\mathbf{x}$ and outputting a continuous random variable is (r, ϵ)-robust to a perturbation $\boldsymbol{\delta}$ if and only if

$$\mathbb{P}\left[||f(\mathbf{x}+\boldsymbol{\delta}) - f(\mathbf{x})||_2 \leq r\right] > \epsilon.$$[1]

The notion of (r, ϵ)-robustness states that a model is robust if, with probability greater than ϵ, changes in the model's outputs induced by input perturbation $\boldsymbol{\delta}$ fall within a hypersphere of radius r about the model's outputs on the unperturbed input. The smaller the r and the larger the ϵ for which (r, ϵ)-robustness holds, the stricter the notion of robustness which is implied. We will refer to $\mathbb{P}\left[||f(\mathbf{x}+\boldsymbol{\delta}) - f(\mathbf{x})||_2 \leq r\right]$ as the *r-robustness probability*. Note that, by enabling flexible specification of r and ϵ, the (r, ϵ)-robustness criterion can be made *arbitrarily strong* to suit the level of robustness required.

The notion of (r, ϵ)-robustness naturally yields that of an *(r, ϵ)-robustness margin* (Camuto et al., 2020):

Definition 2.2. ((r, ϵ)-robustness margin) For $r \in \mathbb{R}^+$ and $\epsilon \in [0, 1)$, a model f has (r, ϵ)-robustness margin $R^{(r,\epsilon)}(\mathbf{x})$ about input $\mathbf{x}$ if $||\boldsymbol{\delta}||_2 < R^{(r,\epsilon)}(\mathbf{x}) \implies \mathbb{P}\left[||f(\mathbf{x}+\boldsymbol{\delta}) - f(\mathbf{x})||_2 \leq r\right] > \epsilon$.

A model with an (r, ϵ)-robustness margin on $\mathbf{x}$ can only be undermined by more than r by perturbations with norm less than $R^{(r,\epsilon)}(\mathbf{x})$ with probability less than $(1 - \epsilon)$. In other words, for appropriately chosen r and ϵ, we can guarantee that a model with an (r, ϵ)-robustness margin cannot be consistently undermined by input perturbations of $\mathbf{x}$ up to a particular magnitude (Camuto et al., 2020).

2.4 Lipschitz Continuity

For completeness, recall the following definition:

Definition 2.3. (Lipschitz continuity) A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is Lipschitz continuous if for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$, $||f(\mathbf{x}_1) - f(\mathbf{x}_2)||_2 \leq M||\mathbf{x}_1 - \mathbf{x}_2||_2$ for constant $M \in \mathbb{R}^+$. The least M for which this holds is called the *Lipschitz constant* of f.

If a function f is Lipschitz continuous with Lipschitz constant M, we say f is M-Lipschitz.

3 CERTIFIABLY ROBUST VAES

We now introduce our approach for achieving a VAE whose robustness levels can be controlled and certified. We do so by targeting the "smoothness" of a VAE's encoder and decoder network, requiring these to be Lipschitz continuous, since a VAE's vulnerability to input perturbation is thought to inversely correlate with the smoothness of its encoder and decoder. By maintaining Lipschitz continuity with known, set Lipschitz constants, we will be able to obtain a chosen degree of robustness *a priori*.

3.1 Bounding The r-Robustness Probability

We first construct an approach for guaranteeing that a VAE's reconstructions will change only to a particular degree under input distortions. We achieve this by specifying our VAEs such that their r-robustness probability is bounded from below.

In the standard setting, this yields an input-dependent characterization of the behavior of the VAE, while bounding the encoder standard deviation or taking it to be a hyperparameter yields global, input-agnostic bounds. This means that for a given input perturbation norm we can guarantee output similarity up to a threshold with a particular probability. Our bounds provide the first global guarantees about the robustness behavior of a VAE.

We use the ℓ_2 distance as our notion of similarity because it has been the basis for previous theoretical work on VAE robustness (Camuto et al., 2020) and also corresponds to the log probability of a Gaussian — a frequently-used likelihood function for VAEs.

The following result shows that, under the common choice of a diagonal-covariance multivariate Gaussian encoder, a lower bound on the r-robustness probability can be provided for VAEs.[2] We use the parameteri-

[1] We use the ℓ_2 norm but the following definitions could also be stated with respect to other norms.

[2] Note that our results operate on the basis that the forward pass involves sampling both at train and test time; sampling at test time is not unreasonable because of the adversarial setting, wherein a VAE's sampling step may be

zation $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}; \mu_\phi(\mathbf{x}), \mathtt{diag}\left(\sigma_\phi^2(\mathbf{x})\right)\right)$, where $\mu_\phi : \mathcal{X} \to \mathbb{R}^{d_z}$ is the *encoder mean* and $\sigma_\phi : \mathcal{X} \to \mathbb{R}^{d_z}_{\geq 0}$ is the *encoder standard deviation.*[3]

Theorem 1 (Probability Bound)**.** *Assume $q_\phi(\mathbf{z}|\mathbf{x})$ is as above and that the deterministic component of the VAE decoder $g_\theta(\cdot)$ is a-Lipschitz, the encoder mean $\mu_\phi(\cdot)$ is b-Lipschitz, and the encoder standard deviation $\sigma_\phi(\cdot)$ is c-Lipschitz. Finally, let $\mathbf{z}_{\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x}+\boldsymbol{\delta})$ and $\mathbf{z}_{\neg\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x})$. Then for any $r \in \mathbb{R}^+$, any $\mathbf{x} \in \mathcal{X}$, and any input perturbation $\boldsymbol{\delta} \in \mathcal{X}$,*

$$\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] \geq 1 - \min\left\{p_1(\mathbf{x}), p_2(\mathbf{x})\right\},$$

where

$$p_1(\mathbf{x}) := \min\left(1, \frac{a^2\left(b^2||\boldsymbol{\delta}||_2^2 + (c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2\right)}{r^2}\right)$$

$$p_2(\mathbf{x}) := \begin{cases} C(d_z)\frac{u(\mathbf{x})^{\frac{d_z}{2}}\exp\left\{-\frac{u(\mathbf{x})}{2}\right\}}{u(\mathbf{x})-d_z+2} & \left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right) \geq 0; \\ & d_z \geq 2; \\ & u(\mathbf{x}) > d_z - 2 \\ 1 & o.w. \end{cases}$$

for $u(\mathbf{x}) := \frac{\left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)^2}{(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2}$ *and constant* $C(d_z) := \frac{1}{\sqrt{\pi}}\exp\left\{\frac{1}{2}(d_z - (d_z - 1)\log d_z)\right\}.$

Proof. See Appendix A. ∎

Theorem 1 tells us that a VAE's r-robustness probability can be bounded in terms of: r; the Lipschitz constants of the encoder and decoder; the norm of the encoder standard deviation; the dimension of the latent space; and the norm of the input perturbation. The latter is most important, as it allows us to link the magnitude of input perturbations to the probability of distortions in reconstructions.

The proof leverages the Lipschitz continuity of the decoder network to relate the distances between reconstructed points in $\mathcal{X}$ to the corresponding distances between their latents in $\mathcal{Z}$. The Lipschitz continuity of the encoder then allows the distribution of distances between samples in latent space — from perturbed and unperturbed posteriors $q_\phi(\mathbf{z}|\mathbf{x}+\boldsymbol{\delta})$ and $q_\phi(\mathbf{z}|\mathbf{x})$ respectively — to be characterized in terms of distances between VAE inputs.

We note that the distribution of ℓ_2 distances between these samples is a generalized χ^2 distribution, which has no closed-form CDF (Liu et al., 2009). The proof therefore employs two tail bounds, Markov's Inequality and a tail bound for standard χ^2 distributions.

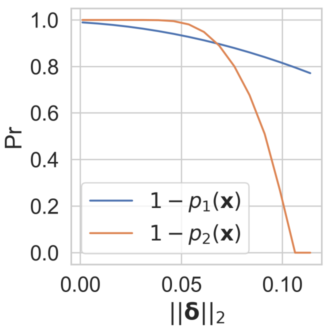


Figure 2: An example of the relative tightness of the bounds in Theorem 1, for $a = b = c = r = d_z = 5$ and fixed $||\sigma_\phi(\mathbf{x})||_2 = 0.1$.

These varyingly dominate each other in tightness for different $||\boldsymbol{\delta}||_2$ (see Figure 2 for a demonstration) and respectively yield $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$.

3.2 Bounding The (r, ϵ)-Robustness Margin

While Theorem 1 allows for the r-robustness probability of a VAE to be lower-bounded for a given input and input perturbation, ideally we would like to guarantee a VAE's robustness at a given input to *all* input perturbations up to some magnitude (for a given ϵ). The following result provides this guarantee, in terms of a lower bound on the (r, ϵ)-robustness margin.

Lemma 2 (Margin Bound)**.** *Given the assumptions of Theorem 1 and some $\epsilon \in [0, 1)$, the (r, ϵ)-robustness margin of this VAE on input $\mathbf{x}$,*

$$R^{(r,\epsilon)}(\mathbf{x}) \geq \max\left\{m_1(\mathbf{x}), m_2(\mathbf{x})\right\}$$

for

$$\begin{aligned} m_1(\mathbf{x}) &:= \frac{-4c||\sigma_\phi(\mathbf{x})||_2 + \sqrt{\Delta}}{2\left(c^2 + b^2\right)}, \\ \Delta &:= (4c||\sigma_\phi(\mathbf{x})||_2)^2 \\ &\quad - 4\left(c^2 + b^2\right)\left(4||\sigma_\phi(\mathbf{x})||_2 - (1 - \epsilon)\left(\frac{r}{a}\right)^2\right), \end{aligned}$$

and $m_2(\mathbf{x}) := \sup\left\{||\boldsymbol{\delta}||_2 : p_2(\boldsymbol{\delta}, \mathbf{x}) \leq (1 - \epsilon)\right\}$, *where* $p_2(\boldsymbol{\delta}, \mathbf{x})$ *is as in Theorem 1.*[4]

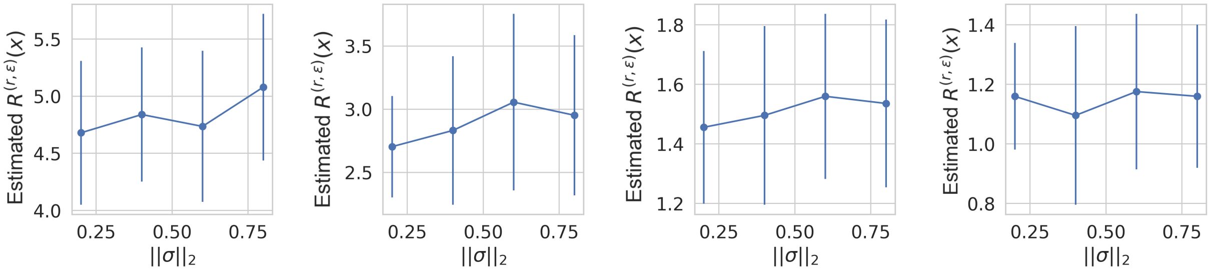
Proof. See Appendix A. ∎

Lemma 2 shows that we can lower bound the radius $R^{(r,\epsilon)}$ about $\mathbf{x}$ within which no input perturbation can undermine (r, ϵ)-robustness; when at least one of $m_1(\mathbf{x})$ and $m_2(\mathbf{x})$ is positive, robustness can be certified. The proof exploits the relationship — established

critical to the robustness of a downstream task.

[3] We assume $\mathcal{X}$ to be Euclidean.

[4] We augment the listed arguments of p_2 to make explicit the dependence on $\boldsymbol{\delta}$.



(a) $M = 5$ (b) $M = 7$ (c) $M = 10$ (d) $M = 12$

Figure 3: Estimated (r, ϵ)-robustness margins (see Appendix D) plotted against the encoder standard deviation norm on MNIST, for fixed r and ϵ and hyperparameter $||\boldsymbol{\sigma}||_2$ as in Theorem 3. Across Lipschitz constants (denoted by M), $||\boldsymbol{\sigma}||_2$ has minimal influence on the estimated robustness margin relative to the choice of Lipschitz constant (compare the ranges across plots). Error bars are the standard deviation over 25 data points.

in Theorem 1 — between the r-robustness probability and the magnitude of input perturbations, finding the largest input perturbation norm such that our lower bound on the r-robustness probability still exceeds ϵ.

3.3 A Global (r, ϵ)-Robustness Margin

We now extend Lemma 2 to provide a *global* margin, which requires bounding $R^{(r,\epsilon)}(\mathbf{x})$ from below for all $\mathbf{x} \in \mathcal{X}$. This can be done either by upper-bounding the encoder standard deviation, since the lower bound on the (r, ϵ)-robustness margin from Lemma 2 is monotonically decreasing in $||\sigma_\phi(\mathbf{x})||_2$, or by lifting the input dependence entirely, by letting $\sigma_\phi(\mathbf{x}) = \boldsymbol{\sigma} \in \mathbb{R}^{d_z}_{\geq 0}$ be a hyperparameter. Since the derivation when using an upper-bound is equivalent to setting the encoder standard deviation as a hyperparameter, we focus on the latter. Fixing the encoder standard deviation can be done either during training — since VAEs can be trained with a fixed encoder standard deviation without serious degradation in performance (Ghosh et al., 2020) — or afterwards, since all that matters to the bound is the value of $\boldsymbol{\sigma}$ at test time.

Theorem 3 (Global Margin Bound)**.** *Given the assumptions of Lemma 2, but with $\sigma_\phi(\mathbf{x}) = \boldsymbol{\sigma} \in \mathbb{R}^{d_z}_{\geq 0}$, the (r, ϵ)-robustness margin of this VAE for all inputs is*

$$R^{(r,\epsilon)} \geq \max\{m_1, m_2\}$$

for

$$m_1 := \frac{\sqrt{-\left(4||\boldsymbol{\sigma}||_2^2 - (1-\epsilon)\left(\frac{r}{a}\right)^2\right)}}{b}$$

and $m_2 := \sup\{||\boldsymbol{\delta}||_2 : p_2(\boldsymbol{\delta}) \leq (1-\epsilon)\}$, where p_2 is as in Theorem 1, but $u := \frac{\left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)^2}{4||\boldsymbol{\sigma}||_2^2}$.

Proof. See Appendix A. ■

This result provides guarantees solely in terms of parameters we can choose *ahead* of training, namely the Lipschitz constants of the networks and $\boldsymbol{\sigma}$, the fixed value of the encoder standard deviation. This importantly distinguishes ours from previous work, which has only provided robustness bounds based on intractable model characteristics that must be empirically estimated *after training* (Camuto et al., 2020).

Note that to further investigate the impact of using fixed encoder standard deviations, we train VAEs with the encoder standard deviation set as a hyperparameter. As shown in Figure 3, we find the Lipschitz constants of the encoder and decoder networks to be most determinative for robustness, with the value of $||\boldsymbol{\sigma}||_2$ being of lesser importance.

4 IMPLEMENTATION

In the last section, we introduced guarantees on robustness assuming the Lipschitz constants of a VAE's networks. We now consider how to train a VAE in a manner that ensures these guarantees are met.

Letting $\mathcal{F}$ be the set of functions that can be learned by an unrestricted neural network, and $\mathcal{L}_M \subset \mathcal{F}$ be the (further restricted) subset of M-Lipschitz continuous functions associated with the sets of neural network parameters $\mathcal{L}_M^\theta, \mathcal{L}_M^\phi$, our constraint can be thought of simply as replacing the standard VAE objective in (1) with the modified objective

$$\underset{\theta,\phi \in \mathcal{L}_M^\theta, \mathcal{L}_M^\phi}{\arg\max} \quad \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) || p(\mathbf{z})\right).$$

Referring to VAEs trained this way as *Lipschitz-VAEs*, the question becomes how to enforce this objective. Using Anil et al. (2019), we focus on fully-connected networks, although similar ideas extend to other ar-

Algorithm 1 The forward pass in a Lipschitz-VAE's encoder or decoder network.

```
BjörckOrthonormalize
    for $k = 1, \ldots, K$ do
        $\mathbf{W}_l^{(k+1)} \leftarrow$
        $\mathbf{W}_l^{(k)} \left( I + \frac{1}{2} Q^{(k)} + \ldots + (-1)^p \binom{-0.5}{p} (Q^{(k)})^p \right)$
        where $Q^{(k)} := I - \left( \mathbf{W}_l^{(k)} \right)^\intercal \mathbf{W}_l^{(k)}$, and $K$ and
        $p$ are hyperparameters.
Input: Data point $\mathbf{x}$
Result: Network output $\mathbf{h}_L$
Requires: Lipschitz constant $M$
Forward pass
    $\mathbf{h}_0 \leftarrow \mathbf{x}$
    for $l = 1, \ldots, L$ do
        $\mathbf{W}_l \leftarrow$ BjörckOrthonormalize($\mathbf{W}_l$)
        pre-activation $\leftarrow M^{\frac{1}{L}} \mathbf{W}_l \mathbf{h}_{l-1}$
        $\mathbf{h}_l \leftarrow$ GroupSort(pre-activation)
```

chitectures (Li et al., 2019). First, note that if layer l has Lipschitz constant M_l, then the Lipschitz constant of the entire network is $M = \prod_{l=1}^{L} M_l$ (Szegedy et al., 2013). For an L-layer fully-connected neural network to be M-Lipschitz, it thus suffices to ensure that each layer has Lipschitz constant $M^{\frac{1}{L}}$. If we choose the network non-linearity $\varphi_l(\cdot)$ to be 1-Lipschitz, and ensure that linear transformation $\mathbf{W}_l$ is also 1-Lipschitz, then a Lipschitz constant of $M^{\frac{1}{L}}$ in each layer follows from scaling the outputs of each layer by $M^{\frac{1}{L}}$.

Building on this, our approach to controlling the Lipschitz continuity of VAE encoders and decoders can be seen in Algorithm 1. The key components are Björck Orthonormalization, which ensures each layer's linear transformation is 1-Lipschitz, and the GroupSort non-linearity from Anil et al. (2019), which is also 1-Lipschitz. See Appendix B for more details.

5 RELATED WORK

Certifiable Robustification Prior work on robustifying models to adversarial attacks can be delineated into a) techniques providing robustness to known types of attack empirically, and b) certifiable techniques providing provable robustness under assumptions. Cohen et al. (2019) argues certifiable techniques should be favored since empirical findings of robustness are predicated on a choice of attack and thus cannot indicate effectiveness against other known or as yet unknown attacks. Indeed, we previously noted instances where empirically-led techniques seemed to induce robustness but were subsequently undone by later-developed attacks (Athalye et al., 2018; Uesato et al., 2018).

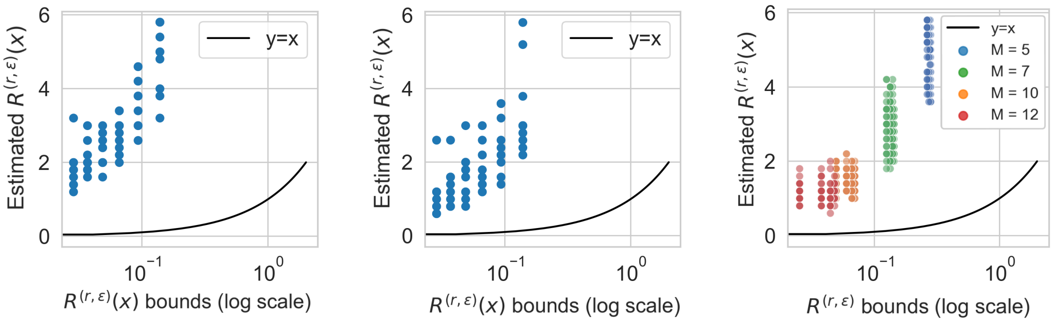
Certifiable Robustness In Classifiers Given their advantages, certifiable robustification techniques have already been developed for classifiers, where approaches employing Lipschitz continuity are particularly illustrative. In particular, Hein and Andriushchenko (2017); Tsuzuku et al. (2018); Anil et al. (2019); Yang et al. (2020) use Lipschitz continuity to provide certified robustness margins for classifiers. We note, however, that in that setting one does not need to handle the probabilistic aspects and continuous changes that one finds in VAEs.

Robustness In VAEs Willetts et al. (2021) argues that the susceptibility of a VAE to adversarial perturbations depends on how much the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ can be changed through changes in input $\mathbf{x}$, and how much reconstruction $g_\theta(\mathbf{z})$ can be changed through changes in the latent variable $\mathbf{z}$. Cemgil et al. (2020a) similarly holds that adversarial examples are possible in VAEs due to "non-smoothness" in the encoding-decoding process, relating this to dissimilarity between a VAE's reconstructions of its reconstructions. Willetts et al. (2021) targets greater smoothness heuristically, controlling the noisiness of the VAE encoding process so that "nearby" inputs correspond to "nearby" latent variables and changes in $q_\phi(\mathbf{z}|\cdot)$ induced by input perturbations have little effect on reconstructions $g_\theta(\mathbf{z})$. Separately, Camuto et al. (2020) proposes (r, ϵ)-robustness and obtains an approximate bound on the (r, ϵ)-robustness margin, allowing the robustness of VAEs to be assessed. That work assumes, however, that input perturbations only affect the encoder mean and not its standard deviation. That work also only allows for the assessment of the robustness of *already trained* VAEs, and unlike our methods does not directly enforce *guaranteed* robustness.

6 EXPERIMENTS

Our aim now is to establish that our theoretical results allow certifying and guaranteeing VAE robustness in practice. We would also like to verify that Lipschitz continuity constraints can endow VAEs with greater robustness to adversarial inputs than standard VAEs.

Experimental Setup We pick a latent space with dimension $d_z = 10$ (unless otherwise stated) and use the same architecture across experiments: encoder mean $\mu_\phi(\cdot)$, encoder standard deviation $\sigma_\phi(\cdot)$ and deterministic component of the decoder $g_\theta(\cdot)$ are all three-layer fully-connected networks with hidden dimensions 512 (for more details, see Appendix F). Following Anil et al. (2019), we start with $K = 3$ Björck Orthonormalization iterations before setting $K = 50$ to finetune to convergence (using $p = 1$ throughout).



(a) Lemma 2: MNIST (b) Lemma 2: Fashion-MNIST (c) Theorem 3: MNIST

Figure 4: Estimated (r, ϵ)-robustness margins plotted against the lower bounds on these margins from Lemma 2, for networks trained on **[Left]** MNIST and **[Center]** Fashion-MNIST. **[Right]** The same plot for the bound in Theorem 3 on MNIST, for fixed $||\boldsymbol{\sigma}||_2 \in \{0.06, 0.13, 0.19, 0.25\}$ and Lipschitz constants $M \in \{5, 7, 10, 12\}$. We plot $y = x$ to illustrate the correctness of the bounds, and use $r = 8$ and $\epsilon = 0.5$ throughout.

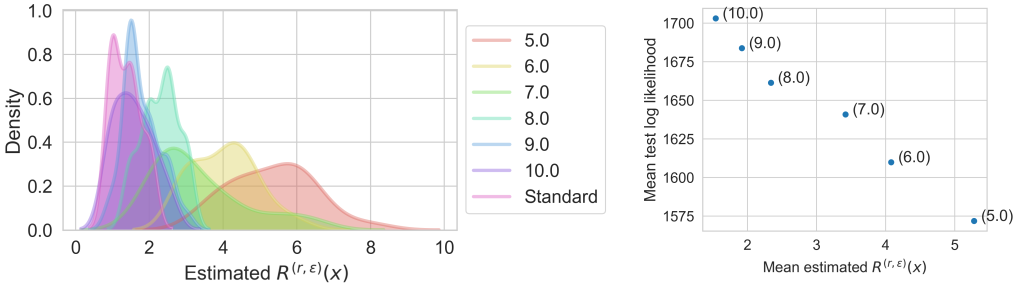
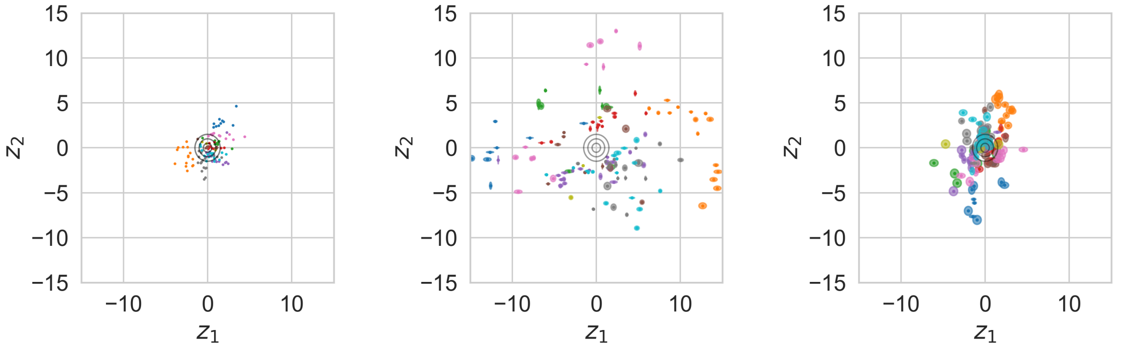


Figure 5: **[Left]** (r, ϵ)-robustness margins $R^{(r,\epsilon)}(\mathbf{x})$ estimated using maximum damage attacks on a randomly-selected collection of MNIST data points in Lipschitz- and standard VAEs, for $r = 8$, $\epsilon = 0.5$, and $||\boldsymbol{\sigma}||_2 = 0.1$. For all Lipschitz constants considered, Lipschitz-VAEs exhibit larger (r, ϵ)-robustness margins on average than a standard VAE, demonstrating the empirical robustness of Lipschitz-VAEs. Larger (r, ϵ)-robustness margins also correlate with smaller Lipschitz constants, as predicted by our bounds. **[Right]** The empirical relationship between a Lipschitz-VAE's reconstruction performance, measured by the mean log likelihood of reconstructions on the MNIST test set, and its mean robustness margin, by Lipschitz constant (in parentheses).

Validating Certifiable Robustness As a sanity check, we first empirically validate that our bounds in Lemma 2 and Theorem 3 allow us to provide the advertised absolute robustness guarantees. Namely, for a given r, ϵ, and Lipschitz-VAE, we compute $\max\{m_1(\mathbf{x}), m_2(\mathbf{x})\}$ and $\max\{m_1, m_2\}$, for Lemma 2 and Theorem 3 respectively, on a randomly-selected sample from MNIST and Fashion-MNIST (see Figure 4).

This experiment empirically validates our bounds, since in all instances the estimated (r, ϵ)-robustness margins (see the following section for estimation) are larger than our corresponding theoretical bounds on these margins. We also see that the bounds on the (r, ϵ)-robustness margin are strictly positive, providing a priori guarantees of robustness when choosing a fixed (or bounded) encoder standard deviation and encoder and decoder Lipschitz constants as in Theorem 3. Our results demonstrate the existence of Lipschitz-VAEs for which meaningful robustness can be certified, a priori.

Though Figure 4 suggests our bounds may often be relatively loose, this is very much consistent with applications of Lipschitz continuity constraints in other settings (Cohen et al., 2019). This looseness is perhaps unavoidable, since an a priori theoretical guarantee of robustness is an extremely strong requirement. As such, our approach is useful in scenarios where robustness must be absolutely guaranteed, even if at times the level of robustness that is guaranteed is lower than the level observed in practice.



(a) Standard VAE

(b) Lipschitz-VAE, $M = 10$

(c) Lipschitz-β-VAE, $M = 10$, β=5

Figure 6: Learned encodings for different types of VAE on MNIST. A colored ellipse represents the posterior $q_\phi(\mathbf{z}|\mathbf{x}_i)$ for a single $\mathbf{x}_i$. The prior, $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$, is overlaid in black for one, two and three standard deviations. Lipschitz-VAEs have encoders that are dispersed in latent space, in contrast with the learned encoder of a standard VAE. Upweighting the KL term in (1), as in a β-VAE (Higgins et al., 2017a), changes this behaviour.

Comparing Empirical Robustness We next empirically assess the (r, ϵ)-robustness margins of Lipschitz-VAEs using the approach of Camuto et al. (2020), leveraging maximum damage attacks (see Appendix D). Assuming no defects in the optimization of (4) and access to infinite samples from the encoder, if a maximum damage attack cannot identify a $\boldsymbol{\delta}^* \leq c$ such that $\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}^*}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}^*})||_2 \leq r\right] \leq \epsilon$, then we can rest assured that the (r, ϵ)-robustness margin of the VAE on input $\mathbf{x}$ is at least c. If (for fixed r and ϵ) one VAE's estimated (r, ϵ)-robustness margins are consistently larger than another's, this strongly suggests that the former is more robust.

In Figure 5 [left], we estimate the (r, ϵ)-robustness margins of several Lipschitz- and standard VAEs on a randomly-selected collection of data points from MNIST. On the same inputs, and for all Lipschitz constants considered, Lipschitz-VAEs exhibit larger estimated (r, ϵ)-robustness margins on average. Figure 5 [left] also validates an implication of our theory, namely that a VAE's (r, ϵ)-robustness margins should monotonically increase as we decrease its Lipschitz constants.

We thus demonstrate that we can manipulate the robustness levels of Lipschitz-VAEs through their Lipschitz constants, fulfilling our objective to develop a VAE whose robustness levels can be controlled *a priori*. Note that, using a unit Gaussian prior, we find the useful range of Lipschitz constants for all networks considered to be between about five and ten: less than this reconstructive performance is excessively impacted, while greater than this Lipschitz-VAEs exhibit robustness comparable to standard VAEs.

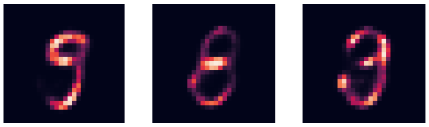


Figure 7: Sample generations from a 10-Lipschitz VAE decoder using noise from a unit Gaussian prior.

Choosing Lipschitz Constants Previously, we saw that the (r, ϵ)-robustness margins of a Lipschitz-VAE could be manipulated through its Lipschitz constants, with smaller Lipschitz constants consistently affording greater robustness. In practice, however, robustness might only be one consideration, alongside reconstruction performance, in choosing between VAEs.

To explore these considerations, we plot reconstruction performance against estimated robustness in Figure 5 [right], measuring reconstruction performance as the mean log likelihood achieved, and estimating robustness in terms of $R^{(r,\epsilon)}(\mathbf{x})$. Recalling that larger log likelihoods imply better reconstructions, we see that reconstruction performance is negatively correlated with estimated robustness, with behavior on each dimension determined by the Lipschitz constants.

Investigating Learned Latent Spaces We now empirically examine whether Lipschitz-VAEs are qualitatively different from standard VAEs in aspects other than robustness, in particular in the latent spaces they learn. As shown in Figure 6, the aggregate posteriors learned by Lipschitz- and standard VAEs differ in their scale. The aggregate posterior of a standard VAE is tightly clustered about the prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$, but that of a Lipschitz-VAE disperses mass more widely over the latent space.

Though this could be an issue when generating samples from the prior, as the prior and aggregate posterior have little overlap, the remedy to this issue is quite simple. We find that upweighting the KL term in the VAE objective by hyperparameter β, as in a β-VAE (Higgins et al., 2017a), mitigates this scaling of the latent space (see Figure 6c). For details, see Appendix C; sample generations can also be seen in Figure 7.

7 CONCLUSION

We have introduced an approach to training VAEs that allows their robustness to adversarial attacks to be guaranteed *a priori*. Specifically, we derived provable bounds on the degree of robustness of a VAE under input perturbation, with these bounds depending on parameters such as the Lipschitz constants of its encoder and decoder networks. We then showed how these parameters can be controlled, enabling our bounds to be invoked in practice and providing an actionable way of ensuring the robustness of a VAE ahead of training.

References

Anil, C., Lucas, J., and Grosse, R. (2019). Sorting Out Lipschitz Function Approximation. In *International Conference on Machine Learning*, pages 291–301.

Athalye, A., Carlini, N., and Wagner, D. (2018). Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. *arXiv preprint arXiv:1802.00420*.

Camuto, A., Willetts, M., Roberts, S., Holmes, C., and Rainforth, T. (2020). Towards a Theoretical Understanding of The Robustness of Variational Autoencoders. *arXiv preprint arXiv:2007.07365*.

Cemgil, T., Ghaisas, S., Dvijotham, K., Gowal, S., and Kohli, P. (2020a). The Autoencoding Variational Autoencoder. In *Advances in Neural Information Processing Systems*.

Cemgil, T., Ghaisas, S., Dvijotham, K., and Kohli, P. (2020b). Adversarially Robust Representations with Smooth Encoders. In *International Conference on Learning Representations*.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. (2019). Certified Adversarial Robustness via Randomized Smoothing. In *International Conference on Machine Learning*.

Ghosh, P., Losalka, A., and Black, M. J. (2019). Resisting Adversarial Attacks using Gaussian Mixture Variational Autoencoders. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:541–548.

Ghosh, P., Sajjadi, M. S. M., Vergari, A., Black, M., and Schölkopf, B. (2020). From Variational to Deterministic Autoencoders. In *International Conference on Learning Representations*.

Gondim-Ribeiro, G., Tabacof, P., and Valle, E. (2018). Adversarial Attacks on Variational Autoencoders. *arXiv preprint arXiv:1806.04646*.

Ha, D. and Schmidhuber, J. (2018). World Models. *arXiv preprint arXiv:1803.10122*, abs/1803.10122.

Hein, M. and Andriushchenko, M. (2017). Formal Guarantees on the Robustness of a Classifier Against Adversarial Manipulation. In *Advances in Neural Information Processing Systems*, pages 2266–2276.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017a). β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *ICLR*.

Higgins, I., Pal, A., Rusu, A., Matthey, L., Burgess, C., Pritzel, A., Botvinick, M., Blundell, C., and Lerchner, A. (2017b). Darla: Improving Zero-Shot Transfer in Reinforcement Learning. In *International Conference on Machine Learning*, pages 1480–1490. PMLR.

Huster, T., Chiang, C.-Y. J., and Chadha, R. (2019). Limitations of the Lipschitz Constant as a Defense Against Adversarial Examples. *Lecture Notes in Computer Science*, page 16–29.

Inglot, T. (2010). Inequalities for Quantiles of the Chi-Square Distribution. *Probability and Mathematical Statistics*, 30(2):339–351.

Kim, Y., Wiseman, S., Miller, A. C., Sontag, D., and Rush, A. M. (2018). Semi-Amortized Variational Autoencoders. *arXiv preprint arXiv:1802.02550*.

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. *arXiv preprint arXiv:1312.6114*.

Kingma, D. P. and Welling, M. (2019). An Introduction to Variational Autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392.

Kos, J., Fischer, I., and Song, D. (2018). Adversarial Examples for Generative Models. *2018 IEEE Security and Privacy Workshops (SPW)*.

Kumar, A. and Poole, B. (2020). On Implicit Regularization in β-VAEs. *arXiv preprint arXiv:2002.00041*.

Li, Q., Haque, S., Anil, C., Lucas, J., Grosse, R. B., and Jacobsen, J.-H. (2019). Preventing Gradient Attenuation in Lipschitz Constrained Convolutional Networks. In *Advances in neural information processing systems*, pages 15390–15402.

Liu, H., Tang, Y., and Zhang, H. H. (2009). A New Chi-Square Approximation to the Distribution of Non-Negative Definite Quadratic Forms in Non-Central Normal Variables. *Computational Statistics & Data Analysis*, 53(4):853–856.

Loaiza-Ganem, G. and Cunningham, J. P. (2019). The Continuous Bernoulli: Fixing a Pervasive Error in Variational Autoencoders. In *Advances in Neural Information Processing Systems*, pages 13287–13297.

Mathieu, E., Rainforth, T., Siddharth, N., and Teh, Y. W. (2019). Disentangling Disentanglement in Variational Autoencoders. In *International Conference on Machine Learning*, pages 4402–4412.

Razavi, A., van den Oord, A., and Vinyals, O. (2019). Generating Diverse High-Fidelity Images with VQ-VAE-2. In *Advances in Neural Information Processing Systems*, pages 14866–14876.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *arXiv preprint arXiv:1401.4082*.

Salman, H., Li, J., Razenshteyn, I., Zhang, P., Zhang, H., Bubeck, S., and Yang, G. (2019). Provably Robust Deep Learning via Adversarially Trained Smoothed Classifiers. In *Advances in Neural Information Processing Systems*, pages 11292–11303.

Schott, L., Rauber, J., Bethge, M., and Brendel, W. (2018). Towards the First Adversarially Robust Neural Network Model on MNIST. *arXiv preprint arXiv:1805.09190*.

Shao, J. (2015). Noncentral Chi-Squared, t- and F-Distributions. Lecture.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing Properties of Neural Networks. *arXiv preprint arXiv:1312.6199*.

Tabacof, P., Tavares, J., and Valle, E. (2016). Adversarial Images for Variational Autoencoders. *arXiv preprint arXiv:1612.00155*.

Tsuzuku, Y., Sato, I., and Sugiyama, M. (2018). Lipschitz-Margin Training: Scalable Certification of Perturbation Invariance for Deep Neural Networks. In *Advances in neural information processing systems*, pages 6541–6550.

Uesato, J., O'Donoghue, B., Oord, A. v. d., and Kohli, P. (2018). Adversarial Risk and the Dangers of Evaluating Against Weak Attacks. *arXiv preprint arXiv:1802.05666*.

Virmaux, A. and Scaman, K. (2018). Lipschitz Regularity of Deep Neural Networks: Analysis and Efficient Estimation. In *Advances in Neural Information Processing Systems*, pages 3835–3844.

Willetts, M., Camuto, A., Rainforth, T., Roberts, S., and Holmes, C. (2021). Improving VAEs' Robustness to Adversarial Attacks. In *International Conference on Learning Representations*.

Yang, Y.-Y., Rashtchian, C., Zhang, H., Salakhutdinov, R., and Chaudhuri, K. (2020). Adversarial Robustness Through Local Lipschitzness. *arXiv preprint arXiv:2003.02460*.

Supplementary Material: Certifiably Robust Variational Autoencoders

A PROOFS

Theorem 1 (Probability Bound)**.** *Assume* $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}; \mu_\phi(\mathbf{x}), \textit{\texttt{diag}}\left(\sigma^2_\phi(\mathbf{x})\right)\right)$ *and that the deterministic component of the VAE decoder* $g_\theta(\cdot)$ *is a-Lipschitz, the encoder mean* $\mu_\phi(\cdot)$ *is b-Lipschitz, and the encoder standard deviation* $\sigma_\phi(\cdot)$ *is c-Lipschitz. Finally, let* $\mathbf{z}_{\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x}+\boldsymbol{\delta})$ *and* $\mathbf{z}_{\neg\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x})$*. Then for any* $r \in \mathbb{R}^+$*, any* $\mathbf{x} \in \mathcal{X}$*, and any input perturbation* $\boldsymbol{\delta} \in \mathcal{X}$*,*

$$\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] \geq 1 - \min\left\{p_1(\mathbf{x}), p_2(\mathbf{x})\right\},$$

where

$$p_1(\mathbf{x}) := \min\left(1, \frac{a^2\left(b^2||\boldsymbol{\delta}||_2^2 + (c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2\right)}{r^2}\right)$$

and

$$p_2(\mathbf{x}) := \begin{cases} C(d_z)\frac{u(\mathbf{x})^{\frac{d_z}{2}}\exp\left\{-\frac{u(\mathbf{x})}{2}\right\}}{u(\mathbf{x})-d_z+2} & \left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right) \geq 0; d_z \geq 2; u(\mathbf{x}) > d_z - 2 \\ 1 & o.w. \end{cases}$$

for $u(\mathbf{x}) := \frac{\left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)^2}{(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2}$ *and constant* $C(d_z) := \frac{1}{\sqrt{\pi}}\exp\left\{\frac{1}{2}(d_z - (d_z-1)\log d_z)\right\}$.

Proof. Since $g_\theta(\cdot)$ is a-Lipschitz,

$$||g_\theta(\mathbf{z}_1) - g_\theta(\mathbf{z}_2)||_2 \leq a||\mathbf{z}_1 - \mathbf{z}_2||_2 \tag{5}$$

for all $\mathbf{z}_1, \mathbf{z}_2 \in \mathcal{Z}$.

Now assume $\mathbf{z}_1 \sim q_\phi(\mathbf{z}|\mathbf{x}_1)$ and $\mathbf{z}_2 \sim q_\phi(\mathbf{z}|\mathbf{x}_2)$ for some $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$, such that $g_\theta(\mathbf{z}_1)$ and $g_\theta(\mathbf{z}_2)$ are random variables. (5) then implies

$$\left\{||g_\theta(\mathbf{z}_1) - g_\theta(\mathbf{z}_2)||_2 \leq r\right\} \supseteq \left\{a||\mathbf{z}_1 - \mathbf{z}_2||_2 \leq r\right\},$$

which in turn implies

$$\mathbb{P}\left[||g_\theta(\mathbf{z}_1) - g_\theta(\mathbf{z}_2)||_2 \leq r\right] \geq \mathbb{P}\left[a||\mathbf{z}_1 - \mathbf{z}_2||_2 \leq r\right]. \tag{6}$$

Letting $\mathbf{x}_1 = \mathbf{x} + \boldsymbol{\delta}$ and $\mathbf{x}_2 = \mathbf{x}$ such that $\mathbf{z}_1 = \mathbf{z}_{\boldsymbol{\delta}}$ and $\mathbf{z}_2 = \mathbf{z}_{\neg\boldsymbol{\delta}}$, $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}; \mu_\phi(\mathbf{x}), \texttt{diag}\left(\sigma^2_\phi(\mathbf{x})\right)\right)$ means

$$\mathbf{z}_{\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x}+\boldsymbol{\delta}) = \mathcal{N}\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}), \texttt{diag}\left(\sigma^2_\phi(\mathbf{x}+\boldsymbol{\delta})\right)\right)$$

and

$$\mathbf{z}_{\neg\boldsymbol{\delta}} \sim q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mu_\phi(\mathbf{x}), \texttt{diag}\left(\sigma^2_\phi(\mathbf{x})\right)\right).$$

Further, since samples from $q_\phi(\mathbf{z}|\cdot)$ are drawn independently in every VAE forward pass, we also know $\mathbf{z}_{\boldsymbol{\delta}}$ and $\mathbf{z}_{\neg\boldsymbol{\delta}}$ are independent, and thus, because the difference of independent multivariate Gaussian random variables is multivariate Gaussian,

$$\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} \sim \mathcal{N}\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}), \texttt{diag}\left(\sigma^2_\phi(\mathbf{x}+\boldsymbol{\delta})\right) + \texttt{diag}\left(\sigma^2_\phi(\mathbf{x})\right)\right).$$

Returning to (6), since $||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2$ is a continuous random variable, we can write

$$\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] \geq \mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \leq \frac{r}{a}\right] = 1 - \mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right]. \tag{7}$$

The proof now diverges, yielding $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$ respectively.

Obtaining $p_1(\mathbf{x})$: Recall $\mathcal{Z} = \mathbb{R}^{d_z}$, apply the definition of the ℓ_2 norm, and invoke Markov's Inequality to obtain

$$\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right] = \mathbb{P}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2 \geq \left(\frac{r}{a}\right)^2\right] \leq \frac{\mathbb{E}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2\right]}{\left(\frac{r}{a}\right)^2}. \tag{8}$$

Now note that

$$\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2 = \sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j \frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j},$$

so that by the linearity of expectations,

$$\begin{aligned}
&\mathbb{E}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2\right] \\
&= \mathbb{E}\left[\sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j \frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}\right] \\
&= \sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j \mathbb{E}\left[\frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}\right].
\end{aligned} \tag{9}$$

Because $\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}$ is diagonal-covariance multivariate Gaussian, the $\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j$ are jointly independent for all $j = 1, \ldots, d_z$, and so we recognize that

$$\frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}$$

has a non-central χ^2 distribution with one degree of freedom and non-centrality parameter

$$\frac{\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}.$$

Since for a non-central χ^2 random variable Y with n degrees of freedom and non-centrality parameter γ (Shao, 2015), $\mathbb{E}[Y] = n + \gamma$, we have

$$\mathbb{E}\left[\frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}\right] = 1 + \frac{\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j},$$

and so plugging into (9),

$$\begin{aligned}
&\mathbb{E}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2\right] \\
&= \sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j \left(1 + \frac{\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}\right) \\
&= \sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j + \sum_{j=1}^{d_z} \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2.
\end{aligned}$$

Using

$$\sum_{j=1}^{d_z} \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2 = ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2^2$$

(the definition of the ℓ_2 norm), and

$$||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2 \leq b||\boldsymbol{\delta}||_2,$$

(since $\mu_\phi(\cdot)$ is b-Lipschitz), we obtain

$$\sum_{j=1}^{d_z} \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2 = ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2^2 \leq \left(b||\boldsymbol{\delta}||_2\right)^2 = b^2||\boldsymbol{\delta}||_2^2. \tag{10}$$

Similarly, using

$$\begin{aligned}
&\sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j \\
&\leq \sum_{j=1}^{d_z} \sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta})_j + \sigma_\phi^2(\mathbf{x})_j + 2\sigma_\phi(\mathbf{x}+\boldsymbol{\delta})_j\sigma_\phi(\mathbf{x})_j \\
&= \sum_{j=1}^{d_z} \left(\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi(\mathbf{x})\right)_j^2 \\
&= \left(\sqrt{\sum_{j=1}^{d_z} \left(\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi(\mathbf{x})\right)_j^2}\right)^2 \\
&= ||\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi(\mathbf{x})||_2^2
\end{aligned}$$

(where the above inequality follows from $\sigma_\phi : \mathcal{X} \to \mathbb{R}_{\geq 0}^{d_z}$, and the last equality follows from the definition of the ℓ_2 norm), and

$$\begin{aligned}
&||\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi(\mathbf{x})||_2 \\
&= ||\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) - \sigma_\phi(\mathbf{x}) + 2\sigma_\phi(\mathbf{x})||_2 \\
&\leq ||\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) - \sigma_\phi(\mathbf{x})||_2 + 2||\sigma_\phi(\mathbf{x})||_2 \\
&\leq c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2
\end{aligned}$$

(where the first inequality follows by the triangle inequality, and the second follows from the assumption that $\sigma_\phi(\cdot)$ is c-Lipschitz), we find

$$\sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j \leq ||\sigma_\phi(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi(\mathbf{x})||_2^2 \leq \left(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2\right)^2. \tag{11}$$

Hence, returning to (8), we see

$$\begin{aligned}
&\frac{\mathbb{E}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2\right]}{\left(\frac{r}{a}\right)^2} \\
&= \frac{\sum_{j=1}^{d_z} \left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j + \sum_{j=1}^{d_z} \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2}{\left(\frac{r}{a}\right)^2} \\
&\leq \frac{b^2||\boldsymbol{\delta}||_2^2 + \left(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2\right)^2}{\left(\frac{r}{a}\right)^2} \\
&= \frac{a^2\left(b^2||\boldsymbol{\delta}||_2^2 + (c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2\right)}{r^2},
\end{aligned}$$

such that

$$\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right] \leq \frac{\mathbb{E}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2\right]}{\left(\frac{r}{a}\right)^2} \leq \frac{a^2\left(b^2||\boldsymbol{\delta}||_2^2 + (c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2\right)}{r^2}.$$

Noting that the right-most term is non-negative, and wanting to have a well-defined probability, we take

$$p_1(\mathbf{x}) := \min\left(1, \frac{a^2\left(b^2||\boldsymbol{\delta}||_2^2 + (c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2)^2\right)}{r^2}\right),$$

such that

$$\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right] \leq p_1(\mathbf{x}).$$

Obtaining $p_2(\mathbf{x})$: Return to (7). By the triangle inequality,

$$||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \leq ||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)||_2 + ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2,$$

and hence

$$\begin{aligned} &\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right] && (12)\\ &\leq \mathbb{P}\left[\left(||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)||_2 + ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right) \geq \frac{r}{a}\right] \\ &= \mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)||_2 \geq \left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right)\right]. \end{aligned}$$

Then, again recalling $\mathcal{Z} = \mathbb{R}^{d_z}$,

$$\begin{aligned} &\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - \left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)||_2 \geq \left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right)\right] && (13)\\ &= \mathbb{P}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - (\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}))\right)_j^2 \geq \left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right)^2\right] \\ &\leq \mathbb{P}\left[\sum_{j=1}^{d_z} \frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - (\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}))\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j} \geq \frac{\left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right)^2}{\left(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2\right)^2}\right], && (14) \end{aligned}$$

where the first equality uses the definition of the ℓ_2 norm, and the above inequality between probabilities uses the inequality from (11).

Now, since

$$\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} \sim \mathcal{N}\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}), \mathtt{diag}\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta})\right) + \mathtt{diag}\left(\sigma_\phi^2(\mathbf{x})\right)\right),$$

it follows that

$$\frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - (\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}))\right)_j}{\sqrt{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}} \sim \mathcal{N}(0,1).$$

In particular, note that since $\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}$ is diagonal-covariance multivariate Gaussian, the

$$\frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - (\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}))\right)_j}{\sqrt{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j}}$$

are jointly independent for all $j = 1, \ldots, d_z$. Hence, because the sum of squares of d_z independent standard Gaussian random variables has a standard χ^2 distribution with d_z degrees of freedom,

$$\sum_{j=1}^{d_z} \frac{\left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}} - (\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x}))\right)_j^2}{\left(\sigma_\phi^2(\mathbf{x}+\boldsymbol{\delta}) + \sigma_\phi^2(\mathbf{x})\right)_j} =: Y \sim \chi_{d_z}^2.$$

Letting

$$u'(\mathbf{x}) := \frac{\left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right)^2}{\left(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2\right)^2} \quad \text{and} \quad u(\mathbf{x}) := \frac{\left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)^2}{\left(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2\right)^2},$$

we have $u'(\mathbf{x}) \geq u(\mathbf{x})$ by the assumption that $\mu_\phi(\cdot)$ is b-Lipschitz, since

$$||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2 \leq b||\boldsymbol{\delta}||_2,$$

and therefore

$$\left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right) \geq \left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)$$

(note also that $\left(c||\boldsymbol{\delta}||_2 + 2||\sigma_\phi(\mathbf{x})||_2\right)^2 \geq 0$). Then, using (14) with the requirement that

$$\left(\frac{r}{a} - ||\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})||_2\right) \geq \left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right) \geq 0$$

to ensure the inequality in (13) is meaningful,

$$\mathbb{P}\left[Y \geq u'(\mathbf{x})\right] \leq \mathbb{P}\left[Y \geq u(\mathbf{x})\right].$$

The tail bound for standard χ^2 random variables in (3.1) from Inglot (2010) (which requires $u(\mathbf{x}) > d_z - 2$ and $d_z \geq 2$) then yields

$$\mathbb{P}\left[Y \geq u(\mathbf{x})\right] \leq C(d_z)\frac{u(\mathbf{x})^{\frac{d_z}{2}} \exp\left\{-\frac{u(\mathbf{x})}{2}\right\}}{u(\mathbf{x}) - d_z + 2}$$

for constant $C(d_z) := \frac{1}{\sqrt{\pi}} \exp\left\{\frac{1}{2}(d_z - (d_z - 1)\log d_z)\right\}$. Since the expression on the right-hand side is non-negative under the above conditions, we define

$$p_2(\mathbf{x}) := \begin{cases} C(d_z)\frac{u(\mathbf{x})^{\frac{d_z}{2}} \exp\left\{-\frac{u(\mathbf{x})}{2}\right\}}{u(\mathbf{x}) - d_z + 2} & \left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right) \geq 0; d_z \geq 2; u(\mathbf{x}) > d_z - 2 \\ 1 & \text{o.w.} \end{cases}$$

to ensure a well-defined probability. Then, by the inequalities starting from (12),

$$\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right] \leq p_2(\mathbf{x}).$$

Obtaining the final bound: Choosing the least of $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$ to obtain the tighter upper bound on $\mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right]$, we can plug in to (7), which gives

$$\begin{aligned} &\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] \\ &\geq 1 - \mathbb{P}\left[||\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}||_2 \geq \frac{r}{a}\right] \\ &\geq 1 - \min\{p_1(\mathbf{x}), p_2(\mathbf{x})\}. \end{aligned}$$

■

Lemma 2 (Margin Bound)**.** *Given the assumptions of Theorem 1 and some $\epsilon \in [0,1)$, the (r,ϵ)-robustness margin of this VAE on input $\mathbf{x}$,*

$$R^{(r,\epsilon)}(\mathbf{x}) \geq \max\left\{m_1(\mathbf{x}), m_2(\mathbf{x})\right\}$$

for

$$m_1(\mathbf{x}) := \frac{-4c||\sigma_\phi(\mathbf{x})||_2 + \sqrt{\left(4c||\sigma_\phi(\mathbf{x})||_2\right)^2 - 4\left(c^2+b^2\right)\left(4||\sigma_\phi(\mathbf{x})||_2 - (1-\epsilon)\left(\frac{r}{a}\right)^2\right)}}{2\left(c^2+b^2\right)}$$

and $m_2(\mathbf{x}) := \sup\left\{||\boldsymbol{\delta}||_2 : p_2(\boldsymbol{\delta}, \mathbf{x}) \leq (1-\epsilon)\right\}$, where $p_2(\boldsymbol{\delta}, \mathbf{x})$ is as in Theorem 1 and we augment the listed arguments of p_2 to make explicit the dependence on $\boldsymbol{\delta}$.

Proof. By Theorem 1, for any input perturbation $\boldsymbol{\delta} \in \mathcal{X}$ and any input $\mathbf{x} \in \mathcal{X}$,

$$\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] \geq 1 - \min\{p_1(\mathbf{x}), p_2(\mathbf{x})\}.$$

Hence, for our Lipschitz-VAE to be (r, ϵ)-robust to perturbation $\boldsymbol{\delta}$ on input $\mathbf{x}$ for threshold $\epsilon \in [0, 1)$, by Definition 2.1 it suffices that

$$1 - \min\{p_1(\mathbf{x}), p_2(\mathbf{x})\} > \epsilon.$$

Recalling Definition 2.2, since for a model f, $R^{(r,\epsilon)}(\mathbf{x})$ is defined by

$$||\boldsymbol{\delta}||_2 < R^{(r,\epsilon)}(\mathbf{x}) \implies \mathbb{P}\left[||f(\mathbf{x}+\boldsymbol{\delta}) - f(\mathbf{x})||_2 \leq r\right] > \epsilon,$$

for our Lipschitz-VAE $R^{(r,\epsilon)}(\mathbf{x})$ is at least the maximum perturbation norm such that

$$1 - \min\{p_1(\boldsymbol{\delta}, \mathbf{x}), p_2(\boldsymbol{\delta}, \mathbf{x})\} \geq \epsilon,$$

or equivalently,

$$\max\left\{\sup\left\{||\boldsymbol{\delta}||_2 : p_1(\boldsymbol{\delta}, \mathbf{x}) \leq (1-\epsilon)\right\},\ \sup\left\{||\boldsymbol{\delta}||_2 : p_2(\boldsymbol{\delta}, \mathbf{x}) \leq (1-\epsilon)\right\}\right\}$$

(where we make explicit the dependence on $\boldsymbol{\delta}$).

Denoting $m_1(\mathbf{x}) := \sup\left\{||\boldsymbol{\delta}||_2 : p_1(\boldsymbol{\delta}, \mathbf{x}) \leq (1-\epsilon)\right\}$ and rearranging, $m_1(\mathbf{x})$ becomes

$$\sup\left\{||\boldsymbol{\delta}||_2 : \left(c^2 + b^2\right)||\boldsymbol{\delta}||_2^2 + 4c||\sigma_\phi(\mathbf{x})||_2||\boldsymbol{\delta}||_2 + 4||\sigma_\phi(\mathbf{x})||_2^2 - (1-\epsilon)\left(\frac{r}{a}\right)^2 \leq 0\right\}.$$

Excluding the degenerate case of $c = 0$, that is assuming $c > 0$, this is attained at the maximum root of the quadratic equation

$$\left(c^2 + b^2\right)||\boldsymbol{\delta}||_2^2 + 4c||\sigma_\phi(\mathbf{x})||_2||\boldsymbol{\delta}||_2 + 4||\sigma_\phi(\mathbf{x})||_2^2 - (1-\epsilon)\left(\frac{r}{a}\right)^2 = 0,$$

provided a root exists, and so by the quadratic formula,

$$m_1(\mathbf{x}) = \frac{-4c||\sigma_\phi(\mathbf{x})||_2 + \sqrt{\left(4c||\sigma_\phi(\mathbf{x})||_2\right)^2 - 4\left(c^2+b^2\right)\left(4||\sigma_\phi(\mathbf{x})||_2 - (1-\epsilon)\left(\frac{r}{a}\right)^2\right)}}{2\left(c^2+b^2\right)}.$$

The second case does not admit a closed-form solution, so we will simply write

$$m_2(\mathbf{x}) := \sup\left\{||\boldsymbol{\delta}||_2 : p_2(\boldsymbol{\delta}, \mathbf{x}) \leq (1-\epsilon)\right\}.$$

Choosing the maximum of $m_1(\mathbf{x})$ and $m_2(\mathbf{x})$ then yields

$$R^{(r,\epsilon)}(\mathbf{x}) \geq \max\left\{m_1(\mathbf{x}), m_2(\mathbf{x})\right\}.$$

■

Theorem 3 (Global Margin Bound)**.** *Given the assumptions of Lemma 2, but with $\sigma_\phi(\mathbf{x}) = \boldsymbol{\sigma} \in \mathbb{R}^{d_z}_{\geq 0}$, the (r, ϵ)-robustness margin of this VAE for all inputs is*

$$R^{(r,\epsilon)} \geq \max\left\{m_1, m_2\right\},$$

where

$$m_1 := \frac{\sqrt{-\left(4||\boldsymbol{\sigma}||_2^2 - (1-\epsilon)\left(\frac{r}{a}\right)^2\right)}}{b}$$

and $m_2 := \sup\left\{||\boldsymbol{\delta}||_2 : p_2(\boldsymbol{\delta}) \leq (1-\epsilon)\right\}$, where p_2 is as in Theorem 1, but $u := \frac{\left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)^2}{4||\boldsymbol{\sigma}||_2^2}$.

Proof. Given a fixed encoder standard deviation, that is substituting $\sigma_\phi(\mathbf{x}) = \boldsymbol{\sigma} \in \mathbb{R}^{d_z}_{\geq 0}$, we first have to derive a lower bound on the r-robustness probability to then bound the (r, ϵ)-robustness margin globally. We do this using the machinery of Theorem 1, which — lifting the now-redundant requirement that the encoder standard deviation be c-Lipschitz — can be invoked without loss of generality.

In the case of p_1 (recall the two bounds in the proof of Theorem 1), plugging in $\boldsymbol{\sigma}$ yields

$$\begin{aligned}\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] &\geq 1 - \frac{\mathbb{E}\left[\sum_{j=1}^{d_z} \left(\mathbf{z}_{\boldsymbol{\delta}} - \mathbf{z}_{\neg\boldsymbol{\delta}}\right)_j^2\right]}{\left(\frac{r}{a}\right)^2} \\ &= 1 - \frac{\sum_{j=1}^{d_z}\left(\boldsymbol{\sigma}^2 + \boldsymbol{\sigma}^2\right)_j + \sum_{j=1}^{d_z}\left(\mu_\phi(\mathbf{x}+\boldsymbol{\delta}) - \mu_\phi(\mathbf{x})\right)_j^2}{\left(\frac{r}{a}\right)^2} \\ &\geq 1 - \frac{b^2||\boldsymbol{\delta}||_2^2 + 4||\boldsymbol{\sigma}||_2^2}{\left(\frac{r}{a}\right)^2} \\ &= 1 - p_1\end{aligned}$$

for $p_1 := \frac{a^2\left(b^2||\boldsymbol{\delta}||_2^2 + 4||\boldsymbol{\sigma}||_2^2\right)}{r^2}$, where the penultimate step follows by (10) and (11). In the case of p_2, we can directly substitute, obtaining

$$\mathbb{P}\left[||g_\theta(\mathbf{z}_{\boldsymbol{\delta}}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}})||_2 \leq r\right] \geq 1 - p_2$$

for

$$p_2 := \begin{cases} C(d_z)\frac{u^{\frac{d_z}{2}}\exp\left\{-\frac{u}{2}\right\}}{u - d_z + 2} & \left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right) \geq 0; d_z \geq 2; u > d_z - 2 \\ 1 & \text{o.w.}\end{cases}$$

and $u := \frac{\left(\frac{r}{a} - b||\boldsymbol{\delta}||_2\right)^2}{4||\boldsymbol{\sigma}||_2^2}$. Theorem 3 then follows by identical reasoning to Lemma 2. ■

B IMPLEMENTING CERTIFIABLY ROBUST VAES

Ensuring the Lipschitz continuity of a deep learning architecture is non-trivial in practice. Using Anil et al. (2019) as a guide, this section elaborates on how to provably control the Lipschitz constants of an encoder and decoder network.[5]

We define a fully-connected network with L layers as the composition of linear transformations $\mathbf{W}_l$ and element-wise activation functions $\varphi_l(\cdot)$ for $l = 1, \ldots, L$, where the output of the l-th layer

$$\mathbf{h}_l := \varphi_l(\mathbf{W}_l\mathbf{h}_{l-1}).$$

B.1 Ensuring Lipschitz Continuity With Constant 1

We would like to ensure a fully-connected network is M-Lipschitz for arbitrary Lipschitz constant M. It has been shown that a natural way to achieve this is by first requiring Lipschitz continuity with constant 1 (Anil et al., 2019).

As 1-Lipschitz functions are closed under composition, if we can ensure that for every layer l, $\mathbf{W}_l$ and $\varphi_l(\cdot)$ are 1-Lipschitz, then the entire network will be 1-Lipschitz. Most commonly-used activation functions, such as the ReLU and the sigmoid function, are already 1-Lipschitz (Huster et al., 2019; Virmaux and Scaman, 2018), and hence we need only ensure that $\mathbf{W}_l$ is also 1-Lipschitz.

This can be done by requiring $\mathbf{W}_l$ to be orthonormal, since $\mathbf{W}_l$ being 1-Lipschitz is equivalent to the condition

$$\sup_{||\mathbf{x}||_2 \leq 1} ||\mathbf{W}_l\mathbf{x}||_2 \leq 1, \tag{15}$$

where $\sup_{||\mathbf{x}||_2 \leq 1} ||\mathbf{W}_l\mathbf{x}||_2$ equals the largest singular value of $\mathbf{W}_l$. The singular values of an orthonormal matrix all equal 1, and so the orthonormality of $\mathbf{W}_l$ implies (15) is satisfied.

[5] For simplicity, we focus on fully-connected architectures, although the same ideas extend, for example, to convolutional architectures (Li et al., 2019).

In practice, $\mathbf{W}_l$ can be made orthonormal through an iterative algorithm called *Björck Orthonormalization*, which on input matrix $\mathbf{A}$ finds the "nearest" orthonormal matrix to $\mathbf{A}$ (Anil et al., 2019). Björck Orthonormalization is differentiable and so allows the encoder and decoder networks of a Lipschitz-VAE to be trained using gradient-based methods, just like a standard VAE.

B.2 Ensuring Lipschitz Continuity With Arbitrary Constants

Now that we can train a 1-Lipschitz network, we would like to generalize this method to arbitrary Lipschitz constant M. To do so, note that if layer l has Lipschitz constant M_l, then the Lipschitz constant of the entire network is $M = \prod_{l=1}^{L} M_l$ (Szegedy et al., 2013).

Hence, for our L-layer fully-connected neural network to be M-Lipschitz, it suffices to ensure that each layer l has Lipschitz constant $M^{\frac{1}{L}}$. This is actually simple to achieve, because if we continue to assume $\varphi_l(\cdot)$ is 1-Lipschitz, a Lipschitz constant of $M^{\frac{1}{L}}$ in layer l follows from scaling the outputs of each layer by $M^{\frac{1}{L}}$.

B.3 Selecting Activation Functions

While the above approach is sufficient to train networks with arbitrary Lipschitz constants, a result from Anil et al. (2019) shows it is not sufficient to ensure the resulting networks are also expressive in the space of Lipschitz continuous functions. Informally, the result states that the expressivity of a Lipschitz-constrained network is limited when its activation functions are not gradient norm-preserving. Since non-linearities such as the ReLU and the sigmoid function do not preserve the gradient norm, the expressivity of Lipschitz-constrained networks that use such activations will be further limited.

To address this, Anil et al. (2019) introduces a gradient norm-preserving activation function called *GroupSort*, which in each layer l groups the entries of matrix-vector product $\mathbf{W}_l \mathbf{h}_{l-1}$ into some number of groups, and then sorts the entries of each group by ascending order. It can be shown that when each group has size two,

$$\begin{pmatrix} 1 & 0 \end{pmatrix}^{\intercal} \text{GroupSort}\left(\begin{pmatrix} y \\ 0 \end{pmatrix}\right) = \text{ReLU}(y)$$

for any scalar y (Anil et al., 2019). Unless we need to restrict a network's outputs to a specific range, we employ the GroupSort activation in our implementation of Lipschitz-VAEs.

C INVESTIGATING LEARNED LATENT SPACES

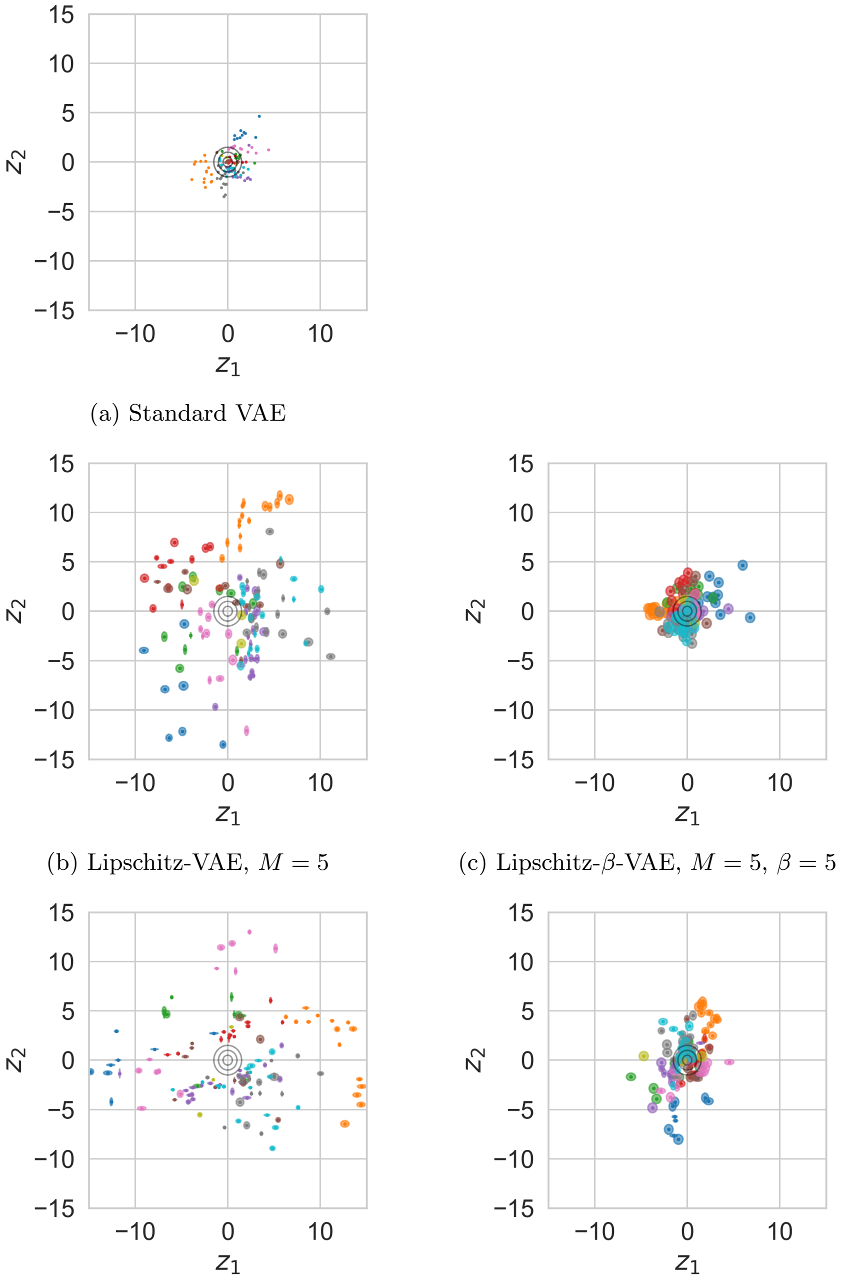
While we are primarily interested in the robustness of VAEs, and their certifiably robust instantiation in Lipschitz-VAEs, we may also wish to understand whether Lipschitz-VAEs qualitatively differ from standard VAEs.

To build our understanding in this regard, we study the latent spaces learned by Lipschitz-VAEs, training standard and Lipschitz-VAEs with latent space dimension $d_z = 2$ and visualizing their learned encoders $q_\phi(\mathbf{z}|\mathbf{x})$. As shown in Figure B.8, the encoders learned by Lipschitz- and standard VAEs differ in their scale. Whereas the encoder of a standard VAE remains tightly clustered about the prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$, the encoders of the Lipschitz-VAEs disperse mass widely in latent space.

This apparent rescaling of the latent space in Lipschitz-VAEs has two important consequences, the first of which is that the prior and encoder have little overlap. This is significant because it is common to generate data points with a trained VAE by drawing samples from the prior and passing these to the decoder. In a rescaled latent space where the prior and encoder have little overlap, many samples from the prior will be "out-of-distribution" inputs to the decoder.

The second consequence of the latent space being rescaled is that there risks being less overlap between $q_\phi(\mathbf{z}|\cdot)$ for any two inputs. In the limit, the latent space then devolves into a look-up table (Mathieu et al., 2019), which is undesirable because the meaning of interpolated points in latent space — that is, points between areas of high density in terms of $q_\phi(\mathbf{z}|\cdot)$ — is lost.

We speculate that the rescaling of latent spaces in Lipschitz-VAEs can be explained by the relative importance of the likelihood and KL terms, $\log p_\theta(\mathbf{x}|\mathbf{z})$ and $\text{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right)$ respectively, in the VAE objective in (1). By Definition 2.3, a Lipschitz continuous function is one whose rate of change is constrained, so in some sense



(a) Standard VAE

(b) Lipschitz-VAE, $M = 5$

(c) Lipschitz-β-VAE, $M = 5$, $\beta = 5$

(d) Lipschitz-VAE, $M = 10$

(e) Lipschitz-β-VAE, $M = 10$, $\beta = 5$

Figure B.8: Learned encodings for different types of VAE on MNIST. A colored ellipse represents the posterior $q_\phi(\mathbf{z}|\mathbf{x}_i)$ for a single $\mathbf{x}_i$. The prior, $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$, is overlaid in black for one, two and three standard deviations. Lipschitz-VAEs have encoders that are dispersed in latent space, in contrast with the learned encoder of a standard VAE. Upweighting the KL term in (1), as in a β-VAE (Higgins et al., 2017a), changes this behaviour.

such a function is “simpler” than others not satisfying the property. It seems plausible then that — to achieve good input reconstructions while using simpler functions than a standard VAE — a Lipschitz-VAE might rescale the latent space to be able to adequately differentiate between latent samples corresponding to different inputs. This might happen even at the expense of the encoder being distant from the prior, causing $\mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right)$ to grow, since the likelihood term typically dominates the KL term and so gains in the likelihood term from rescaling the latent space might outweigh the resulting penalty from the KL term.

We test this hypothesis by training Lipschitz-VAEs with the KL term upweighted by hyperparameter β, as in a β-VAE (Higgins et al., 2017a) (we term Lipschitz-VAEs trained with this modified objective *Lipschitz-β-VAEs*). As can be seen in Figure B.8, and as predicted by our hypothesis, we find that by increasing the weight assigned to the KL term — that is, using $\beta > 1$ — the scaling of the latent space is mitigated.

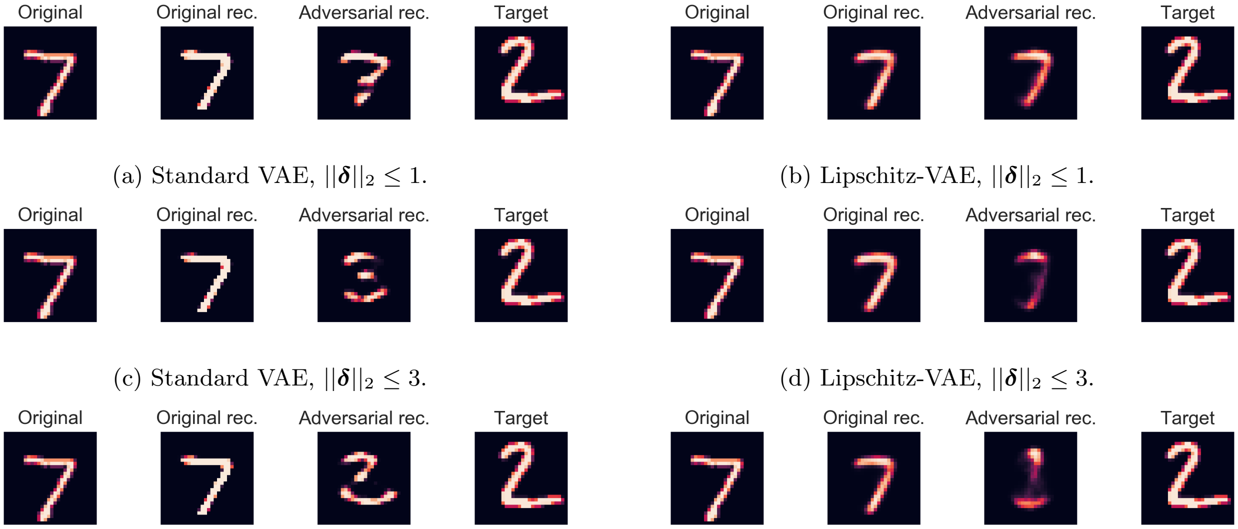
In sum, the experiments in this section reveal that Lipschitz-VAEs learn qualitatively different encoders from standard VAEs, exhibiting rescaling behavior that we link both to the challenge of performing reconstructions using Lipschitz continuous functions and the characteristics of the VAE objective. Our experiments also outline how possible adverse effects of Lipschitz continuity constraints on data generation and latent space interpretability might be addressed through a small modification of the VAE objective.

D ESTIMATING THE (r, ϵ)-ROBUSTNESS MARGIN

Algorithm 2 Camuto et al. (2020)’s algorithm to estimate (r, ϵ)-robustness margin $R^{(r,\epsilon)}(\mathbf{x})$. Starting with estimate `max_R` and decrementing by step size α at each iteration (until reaching 0), the algorithm performs T maximum damage attacks with input perturbations constrained to the current estimate for the (r, ϵ)-robustness margin. The first time (r, ϵ)-robustness is satisfied under all T attacks, the algorithm returns the current estimate as the estimated (r, ϵ)-robustness margin $\hat{R}^{(r,\epsilon)}(\mathbf{x})$.

```
Inputs : $\mathbf{x}$, $r$, $\epsilon$, starting estimate max_R, step size $\alpha$, number of samples $S$, number of random restarts $T$
Output: Estimated $(r, \epsilon)$-robustness margin $\hat{R}^{(r,\epsilon)}(\mathbf{x})$
Estimation routine
    $\hat{R}^{(r,\epsilon)}(\mathbf{x}) \leftarrow$ max_R
    while $\hat{R}^{(r,\epsilon)}(\mathbf{x}) > 0$ do
        probabilities $\leftarrow$ []
        for $t = 1, \dots, T$ do
            // Performs a maximum damage attack according to the objective in (4)
            $\boldsymbol{\delta}_t \leftarrow$ MaxDamageAttack with the constraint $||\boldsymbol{\delta}||_2 \leq \hat{R}^{(r,\epsilon)}(\mathbf{x})$; randomly initialized
            distances $\leftarrow$ []
            for $s = 1, \dots, S$ do
                $\mathbf{z}_{\boldsymbol{\delta}_t} \sim q_\phi(\mathbf{z}|\mathbf{x} + \boldsymbol{\delta}_t)$  $\mathbf{z}_{\neg\boldsymbol{\delta}_t} \sim q_\phi(\mathbf{z}|\mathbf{x})$
                distances.append($||g_\theta(\mathbf{z}_{\boldsymbol{\delta}_t}) - g_\theta(\mathbf{z}_{\neg\boldsymbol{\delta}_t})||_2$)
            // Estimates the $r$-robustness probability
            probability $\leftarrow \frac{\text{length(distances[distances} \leq r\text{])}}{S}$
            probabilities.append(probability)
        // Checks that the estimated probabilities are greater than $\epsilon$, across random restarts
        if length(probabilities[probabilities $> \epsilon$])$= T$ then
            return $\hat{R}^{(r,\epsilon)}(\mathbf{x})$
        $\hat{R}^{(r,\epsilon)}(\mathbf{x}) \leftarrow \hat{R}^{(r,\epsilon)}(\mathbf{x}) - \alpha$
    // Indicates when no positive $(r, \epsilon)$-robustness margin is found
    return “No positive $R^{(r,\epsilon)}(\mathbf{x})$ found.”
```

E QUALITATIVELY EVALUATING ROBUSTNESS



(a) Standard VAE, $||\boldsymbol{\delta}||_2 \leq 1$.

(b) Lipschitz-VAE, $||\boldsymbol{\delta}||_2 \leq 1$.

(c) Standard VAE, $||\boldsymbol{\delta}||_2 \leq 3$.

(d) Lipschitz-VAE, $||\boldsymbol{\delta}||_2 \leq 3$.

(e) Standard VAE, $||\boldsymbol{\delta}||_2 \leq 5$.

(f) Lipschitz-VAE, $||\boldsymbol{\delta}||_2 \leq 5$.

Figure E.9: Representative results from latent space attacks as in (3) on a standard VAE and a Lipschitz-VAE with Lipschitz constants 5. Each latent space attack looks for an input perturbation $\boldsymbol{\delta}$ such that, applied to an image of a written 7, the attacked VAE reconstructs an image resembling a written 2. From left to right in each subfigure: the original image of the written 7; a reconstruction of the original image, absent input perturbation; a reconstruction of the original image under input perturbation; the target image for the latent space attack, a written 2. A latent space attack is more successful when reconstructions of the original image under input perturbation more closely resemble the target image. We see latent space attacks are more successful in both the standard and Lipschitz-VAE as the norm of the perturbation $||\boldsymbol{\delta}||_2$ is allowed to increase (moving from top to bottom), but for a given perturbation norm are less successful on the Lipschitz-VAE (right column) than on the standard VAE (left column).

F EXPERIMENTAL SETUP

To properly handle reconstructions on $[0, 1]$-valued data, we let the likelihood in the VAE objective be Continuous Bernoulli (Loaiza-Ganem and Cunningham, 2019).

In the Lipschitz-VAEs we train, all activation functions bar the final-layer activations are the GroupSort activation (recall Section B.3), while in the standard VAEs we train, these are the ReLU. In both types of VAE, the final-layer activation in the encoder standard deviation $\sigma_\phi(\cdot)$ is the sigmoid function to ensure positivity, while the final-layer activation in the deterministic component of the decoder is the sigmoid function to ensure reconstructions are appropriate for binary data. The final layer of the encoder mean takes no activation function.

All models were trained on a 13-inch Macbook Pro from 2017 with 8GB of RAM and 2 CPUs.